

1	/14
2	/16
3	/16
4	/10
5	/14
6	/5
7	/5
8	/20
9	/35
Total	/135

- Do not begin the exam until you are told to do so.
- You have 120 minutes to complete the exam.
- There are 11 single-sided pages; please sign your name on *all* of them.

1. (14 points) True or False? Circle the appropriate answer.

- a. T F Improperly applied cryptography is the #1 source of security violations in the Internet today.
- b. T F It is impossible to write a program that detects all viruses without generating false positives.
- c. T F Denial of service attacks against a web server can be prevented by having the web server handle only a fixed number of requests at any given time.
- d. T F To achieve the highest TCSEC rating, a piece of software must have a formally specified and verified security model.
- e. T F Under mandatory access controls, the creator of a file may grant permissions to any users of their choice.
- f. T F A digital watermark should be imperceptible and hard to remove.
- g. T F The X.509 certificates used in the SSL protocol protect the integrity of transmitted data.
- h. T F A system composed of many different kinds of computing platforms and operating systems is more resilient to viruses and worms than one composed of identical machines running identical operating systems.
- i. T F The principle of complete mediation suggests that using a secure virtual machine (like the JVM) is an attractive way to provide security against buggy or malicious code.
- j. T F The trusted computing base is made up of those components that have been certified not to fail.
- k. T F Polymorphic viruses hide disguise their signatures by modifying system resources like the boot sector.
- l. T F For cryptosystems like DES, some keys are more secure than others.
- m. T F Sequence numbers, timestamps, and nonces are used to prevent replay attacks.

Name: _____

3

- n. T F Most recent worms and macro viruses have caused the most damage by corrupting the integrity of sensitive system files.

2. (16 points) Projects 3 and 4

Answer each of the following questions using at most three or four sentences.

- a. (8 points) Describe how you could modify the implementation of the `BankServer` and `Teller` from Project 4 to establish a covert channel that could leak information about a teller's password to a customer using the `ATMClient`. Also, briefly explain one strategy to prevent such an insider attack.

Modify the `Teller`'s change password functionality to piggyback an encoding of the password on the messages exchanged with the bank server. The bank server can then encode the password in its replies to the ATM.

Audit the code, monitor the execution for strange behavior.

- b. (8 points) Describe denial-of-service attacks against two distinct resources used by the Bank Server program of Projects 3 and 4. Indicate what resource is vulnerable in each attack.

i. resource & attack :

ii. resource & attack :

Network bandwidth: flood of session init messages.

Disk space: intentionally cause the log to write errors to the disk, exhausting disk space.

CPU: intentionally engage in many partially correct runs of the protocol to cause a lot of cryptographic processing. (This one is harder to achieve.)

3. (16 points) System Security

Answer each of the following questions using at most three or four sentences.

- a. (8 points) Give two reasons, other than buffer-overflow attacks, why a network with both a firewall and virus-detection software may still be insecure. Assume that both the firewall and the virus scanner are configured correctly with up-to-date information and that all network connections go through the firewall.

i. vulnerability 1:

ii. vulnerability 2:

1. Mobile code. 2. Previously unseen viruses or worms. 3. Insider attacks, 4. Buffer overflow errors, etc., etc.

- b. (8 points) Describe one way a consumer of software without access to the source code may protect against buffer overrun attacks. Describe a different way that a software implementor may prevent buffer overrun attacks.

i. consumer:

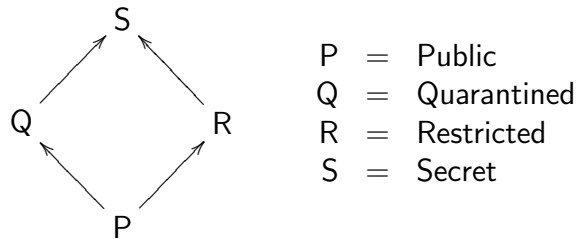
ii. implementor:

Consumer: "Sandbox" the program: use a utility that inserts appropriate array-bounds checks.

