Computer Security: Art and Science

Answer Key

Contents

Chapter 1

An Overview of Computer Security

1. Classify each of the following as a violation of confidentiality, of integrity, of availability, or of some combination thereof.

   (a) John copies Mary's homework.

   (b) Paul crashes Linda's system.

   (c) Carol changes the amount of Angelo's check from $100 to $1,000.

   (d) Gina forges Roger's signature on a deed.

   (e) Rhonda registers the domain name "Pearson.com" and refuses to let the publishing house buy or use that domain name.

   (f) Jonah obtains Peter's credit card number and has the credit card company cancel the card and replace it with another card bearing a different account number.

   (g) Henry spoofs Julie's IP address to gain access to her computer.

   Answer:

   (a) John copying Mary's homework is a violation of confidentiality. John should not see Mary's homework because to copy homework is cheating.

   (b) Paul crashing Linda's system is a violation of availability. Linda's system is no longer available to her, or anyone else.

   (c) Carol changing the amount of Angelo's check from $100 to $1000 is a violation of integrity (specifically, data integrity). The amount written on the check has been changed.

   (d) Gina forging Roger's signature on a deed is a violation of integrity (specifically, integrity of origin). The deed appears to have come from Roger, when in fact it came from Gina.

   (e) Rhonda registering the domain name "Pearson.com" and refusing to let the publishing house buy or use that domain name is a violation of availability. The name "Addison-Wesley" is not available to anyone, including the owner of that name, except Rhonda.

   (f) Jonah obtaining Peter's credit card number, and having the credit card company cancel the card and replace it with another bearing a different account, is a violation of integrity (specifically, integrity of origin). The request appears to come from Peter (else the credit card company would not have honored it), but in reality came from Jonah.

(g) Henry spoofing Julie's IP address to gain access to her computer is a violation of integrity (specifically, integrity of origin). The messages from Henry appear to come from Julie's IP address, when in fact they do not.

2. Identify mechanisms for implementing the following. State what policy or policies they might be enforcing.

   (a) A password-changing program will reject passwords that are less than five characters long or that are found in the dictionary.

   (b) Only students in a computer science class will be given accounts on the department's computer system.

   (c) The login program will disallow logins of any students who enter their passwords incorrectly three times.

   (d) The permissions of the file containing Carol's homework will prevent Robert from cheating and copying it.

   (e) When World Wide Web traffic climbs to more than 80% of the network's capacity, systems will disallow any further communications to or from Web servers.

   (f) Annie, a systems analyst, will be able to detect a student using a program to scan her system for vulnerabilities.

   (g) A program used to submit homework will turn itself off just after the due date.

   Answer:

   (a) The policy element is that easily guessed passwords are forbidden. The mechanism element is the program checking for, and rejecting, those passwords.

   (b) The policy element is that only students in that class may use the department's computer system. The mechanism element is the procedure of not giving other students an account.

   (c) The policy element is that only authorized users may log in. The mechanism element is that after three failed login attempts, the system disables the account to prevent further guessing of the password

   (d) The policy element is that no student may read another student"s homework. The mechanism element is the file protection mechanism that restricts read access.

   (e) The policy element is that World Wide Web traffic may not interfere with other network traffic, such interference being defined as using more than 80% of the bandwidth. The mechanism element is to block any traffic to or from Web servers.

   (f) The policy element is that systems may not be scanned for vulnerabilities. The mechanism element is whatever Annie used to detect the scanning.

   (g) The policy element is that late homework is not accepted. The mechanism element is the program disabling itself after the due date.

3. The aphorism "security through obscurity" suggests that hiding information provides some level of security. Give an example of a situation in which hiding information does not add appreciably to the security of a system. Then give an example of a situation in which it does.

