

What to turn in: Submit hardcopy answers to the questions below. Please include your name, e-mail address, and the number of hours you spent working on the assignment.

1. True or False

Indicate the appropriate answer.

- a. T F A good password for human authentication should contain about 64 bits of random information, as provided by a typical mixed-case, alphanumeric, 8-character ASCII string.
- b. T F Digital signatures require the property of *nonrepudiation*, which says that a principal should not be able to spoof another principal's signature.
- c. T F Kerberos is an example of an arbitrated protocol.
- d. T F Good protocol design suggests that message formats be kept as uniform as possible to simplify the end hosts and reduce the trusted computing base.
- e. T F It is infeasible in practice to arrange for users of a computer system to authenticate using one-time passwords.
- f. T F For shared-key protocols that employ a trusted third party (like Needham–Schroeder and Kerberos), it is necessary to distribute $O(n^2)$ keys before any session keys may be generated. (Here, n is the number of principals.)
- g. T F Under mandatory access controls, the creator of a file may grant permissions to any users of their choice.
- h. T F Sequence numbers, timestamps, and nonces are used to prevent replay attacks.
- i. T F If there are n subjects and m objects in an access control matrix, it can take $O(n)$ time to revoke all rights of a given subject when the matrix is represented as capabilities

2. Cryptographic Protocols

Find a flaw in the following authentication protocol that establishes a session key k for use by two principals A and B . In this scenario, A and B want to authenticate to each other using an existing shared (symmetric) key K_{AB} . N_A and N_B are nonces generated by A and B , respectively, and the session key k is freshly created by B . Assume nonces and keys have the same number of bits.

1. $A \xrightarrow{A, N_A} B$
2. $A \xleftarrow{K_{AB}\{N_A\}, N_B} B$
3. $A \xrightarrow{K_{AB}\{N_B\}} B$
4. $A \xleftarrow{K_{AB}\{k\}} B$

ANSWER: An intruder M can impersonate B and trick A into using a key k_M of its choice. Note that this depends on nonces and keys being the same size. If M doesn't want to choose a key, he can simply send A the nonce N_B to use as a key in step 4 rather than initiating another run of the protocol with B .

1. $A \xrightarrow{A, N_A} M \xrightarrow{A, N_A} B$
2. $A \xleftarrow{K_{AB}\{N_A\}, N_B} M \xleftarrow{K_{AB}\{N_A\}, N_B} B$
3. $A \xrightarrow{K_{AB}\{N_B\}} M \xrightarrow{K_{AB}\{N_B\}} B$
- 1'. $A \quad M \xrightarrow{A, k_M} B$
- 2'. $A \quad M \xleftarrow{K_{AB}\{k_M\}, N'_B} B$
4. $A \xleftarrow{K_{AB}\{k_M\}} M$

Using a similar trick, M can also pretend to be A , although that attack does not reveal the shared key to M .

4. Unix Access Control

Recall that RUID stands for “Real User ID”, EUID stands for “Effective User ID”, and SUID stands for “Saved User ID”; recall also that programs can change their EUID (in restricted ways) by making the `seteuid` system call.

Suppose that a Unix directory contains the following files with permissions set as shown. Assume that all SetGID and Sticky bits are turned off.

File Description			Permissions			
Filename	Owner	Group	SetUID	Owner	Group	Other
foo.txt	15	15	-	rw-	r--	---
bar.txt	15	99	-	---	rw-	rw-
baz.txt	75	75	-	rw-	-w-	---
quk.txt	25	25	-	rw-	r--	r--
wordpro	25	99	y	--x	--x	---
Userver	25	75	-	--x	--x	---

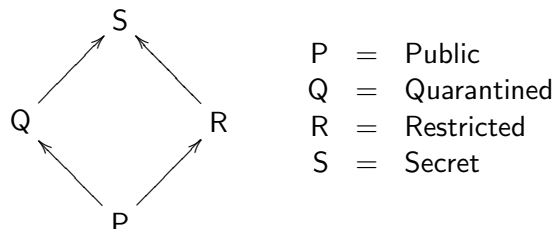
Assume that user 15 is in groups 15, 99, and 75.

Assume that user 25 is only in group 25.

- Consider a process running with RUID=25, EUID=25, and SUID=25. Assuming no other process changes any of the file permissions, which of these files could it read?
bar.txt, quk.txt, wordpro, Userver
- Consider a process running with RUID=15, EUID=15, and SUID=15. Assuming no other process changes any of the file permissions, which of these files could it write?
foo.txt, bar.txt, baz.txt
- Suppose a process running with RUID=0, EUID=0, and SUID=0 calls the `exec` system call to run the program `wordpro`. Assuming no other process changes any of the file permissions, which of these files could that instance of `wordpro` read?
all files
- Suppose a process running with RUID=15, EUID=15, and SUID=15 calls the `exec` system call to run the program `Userver`. Assuming no other process changes any of the file permissions, which of these files could that instance of `Userver` execute?
foo.txt, bar.txt, wordpro, Userver

5. Multilevel Security (15 points)

Consider a Bell & LaPadula style multilevel security setting in which all data is labeled with one of four security levels drawn from the following hierarchy (higher is more confidential):



Consider the following program that calculates output variables v , w , x , y , and z from input variables p , q , r , and s . Assume that the input variables have security labels that correspond to their names (i.e. variable p has label P , etc.). Assume that all output variables are initially set to 0.

```
v = p + q;
if (r > 0) then {
  w = 1;
  if (v > 0) then {
    x = 1;
  }
} else {
  y = 1;
}
z = s - s;
```

For each output variable indicate the *minimal* security label it can be given so that system is secure according to the information-flow policy. Each answer should be one of $\{P, Q, R, S\}$.

- label of $v = Q$
- label of $w = R$
- label of $x = S$
- label of $y = R$
- label of $z = P$