Vendor: Use a type-safe language like Java.

4. (10 points) Multilevel Security

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



Suppose that some piece of software takes four *input* variables labeled as follows:

Input	Label
p	P
q	Q
r	R
s	S

The software produces outputs v , w , x , y , and z via the following program:

```

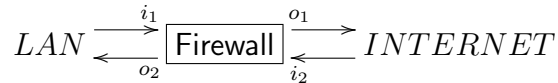
 $x = p;$ 
 $y = q;$ 
 $v = s + r;$ 
if ( $y > 0$ ) {
   $z = r;$ 
} else {
   $w = x;$ 
}
  
```

Indicate the *lowest* security clearance sufficient to securely read the contents of each output variable. Your answer for each entry should be one of P, Q, R, or S.

Output	Clearance Level
v	S
w	Q
x	P
y	Q
z	S

5. (14 Points) Firewall Security

Suppose we place a filtering firewall between a LAN and the Internet as shown in the figure below, where the labels on the arrows indicate the physical ports on the firewall router. The LAN hosts inside the firewall have IP addresses described by the regular expression 240.155.16.*.



The filter table for port i_1 is:

Line	Action	Source	Port	Dest.	Port
1.	Allow	240.155.16.*	*	*	*
2.	Block	*	*	*	*

The filter table for port i_2 is:

Line	Action	Source	Port	Dest.	Port
1.	Block	158.214.5.*	80	240.155.16.*	*
2.	Allow	*	80	240.155.16.*	*
3.	Allow	*	*	240.144.16.3	25
4.	Block	*	*	*	*

The filter tables for ports o_1 and o_2 permit *all* traffic.

- a. (4 points) For each of the following packet descriptions arriving at port i_2 of the firewall, fill in which line of i_2 's filter table will be used to determine whether the packet is Blocked or Allowed. Each answer should be a single number 1, 2, 3, or 4.

Source	Port	Dest.	Port	ANSWER
158.214.5.12	2400	240.155.16.3	23	4
128.7.244.129	80	240.155.16.3	6002	2
128.7.244.129	25	240.155.16.3	23	3
158.214.5.12	80	240.144.16.3	25	1

- b. (5 points) Give a one or two sentence description of the benefit of configuring the filter table on port i_1 as above.

This filtering prevents any machine inside the LAN from spoofing its address on the way out. This firewall helps prevent DoS attacks from being launched within the LAN against machines outside the LAN.

- c. (5 points) Recall that HTTP servers run at port 80. Does the firewall permit a LAN host to establish an HTTP connection with a web server with IP address 158.214.5.12? Explain your answer in one sentence.

No. Port i_2 's filter in line 1 rejects the SYN/ACK of the TCP connection.

6. (5 Points) Security Properties

For each of the following scenarios indicate which of Confidentiality, Integrity, and Availability are most likely to be directly compromised. There may be zero or more correct answers.

- a. C I A Compromised RSA private key.
- b. C I A An attacker exploits a buffer overflow in a web browser.
- c. C I A A fiberoptic cable is accidentally cut by back hoe.
- d. C I A A malicious user runs their Ethernet card in promiscuous mode.
- e. C I A An insider perpetrates a Salami attack.

7. (5 points) RSA Cryptosystems

Recall that in the RSA public key cryptosystem the public key is of the form $K = (e, n)$ and the corresponding private key is of the form $k = (d, n)$ where $e \equiv d^{-1} \pmod{\varphi(n)}$. If m is a message encoded as an integer, the encryption and decryption functions are given by:

$$\begin{aligned} K\{m\} &= E(m, (e, n)) = m^e \pmod n \\ k\{m\} &= D(c, (d, n)) = c^d \pmod n \end{aligned}$$

Show that RSA encryption under two keys k_1 and k_2 is *commutative* if the two keys use the same modulus n . That is, prove that $K_1\{K_2\{m\}\} = K_2\{K_1\{m\}\}$.