   Answer:

   An example of a situation in which hiding information does not add appreciably to the security of a system is hiding the implementation of the UNIX password hashing algorithm. The algorithm can be determined by extracting the object code of the relevant library routine and disassembling it. (The library must be world readable in order for user programs to load the routine.) Revealing the algorithm does not appreciably simplify the task of an attacker because he knows how to hash passwords, but he still must guess the password itself.

   An example of a situation in which hiding information adds appreciably to the security of a system is hiding a password or cryptographic key. This is a private piece of information affecting only a single user. Revealing it would give an attacker immediate access to the system.

4. Give an example of a situation in which a compromise of confidentiality leads to a compromise in integrity.

   Answer:

   If the confidentiality of a password is compromised, the attacker may be able to impersonate a user authorized to change data. As integrity requires that only authorized users make only authorized changes to data, and the attacker is not an authorized user, there is a violation of integrity.

5. Show that the three security services—confidentiality, integrity, and availability—are sufficient to deal with the threats of disclosure, disruption, deception, and usurpation.

   Answer:

   Disclosure is the revealing of information, so the confidentiality security service is sufficient to deal with that threat. Disruption is the interruption or prevention of a service. The security service of availability counters interruption, and ensures the service can be supplied, counter- ing prevention also. Deception is the acceptance of false data. The data may be the contents of something, or the origin of something. The security service of integrity handles both of these. Usurpation is the unauthorized control of a service. The security service of integrity prevents an unauthorized user from altering the origin of the control of the service; the security service of availability ensures the authorized controller an still control the service.

6. In addition to mathematical and informal statements of policy, policies can be implicit (not stated). Why might this be done? Might it occur with informally stated policies? What problems can this cause?

   Answer:

   Policies may be implicit for a number of reasons. The policy may be ambiguous, and the resolution of the ambiguity left to the reader; thus, the exact policy is not

explicitly stated. The policy may not cover all aspects of the system; those aspects not covered by the explicit policy would presumably be covered by the implicit policy. The institution owning the computer may simply choose to tell users to use "common sense"; this is also an implicit policy. It is highly likely that informally stated policies will have many areas of ambiguity and not cover all contingencies. Hence these types of policies often lead to implicit policy components.

The main problem with implicit policies is that not all users may know about them, or may have agreed to them. The statement that "common sense is so unusual because it's not common" applies here. Given that people cannot refer to an oracle, or source, for an implicit policy but instead must gather opinions and make their own decisions, which may disagree with those of the system managers, a user may find herself violating the security policy without realizing it or intending to violate it.

7. For each of the following statements, give an example of a situation in which the statement is true.

   (a) Prevention is more important than detection and recovery.

   (b) Detection is more important than prevention and recovery.

   (c) Recovery is more important than prevention and detection.

   Answer:

   (a) An example of when prevention is more important than detection and recovery is the nuclear command and control system. By the time an intrusion is detected and recovered from, an attacker could have launched nuclear weapons.

   (b) An example of when detection is more important than prevention and recovery is in the protection of medical records from unauthorized emergency room personnel. If someone is brought into an emergency room, there may not be time to secure the patient's permission to access his medical records. But if the records are accessed illicitly, the security personnel should detect it.

   (c) An example of when recovery is more important than prevention and detection is on a banking computer that maintains account balances. The bank must be able to recover the balance of all accounts to ensure it provides accurate service to its customers. Prevention and detection, while important, are not so important as keeping the balances accurate.

8. Is it possible to design and implement a system in which no assumptions about trust are made? Why or why not?

   Answer:

   It is not possible to design and implement a system in which no assumptions about trust are made. Designing and implementing any system involves people, and the people must be trusted to design and implement the system correctly. If one does not trust the people, their work must be checked, and the people doing the checking must be trusted. Iterating this lack of trust demonstrates that some people doing checking must be trusted, unless the checking is automated. But in that case, people implemented the automated checker. This is equivalent to the previous case.

9. Policy restricts the use of electronic mail on a particular system to faculty and staff. Students cannot send or receive electronic mail on that host. Classify the following mechanisms as secure, precise, or broad.

   (a) The electronic mail sending and receiving programs are disabled.

   (b) As each letter is sent or received, the system looks up the sender (or recipient) in a database. If that party is listed as faculty or staff, the mail is processed. Otherwise, it is rejected. (Assume that the database entries are correct.)

   (c) The electronic mail sending programs ask the user if he or she is a student. If so, the mail is refused. The electronic mail receiving programs are disabled.

   Answer:

   (a) The mechanism is secure, because students cannot send or receive electronic mail on the system. It is not precise, as faculty cannot send or receive electronic mail on the system, and the security policy says they are allowed to.

   (b) This mechanism is precise, because any mail from or to students is discarded. (You can argue this is broad, because students can execute the "send mail" command, but the mail will never leave the machine. The word "send" is somewhat ambiguous.)

   (c) This mechanism is broad, because a student can claim to be a faculty member when answering the question.

10. Consider a very high-assurance system developed for the military. The system has a set of specifications, and both the design and implementation have been proven to satisfy the specifications. What questions should school administrators ask when deciding whether to purchase such a system for their school's use?

    Answer:

    Some example questions follow.

    (a) Are the specifications appropriate for an educational institution? For example, will the military system meet the availability needs of the university?

    (b) What assumptions about the operating environment does the military system make? Are the assumptions valid in the school's operating environment?

    (c) What procedures must be followed to record and distribute grades? Does the system specification assure this can be done in a way that meets the requirements of the university?

11. How do laws protecting privacy impact the ability of system administrators to monitor user activity?

    Answer:

    Laws protecting privacy forbid the collection of some types of data. The goal of these laws is to prevent an organization, or individuals, from inferring information about individuals' beliefs, behavior, or other personal characteristics from the data being

transmitted. When monitoring user activity, privacy laws affect system administrators because they cannot observe certain data relating to user activity. For example, a user may read private e-mail from her spouse. The contents of that e-mail, if protected by privacy laws, must be suppressed when the system administrator records network traffic. So the system administrators must devise a method to conceal or scramble the information (called sanitization). The problem becomes more complex when the information is relevant to a security analysis. For example, consider a sweep of a network looking for HTTP servers. That this is a sweep will be obvious when the IP addresses are correlated: every IP address on the network will have been probed. But the IP addresses may tie machine use to an individual user, so a law restricting the ability of the system administrator to tie actions to specific users may prevent the recording of the IP addresses. This would hinder the security analysis of the user activity, because some of those activities could not be recorded.

12. Computer viruses are programs that, among other actions, can delete files without a user's permission. A U.S. legislator wrote a law banning the deletion of any files from computer disks. What was the problem with this law from a computer security point of view? Specifically, state which security service would have been affected if the law had been passed.

    Answer:

    The problem with the proposed law was that any deletion was forbidden. As written, if someone dragged a file to the trash can or recycle bin (or otherwise deleted the file), that person would violate the law. Further, not all viruses delete files. Some transmit information; others insert back doors (indeed, Cohen's early viruses were of this type). So the law would not achieve its desired purpose, and indeed would criminalize acts that have nothing to do with computer viruses. The specific security services that could be affected by this law would be availability (if you can't delete files, you will run out of room on the disk) and integrity (the system may require that certain files be deleted to function correctly).

13. Users often bring in programs or download programs from the Internet. Give an example of a site for which the benefits of allowing users to do this outweigh the dangers. Then give an example of a site for which the dangers of allowing users to do this outweigh the benefits.

    Answer:

    An example of a site at which the benefits of allowing users to download programs outweigh the dangers would be a university. Much of the free software that universities depend on, such as the text editor *emacs*, must be downloaded. Without these free programs, students would not be exposed to such a wide variety of software and systems, and this would adversely affect their education. Further, the students rarely have the privilege to alter system programs, so they can damage only their protection domain if they download malicious code.

