Research in Information Security (CSE540)

Mid Semester Examination (Monsoon 2024)

International Institute of Information Technology, Hyderabad

Time: 1 hour and 30 minutes Total Marks: 40

Instructions: Answer ALL questions. This is a closed book and notes examination. Calculator is allowed. No query is allowed during the examination period.

1. (a) Consider the following linear polynomial-based dynamic key management scheme for hierarchical access control with the following phases.

We assume that there are N security classes in the hierarchy which form a set $SC = \{SC_1, SC_2, \ldots, SC_N\}$. $H(\cdot)$ is a secure collision-resistant one-way hash function (for example, SHA-1 hash function). Ω a symmetric key cryptosystem (for example, AES symmetric-key block cipher), and $E_k(\cdot)/D_k(\cdot)$ the symmetric-key encryption/decryption using key k. The scheme makes use of linear polynomials instead of polynomial of higher degree for computational efficiency.

Relationship building phase: In this phase, CA first needs to build the hierarchical structure for controlling access according to the given relationships among the security classes in the hierarchy.

Let $SC_i \in SC$ and $SC_j \in SC$ be two security classes such that the relationship $SC_j \leq SC_i$ hold, that is, SC_i have a higher security clearance than that for SC_j. A legitimate relationship $(SC_i, SC_j) \in R_{i,j}$ between SC_i and SC_j will exist in the hierarchy if SC_i can access SC_j.

Key generation phase: Once the relationship building phase is completed by the CA, the CA can execute the following steps for distributing the secret and public keys to security classes in the given hierarchy:

Step 1. CA first selects a secure collision-resistant one-way hash function $H(\cdot)$ and then a finite field $GF(m)$, where m is either odd prime or prime power. CA also chooses a symmetric key cryptosystem Ω (for example, AES).

Step 2. CA randomly selects its own secret key k_{CA}. After that CA needs to select randomly the secret key sk_i and sub-secret key d_i for each security class SC_i $(1 \leq i \leq N)$ in the hierarchy.

Step 3. Once Step 2 is completed, for each security class SC_i, CA computes the signature $Sign_i$ on sk_i as $Sign_i = H(ID_{CA}||sk_i)$, where ID_{CA} is the identity of the CA. The purpose of signature is used later by the CA and security classes for verification of the secret key sk_i of SC_i. CA declares them as public.

Step 4. For each SC_i such that $(SC_i, SC_j) \in R_{i,j}$, CA then constructs the linear polynomials $f_{i,j}(x) = (x - H(ID_{CA}||Sign_j||d_i)) + sk_j \pmod m$, and declares them publicly.

Step 5. Finally, CA sends d_i to SC_i via a secure channel.

Answer the following questions to this scheme:

(i) Consider the following small poset user hierarchy provided in Figure 1. Compute all the public parameters corresponding to each security class in the scheme to be stored in public domain.

(ii) Explain the key derivation phase related to this scheme in which a security class SC_i can derive the secret key sk_j of its successor SC_j with $(SC_i, SC_j) \in R_{i,j}$.

bad

(b) Explain the "reactive password checking" and "proactive password checking" password selection strategies used in password management.

good

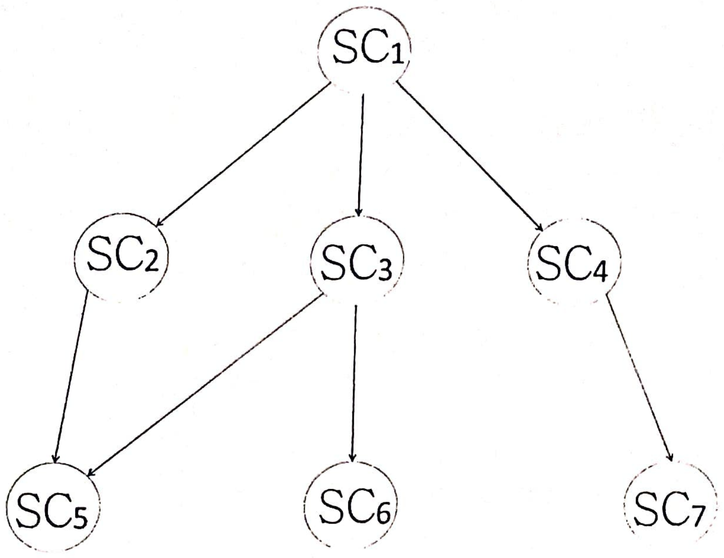

Figure 1: An example of a poset in a user hierarchy

2. (a) Explain the working of end-to-end encryption (EEE) procedure used in encrypting communication channels. What are the advantages and disadvantages of this procedure?

