**Student name:**

**Last 4 digits of student ID:**

**Test 2**

**1) This problem deals with properties that a *cryptographic hash function h(x)* must satisfy. Write the letter of the definition on the right next to the corresponding property on the left.**

| | |
|---|---|
| ___ Compression | **A.** There is no feasible way not invert the hash value. |
| ___ Efficiency | **B.** We cannot find two inputs that hash to the same value. |
| ___ One-way | **C.** Computing a hash value must not take too much effort. |
| ___ Weak collision resistance | **D.** For any size input, the hash value is small (fixed in practice). |
| ___ Strong collision resistance | **E.** We cannot modify a message without changing its hash value. |

**2) This problem deals with categorization of digital watermarks. Write the letter of the definition on the right next to the corresponding category on the left.**

| | |
|---|---|
| ___ Invisible | **A.** These are supposed to be destroyed or damaged if tampering occurs. |
| ___ Visible | **B.** These are designed to remain readable even if attacked or tampered with. |
| ___ Robust | **C.** These are supposed to be observed, such as TOP SECRET on a document. |
| ___ Fragile | **D.** These are not supposed to be readable or perceptible in the media. |

**3) Recall that for message integrity we can compute a MAC where the MAC is the last block of a block cipher running in CBC mode (CBC residue). However, with an HMAC (hashed MAC), we use the hash value of the message to provide integrity.**

**Explain why when computing an HMAC, it is necessary to thoroughly mix a key K into the hash.**

*Hint: Explain why Alice should send “M, h(M,K)” to Bob instead of “M, h(M)”.*

**4) This problem deals with CRC checksums and finding collisions. Recall that CRC has mistakenly been used where a cryptographic hash function should have been used as it is easy to find CRC collisions.**

**4a) Compute the CRC checksum for the following Divisor and Dividend (message).**

*Hint: First, Append n-1 zeros to the dividend where n is the count of bits in the divisor.*
*Hint: Second, perform the long division.*

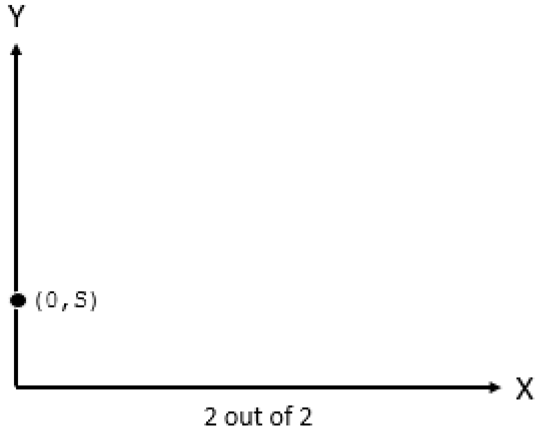
**Divisor: 10011**
**Dividend: 10101011**

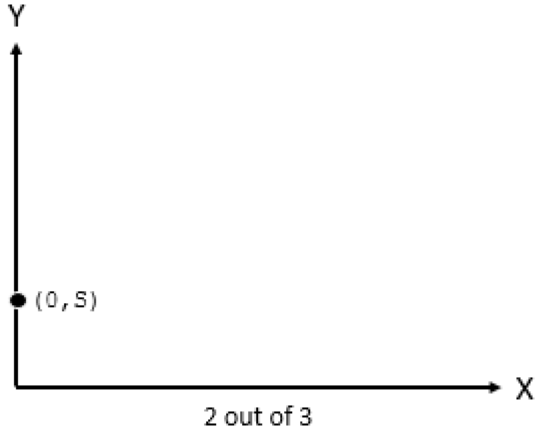
**4b) Find a CRC collision for the Divisor and Dividend (message) from *4a* above.**

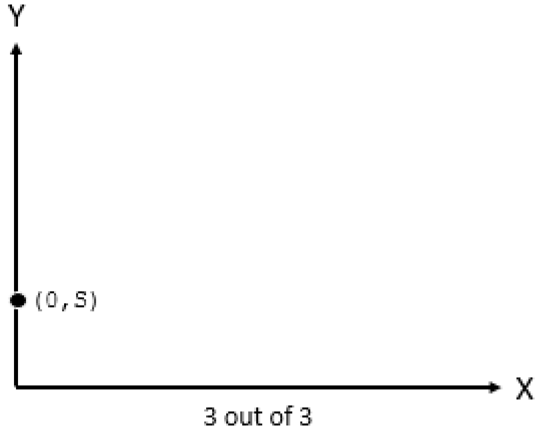
*Hint: First, XOR the divisor with the dividend starting at some position to produce a new dividend (message).*
*Hint: Then append n-1 zeros to the dividend where n is the count of bits in the divisor.*
*Hint: Last, perform the long division.*

**5) This problem deals with secret sharing. Recall that Alice and Bob can share a secret S in such a way that neither Bob nor Alice (nor anyone else) can learn S if acting alone. Of course Alice and Bob can easily learn S if acting together.**

**Illustrate the "2 out of 2", "2 out of 3", and "3 out of 3" secret sharing schemes where the secret S=(0,S) is shared between Alice, Bob, and/or Charlie. Denote each point P=($X_n$,$Y_n$) where P is 'A' for Alice, 'B' for Bob, or 'C' for Charlie ( e.g., A=($X_1$,$Y_1$) ).**







**6) This problem deals with authenticating humans to machines. Write the letter of the definition on the right next to the corresponding category on the left.**

| | |
|---|---|
| ___ Something you know | **A.** Thumbprint (biometrics). |
| ___ Something you have | **B.** A password or PIN. |
| ___ Something you are | **C.** ATM card or a smartcard. |

**7) Check all that apply to password files. We denote hashing a password with salt as *h(password, salt*), and without salt as *h(password)*. Recall that salt is random, and will differ from one password file to another even for the same password.**

[ ] A typical salted password file entry has 3 values: userid, h(password, salt), salt.

[ ] A typical unsalted password file entry has 2 values: userid, h(password)

[ ] Hashing passwords prevents Trudy from obtaining actual passwords if she gets the password file.

[ ] Encrypting the password file is a secure alternative to storing password hashes.

[ ] Trudy can conduct a forward search attack if she knows a password hash value.

[ ] Salting password hashes prevents Trudy from reusing a dictionary of password hashes.

[ ] The salt value is public and therefore available to Trudy.

**8) This problem deals with the properties that an ideal biometric should satisfy. Write the letter of the definition on the right next to the corresponding property on the left.**

| | |
|---|---|
| ___ Universal | **A.** Can be successfully used in the real world, not just in the laboratory. |
| ___ Distinguishing | **B.** The physical characteristic should be easy to collect without harm to the subject. |
| ___ Permanent | **C.** Ideally the physical characteristic being measured should never change. |
| ___ Collectable | **D.** A biometric should apply to virtually everyone. |
| ___ Reliable, robust, and user-friendly | **E.** A biometric should distinguish with virtual certainty (low error rate). |

**9) This problem deals with the function of biometric systems.**

**9a) Define the two phases of a biometric system (enrollment and recognition).**

**9b) Define two types of errors that can occur in biometric recognition and their corresponding rates.**

**10) This problem deals with the Iris Scan biometric. Recall that two iris codes are compared based on the Hamming distance between the codes. A perfect match between two iris codes yields a distance of 0, where *d(x, y)* = 0. Of course, perfect matches are not expected and therefore we say *x* matches *y* if *d(x, y)* < 0.32.**

**The following 16-bit iris codes were obtained during the enrollment phase:**

| Name | Iris code (binary) | Iris code (hex) |
|---|---|---|
| Alice | `1010101010101010` | `0xAAAA` |
| Bob | `0101010101010101` | `0x5555` |
| Charlie | `1100110011001100` | `0xCCCC` |

**The following 16-bit iris codes were obtained during the recognition phase. Identify who from the table above is a likely match for the iris codes in the table below.**

| Iris code (binary) | d(x, y) | Name |
|---|---|---|
| `1011101110111010` | | |
| `1101110111011100` | | |
| `0100010001000101` | | |

**11) Check all that apply to ACLs and Capabilities (C-lists).**

[ ] ACLs and Capabilities are two different ways to partition an access control matrix.

