CS 6324: Information Security **Spring 2021**

Homework #3 (Optional)

Due date & time: 11:59pm CST on May 14, 2021. Submit to eLearning by the due time.

Late Policy: The late policy of the course is not applicable to this optional homework (due to the University's grading deadline). Late submission will result in zero point.

Additional Instructions: The submitted homework must be typed. Using LaTeX is recommended, but not required.

Problem 1 (15 pts) Identity-Based Encryption (IBE) is a type of public-key encryption in which the public key of a user is some unique information about the identity of the user (e.g. a user's email address). Identity-based systems allow any party to generate a public key from a known identity value such as an ASCII string. A trusted third party, called the Private Key Generator (PKG), generates the corresponding private keys. To operate, the PKG first publishes a master public key, and retains the corresponding master private key (referred to as master key). Given the master public key, any party can compute a public key corresponding to the identity ID by combining the master public key with the identity value. To obtain the private key corresponding to an identity ID string, the party authorized to use the identity ID contacts the PKG, which uses the master private key to generate the private key for identity ID.

Consider Kerberos, Public key infrastructure, and IBE systems. Each of the three systems can enable two users who did not know each other to communicate securely. All of them use a Trusted Third Party: the Key Distribution Center (KDC) in Kerberos, Certificate Authorities (CAs) in public key certificates, and PKGs in IBE systems.

Compare these three systems along the following dimensions (1) System availability and communication overhead for large number of users; (2) The ease of having multiple TTPs in an Internet-scale system (consider how users under different TTPs can communicate). (3) The degree of trust one has to place in the TTPs for confidentiality. (4) For each of the three systems, identify a scenario where the system is the most appropriate.

Problem 2 (18 pts) BLP is designed to enable one to formally show that a computer system can securely process classified information. This problem asks you to assess how well this is done.

- (1) State the BLP notion of security. Start with "We say a system is BLP-secure when $\cdots$. (2) Identify three reasons why a system whose specification is shown to satisfy the BLP notion of security can still have illegal information flows. (3) For each reason identified above, use one sentence to explain whether/how they can be addressed (either by enhancing the BLP model, or by identifying other components)?
- (4) State the Basic Security Theorem. (5) Explain why the BST does not provide an easy tool for verifying whether a system satisfies the BLP notion of security.

- (6) In terms of modeling a system, how do BLP and the Goguen-Meseguer non-interference model differ? (7) Give an example that Goguen-Meseguer model allows an operation, but the BLP model would prevent. Which of BLP and Goguen-Meseguer better matches the high-level intuition of security?
  (8) Give an example that the BLP model allows an operation, but the Goguen-Meseguer model would prevent. Which of BLP and Goguen-Meseguer better matches the high-level intuition of security?

Problem 3 (12 pts) Assume that x, y, z are variables that take values either 0 or 1. Answer the following questions.

1. Give a deterministic function $f(x, y, z)$ such that $w = f(x, y, z)$ satisfies the following conditions: There **exists no** information flow in the non-deducibility sense between x and w, between y and w, between z and w, between $(x + y)$ and w, between $(x + z)$ and w, and between $(y + z)$ and w. But there **exists** information flow in the non-deducibility sense between $(x + y + z)$ and w.
2. Give a deterministic function $f(x, y, z)$ such that $w = f(x, y, z)$ satisfies the same conditions as above, except that now we require that there **exists** information flow in the non-deducibility sense between $(y + z)$ and w.
3. Prove that there does not exist a deterministic and non-constant function $f(x, y, z)$ (the function returns as least two values) such that there **exists no** information flow in the non-deducibility sense between $(x + y + z)$ and $w = f(x, y, z)$.

Question 4 (15 pts) Explore Biba integrity models. When a low-level subject attempts to write to a high-level object, there are three choices: (W1) Forbid it; (W2) Allow it, but drops the integrity level of object; (W3) Allow it, without changing the object's level. When a high-level subject attempts to read a low-level object, there are three choices; (R1) Forbid it; (R2) Allow it, but drops the integrity level of subject; (R3) Allow it, without dropping the subject's integrity level.

- (a, 6 pts) For each of the six choices, identify which kind of trust (if any) it places on the subject. Also indicate whether the object's integrity level indicate quality or importance.
- (b, 9 pts) Analyze the 9 possible combinations of choosing a W rule and an R rule. For each combination, identify which (if any) Biba policy it corresponds to. If there is no Biba policy associated with it, briefly describe whether you think it is a reasonable combination useful for some situations.

Problem 5 (10 pts) Read the following article.

- Ken Thompson's "Reflections on Trusting Trust" (`thompson.pdf` is attached in eLearning).

Write a brief summary which should include: (1) how the attack described in Thompson's article works; (2) how it can be used to compromise the security of real world systems; (3) what are some ways to defend against the attack; (4) what you have learned.

Problem 6 (30 pts) Read the following two papers:

- Part I.A of the "The Protection of Information in Computer Systems" by Saltzer and Schroeder (`saltzer.pdf` is attached in eLearning).
- "Design Principles" by Barnum and Gegick (`barnum.pdf` is attached in eLearning).

The first paper lists 8 design principles. The second paper includes 12 principles (including 7 from the first paper). There are thus 13 principles from the two papers (12 in the second paper plus "open design" in the first paper). For each of the 13 principles: (1) write your understanding of the principle; (2) if appropriate, give an instance where it should be applied but is not; (3) if available, give an instance where the principle is applied.