

CS 378 - Network Security and Privacy  
Fall 2007

FINAL

December 13, 2007

DO NOT OPEN UNTIL INSTRUCTED

YOUR NAME: \_\_\_\_\_

**Collaboration policy**

**No collaboration** is permitted on this midterm. Any cheating (*e.g.*, submitting another person's work as your own, or permitting your work to be copied) will automatically result in a failing grade. The Computer Sciences department code of conduct can be found at <http://www.cs.utexas.edu/users/ear/CodeOfConduct.html>

## Final (125 points)

### Problem 1 (21 points)

Circle only one of the choices (3 points each).

1. **TRUE FALSE** Polymorphic viruses and worms disguise their signatures by modifying system files and utilities.
2. **TRUE FALSE** It is impossible to create an anomaly detector which produces no false positives.
3. **TRUE FALSE** Recall that an RSA modulus  $n$  is a product of two large primes. If someone discovers an efficient algorithm for computing the greatest common divisor of two numbers, then breaking RSA will become feasible.
4. **TRUE FALSE** Plain RSA encryption is insecure against the chosen-plaintext attack.
5. **TRUE FALSE** Using the Diffie-Hellman protocol to establish session keys ensures forward secrecy.
6. **TRUE FALSE** To take advantage of the authentication and confidentiality services provided by IPsec, applications such as Web browsers and FTP clients must be modified accordingly.
7. **TRUE FALSE** Sending messages over an established Tor circuit does not require any public-key operations.

### Problem 2

Alice wishes to authenticate to her bank's website using her Social Security number  $ssn$  by sending a single message which does not reveal  $ssn$  to an eavesdropper, but can be used to verify that Alice indeed knows  $ssn$ .

#### Problem 2a (5 points)

Is sending a hash  $SHA1(ssn)$  or plain RSA encryption  $(ssn)^e \bmod n$ , where  $(n, e)$  is the bank's public key a secure solution? Explain.

**Problem 2b (5 points)**

Can you think of a better way to authenticate Alice to the bank using  $ssn$ , which does not involve exchanging multiple messages between Alice and the bank?

**Problem 3 (6 points)**

The Molvanian Institute of Cryptology proposes the following method for protecting integrity of messages encrypted with a stream cipher. Split the message in two halves, and compute the parity bit for each half. (A parity bit for a bitstring  $b_1, \dots, b_n$  is simply  $b_1 \oplus \dots \oplus b_n$ , where  $\oplus$  is the exclusive-OR operation.) Append the two bits to the message, and encrypt the result.

Is this a good message integrity code? Explain.

**Problem 4**

Suppose the system administrator places a packet-filtering firewall between a LAN (containing all addresses in the  $128.83.139.*$  range) and the Internet.

The rules for the interface from the Internet to the LAN look like this:

| Rule | Action | Source      | Port | Destination   | Port |
|------|--------|-------------|------|---------------|------|
| 1.   | Block  | 129.15.78.* | 80   | 128.83.139.*  | *    |
| 2.   | Allow  | *           | 80   | 128.83.139.*  | *    |
| 3.   | Allow  | *           | *    | 128.83.139.10 | 25   |

**Problem 4a (5 points)**

Do the firewall rules given above permit a host within the LAN to establish an HTTP connection to port 80 at a Web server whose IP address is 129.15.78.10? Explain.

**Problem 4b (8 points)**

For each of the packets arriving from the Internet and matching the following descriptions, write down **which rule** of the above table will be used to determine the firewall action, *i.e.*, whether to block or allow the packet. If no rule applies, write down the default action that the firewall will take.

| Source      | Port | Destination   | Port | ANSWER |
|-------------|------|---------------|------|--------|
| 129.15.78.2 | 4967 | 128.83.139.10 | 22   |        |
| 121.12.12.9 | 80   | 128.83.139.10 | 9018 |        |
| 129.15.78.2 | 25   | 128.83.139.10 | 25   |        |
| 129.15.78.2 | 80   | 128.83.139.10 | 25   |        |

**Problem 4c (5 points)**

Suppose you do not want to restrict any outbound connections from the LAN, but still prevent IP spoofing from within the LAN. Write a single firewall rule for the interface from the LAN to the Internet that would achieve this:

| Action | Source | Port | Destination | Port |
|--------|--------|------|-------------|------|
|        |        |      |             |      |

### Problem 5 (12 points)

Molvanian Security Associates released a new intrusion detection system for detecting malicious port scans and TCP connections with spoofed source addresses. It boasts an impressive accuracy rate, reflected in the following table:

| Type of connection | How this connection is classified |             |        |
|--------------------|-----------------------------------|-------------|--------|
|                    | Port scan                         | IP spoofing | Normal |
| Port scan          | 85%                               | 5%          | 10%    |
| IP spoofing        | 5%                                | 90%         | 5%     |
| Normal             | 5%                                | 5%          | 90%    |

For example, when the IDS observes a malicious port scan, it correctly classifies it as a malicious port scan with probability 85%, misclassifies it as an IP spoofing attack with probability 5%, and misclassifies it as a normal connection with probability 10%.

