

CSE497B/Spring 2007 - Midterm Exam
Thursday, March 8, 2007 — Professor Trent Jaeger

Please read the instructions and questions carefully. You will be graded for clarity and correctness. You have 75 minutes to complete this exam, so focus on those questions whose subject matter you know well. Students who try to read the papers in order to find an answer will not likely do well. Write legibly and check your answers before handing it in.

Short Answer - some will be one or two words – no more than 3 sentences

1. (3pts) Name one significant difference between a *threat* and a *vulnerability*.
answer: Attackers launch threats where defenders are responsible for vulnerabilities. Several others.
2. (3pts) Suppose a password policy requires a 9-character password that may not end with an integer. How does this impact the password keyspace given that the previous policy called for an unrestricted 8-character password?
answer: The increase in password length increases the password entropy, but not by as much as adding another character since some are prohibited.
3. (3pts) What does a *pluggable authentication module* do?
answer: A pluggable authentication module is a library that implements one or more of the PAM interfaces, such as executing password authentication for the authentication interface and logging for the session interface.
4. (3pts) Why must a symmetric key cryptographic algorithm generate a mapping that is *one-to-one*?
answer: Each unique plaintext must be transformed into a unique ciphertext, otherwise we could not decrypt it properly (would decrypt to one of multiple plaintexts).
5. (3pts) What is an advantage of symmetric key cryptography over public key cryptography? What is a disadvantage?
answer: Adv: Symmetric key algorithms are much faster (bits of encryption per second). Disadv: Getting shared keys distributed securely is much harder in the symmetric case.
6. (3pts) Do we typically apply HMAC to symmetric or asymmetric cryptographic statements to protect integrity and provide authenticity? Why?
answer: Symmetric. If we are using public key crypto, we can generate a signature to guarantee integrity/authenticity without needing to distribute an additional symmetric key. We already have established a symmetric key in the symmetric case.
7. (3pts) An HMAC, $H(k + d)$, when sent with a value d enables a remote party that knows k to detect a modification of the value d . How does the *one-way property* of cryptographic hash functions ensure that this is true?
answer: The *one-way property* prevents an attacker from determining the secret k by reversing the hash computation, so only the communication partners with k could have generated HMAC value.

8. (3pts) The SSH protocol does not require a trusted third party (e.g., Kerberos) or a certificate authority (e.g., PKI). What is the basis for trust in the public key of a SSH server?

answer: SSH is a public key approach, rather than a secret key approach in Kerberos, so we need to only obtain the mapping between the public key and machine reliably. This can be done offline since the number of machines that one depends on is modest, but we typically accept such certs blindly (SSH tells us if the key changes though).

9. (3pts) What is the purpose of a *public key infrastructure* (PKI)?

answer: Provide an infrastructure for securely retrieving a principal's public key certificate (public key to identity binding).

10. (3pts) Why is DNSSEC a good candidate to use a tree-based PKI authentication system?

answer: DNS already uses a tree structure for its root servers and domain servers, so a tree-based PKI can leverage this existing structure of DNS.

11. (3pts) What parts of an IP packet are protected by the IPsec *authentication header* protections? Protected for confidentiality, integrity or both?

answer: Integrity only. Packet data and IP header are protected, excepting mutable fields.

12. (3pts) A worm is launched. If it takes 1 minute for each infected system to find and infect another host, how many hosts would be infected after 10 minutes?

answer: 1024

13. (3pts) What does the following protocol prove to Bob about the party claiming to be Alice? What does it prove to "Alice" about Bob?

| | |
|-------------------|----------------|
| $A \rightarrow B$ | "I'm Alice" |
| $B \rightarrow A$ | $E(K_{AB}, R)$ |
| $A \rightarrow B$ | R |

answer: Bob knows that the responding party (perhaps Alice, but maybe not) really knows the key K_{AB} after the third message (presuming Bob chose a good challenge). If Bob has only shared K_{AB} with Alice, and he trusts that Alice protects this key from theft, then he may assume that Alice is indeed the responding party. Alice learns nothing about whether Bob knows K_{AB} as message 2 could be a replay.

14. (3pts) We choose a RSA private key as a number that is *relatively prime* to $\phi(n)$. How many values are relatively prime to $\phi(n)$? So what is the keyspace of a RSA private key? What is the entropy of a RSA private key?

answer: $\phi(n)$ tells us the number of integers that are relatively-prime to any n . A private key will come from the set of integers that are relatively prime to $\phi(n)$ which is $\phi(\phi(n))$. The number of possible private keys is the keyspace. The entropy $\log_2 \phi(\phi(n))$.