    An example of a site at which the dangers of allowing users to download programs outweigh the benefits would be a site at which sensitive data is handled, such as

a medical insurance company (patient medical records) or a classified facility. The problem is that the downloaded code could transmit, alter, or delete data, and the data is very sensitive to exposure or unauthorized alteration. If damaged, reconstructing the data would be very expensive (if the data could be reconstructed); if made public, the damage could not be undone.

14. A respected computer scientist has said that no computer can ever be made perfectly secure. Why might she have said this?

    Answer:

    When the respected computer scientist said that no computer can ever be made perfectly secure, she was probably thinking about the people who use it. No matter how secure the system, some of the users, administrators, and programmers have access to information on the system, and the ability to alter the system programs. (Two or more people may need to work together for this purpose.) The human element here is the weak point, because people can be corrupted or threatened, or otherwise persuaded to breach system security.

15. An organization makes each lead system administrator responsible for the security of the system he or she runs. However, the management determines what programs are to be on the system and how they are to be configured.

    (a) Describe the security problem(s) that this division of power would create.

    (b) How would you fix them?

    Answer:

    (a) The division of power gave the system administrators the responsibility for securing the systems, but denied them the power to determine what programs could be run and how the systems were to be configured. Responsibility without power is untenable because the matter for which one bears responsibility is not under one's control. So, the system administrators were (essentially) scapegoats.

    (b) The best way to fix the problem is to allow the system administrators to determine what programs could be run and how their systems would be configured. So, the managers (and system administrators) would together set a reasonable policy, and then the job of the system administrators would be to ensure their systems (and their system interactions) conform to the policy. This way, the management goals with respect to "security" are clearly stated, and the system administrators are given both the power and the responsibility for ensuring the policy is met on the actual systems.

16. The president of a large software development company has become concerned about competitors learning proprietary information. He is determined to stop them. Part of his security mechanism is to require all employees to report any contact with employees of the company's competitors, even if it is purely social. Do you believe this will have the desired effect? Why or why not?

Answer:

The president's edict raises several issues.

First, will it solve the problem? If the employees are not involved, the measure will not help the situation, and could make matters worse (see below). If the employees are involved, presumably not all of them are involved, so measures that would be effective against the culprits should be taken. If it is not known whether any employees are involved, the intent of this method seems to be that, if the leaks stop, then the employees are leaking the information. But the leaks stopping could also be due to the leaker becoming nervous and deciding to lay low while the ban is in effect, or for a variety of reasons unrelated to the ban. A more precise method of determining which employees, if any, were leaking should be used.

Second, how will the employees feel about it? If the employees understand the reason for the measure, and accept it, there will be no problem. But some employees may feel that the need to report even social contacts is an infringement on their personal lives. These people may resent the edict, and may not comply. Even those who comply may resent the intrusion into their personal lives. Such a situation would be disastrous for employee morale, and may lead to more problems than the leak of proprietary information.

This raises a critical point: how can the president enforce his rule? Consider the case of a corrupt employee who has a role in competitors learning proprietary information. How likely is that employee to report his or her contacts with the competitor's employees? Unless the president has a way of validating that all contacts are indeed reported, the result of the measure seems to be that the honest employees will comply and the dishonest ones will not—achieving exactly the opposite of the goal of the edict.

So, whether this measure has the desired effect depends on two factors. First, if the president can verify that no contacts other than those reported have occurred, then the measure would show which employees are talking to people from the competitors. Second, if the president can establish that information is leaking through contacts such as those, then the president will know which subset of employees have to be watched. But both of these hypotheticals are highly unlikely, for the reasons given above. Further, the edict could hurt morale severely, leading to a loss of productivity and of key people.

17. The police and the public defender share a computer. What security problems does this present? Do you feel it is a reasonable cost-saving measure to have all public agencies share the same (set of) computers?