For the purposes of this problem, assume that port scans are 3% of all connections, and that IP spoofing attacks are 1% of all connections, while 96% of traffic consists of normal connections. Also assume that a connection cannot be *both* a port scan and an IP spoofing attack at the same time.

When the IDS announces that it detected a port scan, what is the probability that the connection is, in fact, normal? Give your calculations.

### Problem 6 (8 points)

Give two reasons, other than buffer-overflow attacks, why a network protected by both a properly configured firewall and an up-to-date virus detection scanner can be vulnerable to a network attack.

### Problem 7

*Key confirmation* is an important concept in key establishment protocols. When the communicating parties (call them Alice and Bob) believe that they already share a symmetric key, it is useful to confirm that Alice's key  $K_A$  is equal to Bob's key  $K_B$ .

Molvanian Institute of Cryptology suggests the following key confirmation method. Alice generates a fresh random string  $r$  of the same length as the key, and sends  $r \oplus K_A$  to Bob, where  $\oplus$  is the bitwise XOR operation. Bob receives message  $m$ , and sends  $m' = m \oplus K_B$  back to Alice. Alice checks whether  $m' = r$  and, if so, concludes that Bob's key is indeed equal to her own.

#### Problem 7a (5 points)

Is this a good key confirmation method? Explain your answer.

**Problem 7b (5 points)**

Suggest a different, secure key confirmation method.

**Problem 8**

Imagine a “white list” scheme for managing public-key certificates. Instead of publishing a list of revoked certificates, the certificate authority gives each certified user a signed list of *valid* certificate numbers. (Assume that certificate numbers are long random values generated by the certificate authority.)

If Alice wants Bob to believe that her public key is valid, she sends her public key, the certificate for that key signed by the certificate authority, and her copy of the white list containing the number of her certificate.

**Problem 8a (5 points)**

Is there any security-related reason for adding expiration dates to the copies of the white list and re-issuing new ones even if no certificates have been revoked?

**Problem 8b (5 points)**

One problem with certificate white lists is that an attacker might look up a certificate number already on the list, and issue a bad certificate with that number. What could the certificate authority publish instead of certificate numbers that would prevent this attack?

**Problem 9**

**Problem 9a (7 points)**

How does AH (Authentication Header) assure integrity of IP packet headers in IPsec?

**Problem 9b (5 points)**

Give an example of an IP header field that is used as an input when the Integrity Check Value is computed in AH, and an example of a header field that is not used. Explain why it is not necessary to verify integrity of the latter field.

### Problem 10 (6 points)

Alice and Bob already share a 128-bit AES key  $K_{AB}$ . Bob generates a fresh session key  $k$ , and wishes to securely communicate it to Alice. In the following protocol, Alice and Bob use their existing shared key  $K_{AB}$  to authenticate each other while they are establishing the new key  $k$ . Assume that nonces  $N_A$  and  $N_B$  are 128 bits each.

$$\begin{aligned} Alice &\rightarrow Bob && A, N_A \\ Alice &\leftarrow Bob && enc_{K_{AB}}(N_A), N_B \\ Alice &\rightarrow Bob && enc_{K_{AB}}(N_B) \\ Alice &\leftarrow Bob && enc_{K_{AB}}(k) \end{aligned}$$

Can evil Charlie exploit this protocol to trick Alice into using his own key  $K_C$ ? Explain.

### Problem 11

For each of the following threats, explain in detail what mechanism is used in SSL/TLS to provide protection, and how it is used. Do not make any assumptions about the specific encryption or signature scheme used by SSL/TLS, as it is supposed to be compatible with multiple schemes.

#### Problem 11a (4 points)

Protection against replay attacks:

**Problem 11b (4 points)**

Protection against man-in-the-middle attacks:

**Problem 11c (4 points)**

Protection against known-plaintext attacks based on pre-computation: