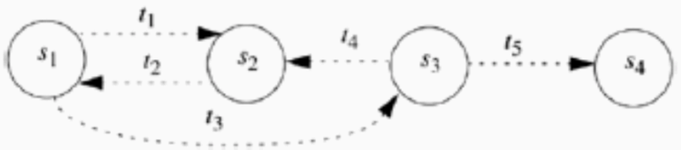
Exercise 4.8 #1:

In Figure 4-1, suppose that edge t_3 went from s_1 to s_4. Would the resulting system be secure?

Figure 4-1. A simple finite-state machine. In this example, the authorized states are s_1 and s_2.



If the edge went from s1 to s4, then the system would be insecure because s_4 is considered part of the unauthorized states, (UA = {s3, s4}). If the edge from s_1 to s_4 were to exist, then the system would enter an unauthorized state from an authorized state.
(adapted from Annie Howard)

ex2
revise the permission
find third person
ex3

3. (a) Assume that the system has no integrity controls. This is true. If a system lacks integrity, then data can be changed without restraint. So, anyone can change another user's authentication information, allowing them access to that user's account—and allowing them to see any data for that user or, by generalizing this in the obvious way, any user on the system. If some integrity controls work, then the ability of the system to provide confidentiality depends on the effectiveness of the integrity controls and their use to protect critical information.

   b. Assume that the system has no confidentiality controls. If there is no confidentiality, then all authentication information will be available. Unless authentication mechanisms do not use secret information (for example, biometrics or positions), any user can authenticate as another user. Hence there is no integrity. Now suppose authentication information does not rely on confidentiality. Can the data in a file be kept confidential? To do so, either the user must be prevented from reading the file (for which there are no controls) or from reading the data in the file (for example, by cryptography). In the latter case, if the data is encrypted on the system, the key must be available, and as there is no confidentiality, the key can be read. If the data is not encrypted on the system, then the data cannot be used on the system but will remain confidential. So, the answer is that the system cannot provide integrity. But if data is protected when placed on the system, it will remain protected as long as confidentiality mechanisms were applied to the data itself (and not to the containing object) and the mechanisms to undo them are not on the system.

A noted computer security expert has said that without integrity, no system can provide confidentiality.1. Do you agree? Justify your answer.
Integrity means that information is correct, and that data has not been corrupted in any way. integrity ensures that information has not been compromised, that the information is valid and is a result of authenticated and controlled activities. If we don't have any way to confirm and ensure that this is true, we can't guarantee confidentiality. (Additional note on question 3).

Can a system provide integrity without confidentiality? Again, justify your answer
In a case where sensitive information needs to be protected it cannot provide integrity if it don't have Confidentiality. Confidentiality requires authentication of people, and integrity requires the information to be a result of authenticated and controlled activity.
(Additional note on question 3).

ex4

4. The problem with the cryptographer's claim is how to protect the keys. At some point, the cryptographic keys must be available to encipher, decipher, or validate the integrity of data. If the keys are kept in memory, they must be protected, either by other cryptographic keys (which require similar protection) or by non-cryptographic access control mechanisms. If the keys are kept off-line (for example, in a smart card or a dongle), access to the external unit must be protected either by cryptographic keys (which require the protection discussed earlier) or non-cryptographic access control. By a simple process of induction, or *reductio ad absurdam*, non-cryptographic based access control mechanisms must be used at some point, refuting the cryptographer's claim.

ex5

5. Classify each of the following as an example of a mandatory, discretionary, or originator controlled policy, or a combination thereof. Justify your answers.

   1. The file access control mechanisms of the UNIX operating system
      discretionary access control
      Since users can assign and modify permissions that they possess, access control is discretionary.

   2. A system in which no memorandum can be distributed without the author's consent
      originator access control
      This would be originator access control. This is because if I am the author of the memorandum I am the one who can say my information can be distributed, no one else can.

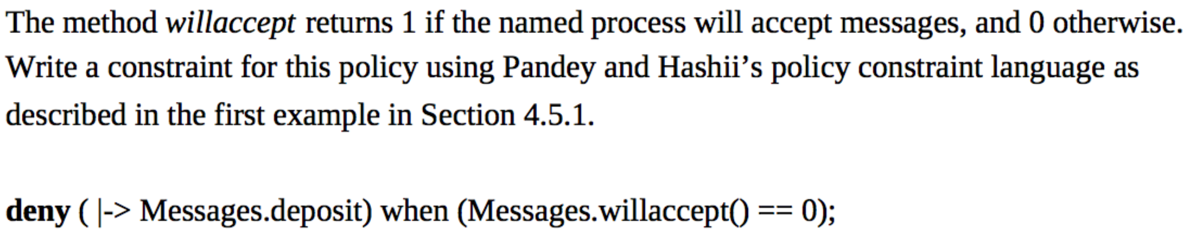
   3. A military facility in which only generals can enter a particular room
      mandatory access control
      The system controls access and an individual cannot change that. There is a somewhat tricky scenario though that could possibly make this discretionary; if there is an owner of the 'military facility' and this person also had the ability to promote military personnel to 'general'. In this way the facility owner could grant access to their facility.

   4. A university registrar's office, in which a faculty member can see the grades of a particular student provided that the student has given written permission for the faculty member to see them.

discretionary access control
Here the student grants the permission to the faculty to see the grades. If he doesn’t grant permission to a particular faculty member, that faculty member can’t see the grades.

This is a combination of an originator controlled access control policy and a discretionary access control policy. The originator, which is the registrar, controls dissemination of the data, but the student also has some control, and allows access to the individual record based upon the identity of the faculty member.

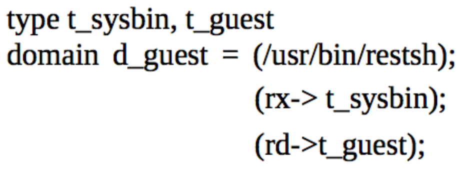
ex6



The method *willaccept* returns 1 if the named process will accept messages, and 0 otherwise. Write a constraint for this policy using Pandey and Hashii’s policy constraint language as described in the first example in Section 4.5.1.

deny (|-> Messages.deposit) when (Messages.willaccept() == 0);

ex7



```
type t_sysbin, t_guest
domain  d_guest  =  (/usr/bin/restsh);
                    (rx-> t_sysbin);
                    (rd->t_guest);
```

ex8

8. Suppose one wishes to confirm that none of the files in the directory */usr/spool/lpd* are world readable.

b. What would the second field of the RIACS database contain?

c. *Tripwire* does not provide a wildcard mechanism suitable for saying, "all files in the directory */usr/spool/lpd* beginning with *cf* or *df*." Suggest a modification of the *tripwire* configuration file that would allow this.

b) The second field of the RIACS database contains the permissions of the directory. The field must have the last digit less than 4 since it represents world permission with read as the first bit.

c) To support wildcard, we could let *tripwire* configuration file support a pattern, such as /usr/spool/lpd/[cd]f*.

ex9
chapter 4.6