Answer:

The biggest problem is that of public confidence. Unless the public (including the police and the public defenders) can be confident that the information belonging to both parties can be kept confidential and trustworthy, they will lose confidence in both organizations. The problem is that attacks or other failures may cause this information to be revealed. There are three cases.

The first case is where the police can access the public defenders' files. They could then see confidential information (because what a defendant tells his or her lawyer is generally privileged, and may not be revealed). This means that lawyer-client confidentiality can be breached. The problem here is not that the police would deliberately do so, but that they might do so unintentionally. Acting on such information risks compromising the case they are working on, because (at least in U. S. courts) evidence obtained in violation of the law cannot be used in court.

The second case is where the public defenders can access the police files. Most of the information in those files will become available through the judicial process, but some parts deemed not relevant by the judge may remain secret. Further, ancillary information unrelated to the defendants will also be secret. Thus, the public defenders would have access to information unrelated to their cases, or to information that they would not otherwise see. This could compromise police investigations.

The third case is where an outside attacker breaks into one group's part of the computer system. In that case, all data in *both* parts of the system is at risk, because they both reside on the same system and, depending on the nature of the compromise, the entire system may be at risk.

Whether this is a reasonable cost-saving measure depends on a number of factors such as the prevalence of crime, the cost of the second system, and the jurisdiction's budget, among other things. We leave these arguments to the reader.

18. Companies usually restrict the use of electronic mail to company business but do allow minimal use for personal reasons.

    (a) How might a company detect excessive personal use of electronic mail, other than by reading it? (*Hint*: Think about the personal use of a company telephone.)

    (b) Intuitively, it seems reasonable to ban *all* personal use of electronic mail on company computers. Explain why most companies do not do this.

    Answer:

    (a) Companies can detect excessive personal use of a telephone by looking at the numbers dialed. If those numbers belong to people not related to, or involved in, the company?s business, the company may investigate further to determine if the employee is using the phone for too much personal business. Similarly, with electronic mail, the company can note the outgoing addresses, and from those determine if the employee is using email for personal business. These methods are typically cumbersome and require investigation, so they tend not to be used unless phone calls or email is severely affecting the budget of the organization or the productivity of the employees.

    (b) Banning all personal use of electronic mail might significantly decrease the time employees spend working. Should a personal call need to be made (or received), the employee would have to find a phone not belonging to the employer. This could take considerably more time than simply making the call from the employee?s phone (for example, if the employee has to go out of the building and

across the street to a drug store or gas station). An additional factor is employee morale; knowing that the employer does not trust employees enough to control their personal calls can hurt morale.

19. Argue for or against the following proposition. Ciphers that the government cannot cryptanalyze should be outlawed. How would your argument change if such ciphers could be used provided that the users registered the keys with the government?

    Answer:

    In what follows, "legal cipher" means one that the government can cryptanalyze and "illegal cipher" means one the government cannot cryptanalyze.

    Some of the typical arguments in favor of this approach include:

    - Criminals who use encrypted messages to conceal their actions from the authorities would no longer be able to do so. So, if the government could not read the message, they could prosecute the sender for using the legal cipher.
    - The government could decode messages and examine who sending encrypted messages, and why. This may help intelligence and police services anticipate crimes or attacks, and warn potential victims or defend against attacks.
    - Honest people don't have anything to hide. If someone needs to use encryption, they must be hiding something, and therefore are suspect. The government is entitled to ensure what they are concealing does not threaten other people or the nation.