$$\begin{aligned} K_1\{K_2\{m\}\} &= E(E(m, (e_2, n)), (e_1, n)) \\ &= (m^{e_2} \pmod n)^{e_1} \pmod n \\ &= (m^{e_2})^{e_1} \pmod n \\ &= m^{(e_2 \times e_1)} \pmod n \\ &= m^{(e_1 \times e_2)} \pmod n \\ &= (m^{e_1})^{e_2} \pmod n \\ &= (m^{e_1} \pmod n)^{e_2} \pmod n \\ &= E(E(m, (e_1, n)), (e_2, n)) \\ &= K_2\{K_1\{m\}\} \end{aligned}$$

8. (20 points) Cryptographic Protocols

Some useful protocols make use of commutative encryption. One such protocol is for *secure, remote coin flipping*. Alice and Bart wish to agree upon a random bit (coin toss), but they can communicate only over the network. Furthermore, neither one trusts the other not to cheat and rig the outcome—as long as at least one of them plays by the rules the bit agreed upon should be truly random. Here's how it works:

1. Alice and Bart each generate a *fresh* RSA key pair with keys K_A , k_A , K_B , and k_B using the same modulus.
2. Alice generates two fresh nonces n_{heads} and n_{tails} and encrypts two messages $m_1 = K_A\{\text{"heads"}, n_{heads}\}$ and $m_2 = K_A\{\text{"tails"}, n_{tails}\}$. She sends both m_1 and m_2 to Bart in a random order.
3. Bart, who cannot read either message, picks one of m_1 and m_2 at random; call Bart's choice m . Bart sends the message $K_B\{m\}$ back to Alice.
4. Alice decrypts it with her private key and sends the result $k_A\{K_B\{m\}\}$ back to Bart.
5. Bart computes $k_B\{k_A\{K_B\{m\}\}\}$ to reveal the coin flip; the random outcome is either "heads", n_{heads} or "tails", n_{tails} . He sends the outcome and keys K_B and k_B to Alice.
6. Alice verifies the nonce and sends the keys K_A and k_A to Bart.

- a. (4 points) Using the result from Question 7, show that when Bart computes the value $k_B\{k_A\{K_B\{m\}\}\}$ in step 5, the result is one of "heads", n_{heads} or "tails", n_{tails} .

$$\begin{aligned}
 k_B\{k_A\{K_B\{m\}\}\} &= k_B\{k_A\{K_B\{K_A\{\text{"s"}, n_s\}\}\}\} && \text{(where } s = \text{heads or } s = \text{tails)} \\
 &= k_B\{k_A\{K_A\{K_B\{\text{"s"}, n_s\}\}\}\} && \text{by commutativity} \\
 &= k_B\{K_B\{\text{"s"}, n_s\}\} && \text{by inverse keys of RSA} \\
 &= \text{"s"}, n_s && \text{by inverse keys of RSA}
 \end{aligned}$$

- b. (8 points) Suppose Alice tries to cheat by sending Bart $m'_2 = K_A\{\text{"heads"}, n_{tails}\}$ instead of message m_2 in step 2—so Bart receives two “heads” messages. How does this protocol allow Bart to detect her cheating?

Bart can store the messages m_1 and m_2 received in step 2. After step 6, Bart can use Alice's private key k_A to check that one of m_1 and m_2 contains the “heads” string and the other contains the “tails” string.

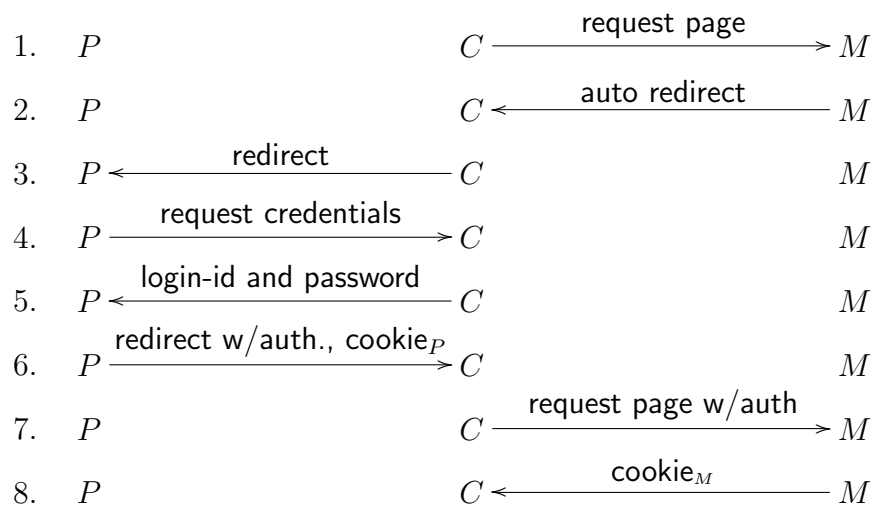
- c. (8 points) Suppose that Bart tries to cheat by lying about the result when he sends Alice the message in step 5. How does this protocol allow Alice to detect his cheating? If Bart sends “heads”, n_{tails} (or vice versa), Alice can detect this by checking the nonce. If Bart sends a bogus nonce, she can catch that too. Note that Bart never has access to both nonces—he only sends one message back to Alice in step 3 for her to decrypt.

9. (35 points) Security Analysis

In this problem, we consider the consequences of *single sign-on* for the Web. Microsoft's Passport is a protocol that enables users to sign onto many different merchants' web pages by authenticating themselves only once to a common server.¹

In the Passport model, there are three principles: (1) the client C at a web browser (usually a consumer who has previously registered with the Passport service), (2) the merchant M (a store or collection of stores wishing to market to the consumer), and (3) the Passport login server P .

Passport's interaction with a user begins when a client, visiting a merchant site, needs to authenticate (to provide some personal information or make a purchase). The merchant web server redirects the customer's browser to a well-known Passport server. The Passport server presents the user with a login page over an SSL connection. The user logs into the Passport server and the Passport server redirects the user back to the end server. Authentication information is included in the redirect message in the query string. This information is encrypted using triple DES with a key previously established between Passport and the merchant server. The end server then sets an encrypted cookie ($cookie_M$) in the client's browser. These steps are illustrated in below:



The idea is that when a user returns to the merchant site, IBM.com for example, then $cookie_M$ is returned as well. The site can decrypt the cookie and verify that the user is already authenticated. The Passport server also sets a cookie in step 6. Thus, if a user visits another site, say dell.com, when the browser is redirected to the Passport server, the user is no longer presented with a login screen because the previous Passport cookie is used. If the Passport cookie contains valid credentials, the client is redirected back to the merchant server without user intervention.

¹Parts of this problem were derived from the paper *Risks of the Passport Single Signon Protocol* by David P. Kormann and Aviel D. Rubin, published in *Computer Networks*, Elsevier Science Press, volume 33, pages 51-58, 2000.

- a. (8 points) If you designed this system, what information would you store as credentials in cookie_P ? Briefly, in one or two sentences, justify your choices.
At least the userid, a (hash of) their passport password, and a timeout.
- b. (12 points) If cookie_M is intercepted by an attacker, the attacker can replay it to web site M to log in as C . Suggest a modification to the protocol that eliminates this attack.
Do what Kerberos does: include as part of cookie_M a shared key K_{CM} , which can be generated by P and transmitted to C over the SSL. Have C and M do mutual challenge-response style authentication using this key.
- c. (15 points) How does the Passport approach (if fixed as in part b) change the security concerns of web-based commerce? Discuss what vulnerabilities this strategy fixes and any new vulnerabilities it introduces. (Use the back of the page if you need more space.)