Long Answer - no more than 2 paragraphs

15. (7pts) What is the purpose of a Kerberos *authenticator*? Which part of the Needham-Schroeder protocol does it implement? How does it satisfy the requirements of those Needham-Schroeder messages?

answer: An *authenticator* enables Bob to prove the freshness of Alice's message by verifying the timestamp is within the current time window (and does not match a previous message). It replaces the 4th and 5th N-S messages. In these messages, Bob generates a nonce and requires that Alice demonstrate her current knowledge of the session key by generating a message that demonstrates her knowledge of the nonce. The timestamp enables this guarantee because only someone with knowledge of the session key could encrypt a current timestamp (if we verify that the message is not a replay).

16. (7pts) Why is Diffie-Hellman susceptible to *man-in-the-middle* attacks? Name one way to prevent such attacks.

answer: Any party could generate a Diffie-Hellman key exchange message as there is no identifying secret in the protocol. With a public key infrastructure, principals could sign the key exchange messages.

17. (7pts) It is said that IPsec *authentication header* (AH) protocol is now subsumed by *encapsulated security payload* (ESP) protocol in tunnel mode. Specify how ESP in tunnel mode achieves the guarantees of AH.

answer: ESP can provide authenticity using HMAC for the packet data (with or without encryption, using "null" encryption). In tunnel mode, the entire IP packet, including the IP header is encapsulated by ESP, so by providing authenticity protection for packet data, ESP in tunnel mode protects the header and packet data as the AH protocol does.

18. (7pts) Suppose there is a proposal for a new message digest algorithm that: (1) breaks a message into 160-bit chunks; (2) applies **xor** to all the chunks to get a 160-bit result; and (3) applies a traditional hash function (SHA-1) to the result. What hash property does this new digest algorithm break and why?

answer: **xor** of different pairs may result in the same value (e.g., $1111 \oplus 1111 == 0000 \oplus 0000$), so this hash variant will not be *collision-free*.

Word Problems - take your time and answer clearly and completely.

19. (10pts) Perform the following RSA key generation steps. Each step must satisfy the requirements for a legitimate RSA key.

(a) (2pts) Suppose n is 55. Find suitable p and q values.

(b) (2pts) Compute $\phi(n)$.

(c) (3pts) If $d = 7$, compute a valid value for e and specify the public and private keys.

(d) (2pts) If Bob uses the same n for his key pair, and his public key is 3, what is his private key?

(e) (1pt) Is it a good idea for multiple users to have public key pairs based on the same n ?

answer: (a) $p = 5$ and $q = 11$.

(b) $\phi(n) = (p - 1)(q - 1) = 4 * 10 = 40$.

(c) $d \text{ mod } \phi(n) = 7 * e \text{ mod } 40$; $7e \text{ mod } 40 = 1$ when $e = 23$. Private key is $\{7, 55\}$ and public key is $\{3, 55\}$.

(d) $d * 3 \text{ mod } \phi(n) = 3d \text{ mod } 40$; $3d \text{ mod } 40 = 1$ when $d = 17$. Private key is $\{17, 55\}$.

(e) No way.

20. (10pts) Design *web cookie* that stores the name of a user N and IP address of her computer IP that meets the specified requirements below. Your server has a symmetric key K and a public key pair K^+ and K^- .

(a) (2pts) Using the symmetric key, design a cookie where the server can verify that it generated the cookie and its integrity is intact.

(b) (2pts) Using the symmetric key, design a cookie where the confidentiality of the information is protected and the server can verify that it generated the cookie and its integrity is intact.

(c) (2pts) Using public key cryptography, design a cookie where the server can verify that it generated the cookie and its integrity is intact.

(d) (2pts) Using public key cryptography, design a cookie where the confidentiality of the information is protected and the server can verify that it generated the cookie and its integrity is intact.

| FW2 | | | | |
|----------|----------|-----------|-----------|-------------|
| Src Addr | Src Port | Dest Addr | Dest Port | Accept/Deny |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

answer:

| FW1 | | | | |
|--------------|----------|--------------|-----------|-------------|
| Src Addr | Src Port | Dest Addr | Dest Port | Accept/Deny |
| 12.*.*.* | * | 128.168.11.2 | 80 | A |
| 11.14.* | * | 128.168.11.2 | 80 | A |
| 128.168.11.2 | 80 | 12.*.*.* | * | A |
| 128.168.11.2 | 80 | 11.14.* | * | A |
| 11.14.*.* | * | 128.168.11.4 | 22 | A |
| 128.168.11.4 | 22 | 11.14.*.* | * | A |
| * | * | * | * | D |

| FW2 | | | | |
|--------------|----------|--------------|-----------|-------------|
| Src Addr | Src Port | Dest Addr | Dest Port | Accept/Deny |
| 128.168.11.* | * | 128.168.12.* | 21 | A |
| 128.168.12.* | 21 | 128.168.11.* | * | A |
| 128.168.11.* | 25 | 128.168.12.* | * | A |
| 128.168.12.* | * | 128.168.11.* | 25 | A |
| * | * | * | * | D |