    Typical arguments against this approach include:

    - Criminals who are engaged in nefarious acts will use illegal ciphers, especially if the penalties for using them are less than those for committing the nefarious acts. Further, in many countries, prosecutors can require the user to produce the cryptographic key, for example pursuant to a judicial order in the United States. Failure to comply results in going to jail. Thus, criminalizing the use of an illegal cipher is redundant at best.
    - Requiring the use of legal ciphers may lead people to believe that any cipher used can be read by the government. In fact, people may use illegal ciphers to conceal information; history shows that the knowledge of, and use of, such ciphers simply cannot be suppressed (especially in these days of global information and instant communication). Thus, if the purpose of requiring the use of legal ciphers is to provide security, the inability to prevent the use of illegal ciphers detracts from that security. Thus, the law provides a false sense of security.
    - If a government can break a cipher, then it is likely that at least some commercial firms can do so—or will be believed to be able to do so. Such firms may obtain their competitors' secrets by breaking the ciphers. Similarly, as the government can obtain that information, insiders or corrupt government employees could leak the information—or the information could leak by accident.

- Given the massive amounts of message traffic sent within most nations, it is unclear whether a government could decrypt messages quickly enough to take preventative measures based on information in the decrypted messages. More likely, the government will record messages and, *after* a nefarious event occurs, go back and decrypt messages to learn the "back story" for future use (in prosecutions and to learn what went wrong, if anything). Thus the argument for prevention seems weak at best, unless the government has reason to suspect the sender or receiver.
- Honest people *do* have things to hide. Everyone has done things, or been accused of doing things, that they would find embarrassing or wish to forget—even though those things may be perfectly legal. Government has no business violating the privacy that protects people from embarrassment, humiliation, or other shame even if the embarrassment, humiliation, or shame arises from that which is not illegal.

Ultimately, the three issues are:

(a) Does requiring the use of legal ciphers benefit the nation?

(b) Do people trust the government (in particular, the specific government agencies and employees involved) to break ciphers, act rationally upon the information decrypted, and protect the information so obtained?

(c) Can make the cost (including the non-financial cost) of using an illegal cipher be made sufficiently high that no-one within the country, *or outside the country and communicating with people in the country*, would want to use an illegal cipher?

The same arguments, and questions, apply to registering keys, as registering a key effectively gives the government the ability to break the message.

20. For many years, industries and financial institutions hired people who broke into their systems once those people were released from prison. Now, such a conviction tends to *prevent* such people from being hired. Why you think attitudes on this issue changed? Do you think they changed for the better or for the worse?

Answer:

Until sometime in the mid-1980s, people who understood the security of computers and systems were considered rare. Someone who could break into a system therefore was believed to understand how to evade very strong controls, and therefore know how to prevent others from evading them. So, at that time, many thought that such attackers would be able to improve the security of their systems by implementing better controls and detecting attacks that more conventionally trained security officers would miss.

After that time, though, things changed. Extensive knowledge of security, and a system, no longer became a prerequisite for breaching security. Security controls became better, as did detection methods, and so those who were caught could rarely improve the security of the system. Hence attitudes changed.

One other issue has always been the trustworthiness of the attackers who were caught. Whether someone who attacked a site could be trusted after being caught and sent to jail is a delicate question that each institution must make based on a variety of factors, including the people involved, the institutional mission, the stakeholders, and many other elements. Perhaps changes in these factors, or in their weighing, also led to a change in attitudes.

21. A graduate student accidentally releases a program that spreads from computer system to computer system. It deletes no files but requires much time to implement the necessary defenses. The graduate student is convicted. Despite demands that he be sent to prison for the maximum time possible (to make an example of him), the judge sentences him to pay a fine and perform community service. What factors do you believe caused the judge to hand down the sentence he did? What would you have done were you the judge, and what extra information would you have needed to make your decision?

    Answer:

    Among the factors involved in sentencing are deterrence, restitution, and punishment. Let us focus on these factors.

    The judge probably considered the amount of damage the program did. It clearly required much effort to clean up, but everyone agreed that no files were deleted. Thus, the damage was not as severe as it could have been. Further, cleaning up after attacks is arguably part of a system administrator's job, so the money lost by the clean-up could be seen as minimal also (overtime is very rare for system administration jobs). Thus, restitution by helping the community in some way, to compensate society for the peoples' time that was taken up by the recovery, seems more appropriate than locking the graduate student up and thus not providing any type of restitution.