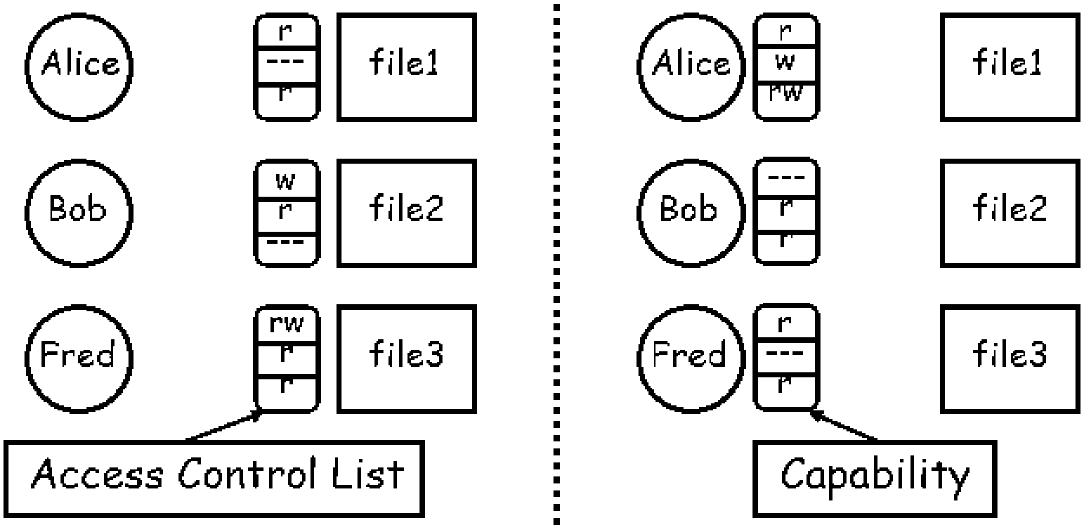
[ ] Capabilities store information by subject (row) whereas ACLs store information by object (column).

[ ] ACLs and Capabilities optimize the performance of authentication operations.

[ ] Capabilities are more difficult to implement than ACLs.

[ ] Capabilities can prevent the confused deputy problem by delegating authority to called programs.

**12) Draw arrows to illustrate how subjects and objects are associated in ACLs and Capabilities.**



**13) Design a simple Covert Channel and estimate its bandwidth (e.g., 1bit/sec). Recall that a covert channel requires that at least two people/machines can access a shared resource and observe changes in its state at a regular interval or time T.**

**14) This problem deals with inference control. Write the letter(s) of the characteristics on the right next to the corresponding techniques on the left. Use all the letters at least once.**

| | |
|---|---|
| ______ Query set size control | **A.** Ensure that no individual or group contributes a majority of the data, thereby preventing their identification. |
| ______ N-respondent, k% dominance rule | **B.** Prevents or distorts important research where the number of respondents is small (rare diseases). |
| ______ Randomization | **C.** May be problematic if precise answers are needed and may even produce false data. |
| | **D.** Ensure that exact values cannot be extracted while preserving trends in the data, allowing it to be analyzed. |

**15) It can be said that some types of security, if easily compromised, are worse than no security at all. (paraphrase of Ross Anderson's quote).**

**15a) Does this statement hold for limited inference control? Why or why not?**

**15b) Does this statement hold for weak encryption? Why or why not?**

**15c) Does this statement hold for limited covert channel reduction? Why or why not?**

**16) This problem deals with authentication using symmetric key cryptography. Illustrate how Alice and Bob can achieve mutual authentication if they both possess a shared key $K_{AB}$. Give your answer in the form of an exchange of messages between Alice and Bob.**

**17) This problem deals with authentication using public key cryptography. Illustrate how Alice and Bob can achieve mutual authentication using public key crypto. Give your answer in the form of an exchange of messages between Alice and Bob.**

**18) This problem deals with establishing a session key $K_S$ with which Alice and Bob will encrypt data they wish to send to each other. Recall that session keys are symmetric keys that are established to either avoid expensive public key operations or to avoid overexposing the shared key $K_{AB}$.**

**Illustrate how Alice and Bob can achieve mutual authentication and establish a session key $K_S$. You may use either a shared symmetric key (prob. 16) or public key crypto (prob. 17) for mutual authentication.**

**19) Recall that Perfect Forward Secrecy prevents Trudy from being able to capture and later decrypt messages exchanged between Alice and Bob if she can steal the symmetric key $K_{AB}$. Explain how Alice and Bob can establish a session key $K_S$ without actually sending $K_S$ in a message.**

*Hint: If Alice sends E($K_S$, $K_{AB}$) to Bob, Trudy can get $K_S$ if she can steal $K_{AB}$ from Alice or Bob's computer.*

**20) Check all that apply to Timestamps.**

[ ] A timestamp T can be used in place of nonce value R, so long as T is a current time.

[ ] One advantage of timestamps is that we don’t waste messages exchanging nonce values.

[ ] One disadvantage to timestamps is that authentication requires system clocks to be synchronized.

[ ] As a general rule, protocols that use timestamps are more secure than those that use nonce values.

[ ] *Clock skew* makes it possible for systems to communicate if their clocks are not perfectly synchronized.

[ ] Even though timestamps prevent replay; Trudy can replay messages if she acts within the clock skew.

**End of Document**