    A second question is deterrence. The program was released accidentally, so there was no maliciousness in its release — just carelessness or stupidity. It is unclear how a prison sentence would deter others from making mistakes like this; more likely, a prison sentence would inhibit research and class exercises in certain types of programs (notably self-replicating code and some types of malware). But the fine and community service would encourage people to be very careful when working with programs like this.

    Third, if this was a first offense, rehabilitation would be much more appropriate than prison for this type of non-violent crime that had no specific victims. The student was not trying to rob people, or steal money, or disable critical infrastructure. Sending him a strong message without ruining his life seems very appropriate — the punishment of community service would force him to help others, teaching them how to use computers safely and with consideration for others. This serves both him and society much better than simply locking him up.

    These may have been some of the factors the judge considered.

Chapter 2

Access Contol Matrix

1. Consider a computer system with three users: Alice, Bob, and Cyndy. Alice owns the file *alicerc*, and Bob and Cyndy can read it. Cyndy can read and write the file *bobrc*, which Bob owns, but Alice can only read it. Only Cyndy can read and write the file *cyndyrc*, which she owns. Assume that the owner of each of these files can execute it.

   (a) Create the corresponding access control matrix.

   (b) Cyndy gives Alice permission to read *cyndyrc*, and Alice removes Bob's ability to read *alicerc*. Show the new access control matrix.

   Answer:

   (a)

	alicerc	*bobrc*	*cyndyrc*
Alice	ox	r	
Bob	r	ox	
Cyndy	r	rw	orwx

   (b)

	alicerc	*bobrc*	*cyndyrc*
Alice	ox	r	r
Bob		ox	
Cyndy	r	rw	orwx

2. Consider the following change in the rules associated with each (object, verb) pair in Miller and Baldwin's model (see Section 2.2.1):

name	*rules*
recipes	write: 'creative' in subject.group and 'chef' in subject.role
overpass	write: 'artist' in subject.role and 'creative' in subject.group
.shellrct	write: 'hack' in subject.group and (time.hour $<$ 4 or time.hour $>$ 20) and time.hour $>$ 0
oven.dev	temp_ctl: 'kitchen' in subject.program and 'chef' in subject.role

   How does this change the access control matrices shown at the end of that section?

   Answer:

   Between 8PM and 4AM, the access control matrix is:

	recipes	*overpass*	*.shellrct*	*oven.dev*
matt	read	read	read, write	
holly	read	read, write	read	
heidi	read, write	read	read	temp_ctl

After 4AM and before 8PM, the access control matrix is:

	recipes	*overpass*	*.shellrct*	*oven.dev*
matt	read	read	read	
holly	read	read, write	read	
heidi	read, write	read	read	temp_ctl

3. Consider a mechanism that *amplifies* rights instead of reducing them. Associate with each (system) routine a *template* of rights. When the routine runs, the rights of the process are augmented by the rights in the associated template, and when the routine exits, the rights added by the template are deleted. Contrast this with the mechanism described in Section 2.2.2.

   (a) What are some of the advantages and disadvantages of the amplification mechanism?

   (b) What are some of the advantages and disadvantages of the reducing mechanism?

   Answer:

   (a) The amplification mechanism allows a process to increase its rights temporarily. This is an advantage in that rights needed to perform some action may be given temporarily to the process, so it need not start with those rights; but it is also a disadvantage, in that the acquisition of those rights enables the process to use them in ways perhaps not contemplated, for example if the routine is poorly programmed.

   (b) The reducing mechanism does not allow for the temporary restoration of rights. Thus the process must start with rights sufficient to perform its task, and must not run any routine that eliminates those rights—meaning the routine's static rights must include those rights. The advantage of this method is it eliminates rights, inhibiting the flow of information to objects the access to which has been eliminated by a routine.

4. Consider the set of rights { *read*, *write*, *execute*, *append*, *list*, *modify*, *own* }.

   (a) Using the syntax in Section 2.3, write a command *delete_ all_ rights*(p, q, o). This command causes p to delete all rights the subject q has over an object o.

   (b) Modify your command so that the deletion can occur only if p has *modify* rights over o.

   (c) Modify your command so that the deletion can occur only if p has *modify* rights over o and q does not have *own* rights over o.

   Answer:

   In these answers, r, w, x, a, l, m, and o represent the read, write, execute, append, list, modify, and own rights, respectively.

   (a) The key observation is that anyone can delete the rights, not p. So:

```
command delete_all_rights(p, q, s)
        delete r from A[q, s];
        delete w from A[q, s];
        delete x from A[q, s];
        delete a from A[q, s];
        delete l from A[q, s];
        delete m from A[q, s];
        delete o from A[q, s];
end
```

(b) Here, we must condition the command on the presence of rights that p has over s:

```
command delete_all_rights(p, q, s)
        if m in A[p, s] then
                delete r from A[q, s];
                delete w from A[q, s];
                delete x from A[q, s];
                delete a from A[q, s];
                delete l from A[q, s];
                delete m from A[q, s];
                delete o from A[q, s];
end;
```

(c) This one is trickier. We cannot test for the absence of rights directly, so we build a surrogate object z. The idea is that A[q, z] will contain the right o if q does not have o rights over s, and will contain the right m if q has o rights over s. So, we need some auxiliary commands:

```
command make_aux_object(q, z)
        create object z;
        enter o into A[q, z];
end;
command fixup_aux_object(q, s, z)
        if o in A[q, s] then
                delete o from A[q, z];
                enter m into A[q, z];
end;
```

Now we write the command to delete the rights if p has m rights over s and q does not have o rights over s. The last condition is logically equivalent to q having o rights over z:

```
command prelim_delete_all_rights(p, q, s, z)
        if m in A[p,s] and o in A[q,z] then
                delete r in A[q, s];
                delete w in A[q, s];
                delete x from A[q, s];
                delete a from A[q, s];
                delete l from A[q, s];
                delete m from A[q, s];
```

```
                delete o from A[q, s];
end;
```

Finally, we create the actual delete command:

```
command delete_all_rights(p, q, s, z)
        make_aux_object(q, z);
        fixup_aux_object(q, s, z);
        prelim_delete_all\_rights(p, q, s, z);
        destroy object z;
end;
```

5. Let c be a copy flag and let a computer system have the same rights as in Exercise 4.

   (a) Using the syntax in Section 2.3, write a command *copy_ all_ rights*(p, q, s) that copies all rights that p has over s to q.

   (b) Modify your command so that only those rights with an associated copy flag are copied. The new copy should *not* have the copy flag.

   (c) In the previous part, what conceptually would be the effect of copying the copy flag along with the right?

Answer:

The copy flag defines an additional set of rights. We treat r (*read* right) and rc (*read* right with copy flag) as read rights, but the first may not be copied whereas the second may be.

(a) We build this command from a set of smaller commands. First, we define *copy_ r*(p, q, s) as:

```
command copy_right(r, p, q, s)
        if r in A[p, s] then
                enter r into A[q, s]
end
```

Then the required command is:

```
command copy_all_rights(p, q, s)
        copy_right(r, p, q, s)
        copy_right(rc, p, q, s)
        copy_right(w, p, q, s)
        copy_right(wc, p, q, s)
        copy_right(x, p, q, s)
        copy_right(xc, p, q, s)
        copy_right(a, p, q, s)
        copy_right(ac, p, q, s)
        copy_right(l, p, q, s)
        copy_right(lc, p, q, s)
        copy_right(m, p, q, s)
        copy_right(mc, p, q, s)
```