   (b) Recall that we have discussed a random key distribution scheme proposed by Eschenauer and Gligor, in which any two neighbor sensor nodes can establish a secret key (secure link) if they share at least one common key between their key rings pre-loaded during their pre-deployment phase. Note that the network connectivity probability for this scheme is $p_{EG} = 1 - \frac{\binom{M-m}{m}}{\binom{M}{m}} = 1 - \prod_{i=0}^{m-1} \frac{M-m-i}{M-i}$, where M is the key pool size and m the key ring size of a sensor node.
   Consider the improved version of this scheme, where any two neighbor sensor nodes can establish a secret key (secure link) if they share at least q ($q \geq 1$) common keys between their key rings pre-loaded during their pre-deployment phase. We call this scheme as q-composite scheme. Derive the network connectivity probability for the q-composite scheme.

   (c) What is wormhole attack? Briefly explain it with an example.

[3 + 4 + 3 = 10]

3. (a) Distinguish between the proxy signature by partial delegation and delegation by warrant.

   (b) In a TCP SYN flood attack, the victim sees a dis-appropriate number of SYN packets compared to FIN packets. With respect to distributed Denial-of-Service (DDoS) attack, define the "smoothed average" decision variable and then its intrusion detection mechanism.

   (c) Explain various attacks related to AI/ML.

[2 + 5 + 3 = 10]

4. (a) Consider the following password-based user authentication scheme developed for IoT-based WSN security, which consists of the following registration phase, login phase and verification phase.

   Registration phase: This phase is invoked when a user, U_i, wants to register with the WSN. U_i submits his/her identity (ID_i) and password (PW_i) to the GW-node (base station) in a secure manner. Upon receiving the registration request, the GW-node computes $N_i = h(ID_i||PW_i) \oplus h(K)$, where K is a symmetric key known to only GW-node, and "||" is bit-wise concatenation operator. Then the GW-node personalizes a smart card with the parameters $h(\cdot), ID_i, N_i, h(PW_i)$ and x_a, where $h(\cdot)$ is a cryptographically secure hash function. Here, x_a is a secret parameter generated securely by the GW-node and stored in some designated sensor nodes before deploying the nodes in the field, who are responsible to exchange data with users.

Login phase: U_i inserts her/his smart card to a terminal, and keys ID_i and PW_i. The smart card validates ID_i and PW_i with the stored ones in it. If the entered ID_i and PW_i are correct, the smart card performs the following operations:
Step L1. Compute $DID_i = h(ID_i||PWi) \oplus h(x_a||T)$, where T is the current timestamp of U_i's system.
Step L2. Compute $C_i = h(N_i||x_a||T)$. Then send the login request $\langle DID_i, C_i, T\rangle$ to the GW-node via a public channel.

Verification phase: Upon receiving the login request $\langle DID_i, C_i, T\rangle$ at time T^*, the GW-node authenticates U_i by the following steps:
Step V1. Validate T. If $(T^* - T) \leq \Delta T$, the GW- node proceeds to next step, else abort, where ΔT denotes the expected time interval for the transmission delay.
Step V2. Compute $h(ID_i||PW_i)^* = DID_i \oplus h(x_a||T)$ and $C_i^* = h((h(ID_i||PW_i)^* \oplus h(K))||x_a||T)$.
Step V3. If $C_i^* = C_i$, the GW-node accepts the login request; else rejects it.
Step V4. The GW-node now sends a message $\langle DID_i, A_i, T\rangle$ to some nearest sensor node, say, S_n, over a public channel to respond the query/data what U_i is looking for, where $A_i = h(DID_i||S_n||x_a\ ||T)$, and T is the current timestamp of GW-node's system.
Step V5. The sensor node S_n first validates T in similar line of Step V1. Then S_n computes $h(DID_i||S_n||x_a||T)$ and checks whether it is equal to A_i. If these two checks pass correctly then S_n responds to U_i's query.

(i) Verify whether the above scheme resists the user impersonation attack. Provide your argument very clearly.

(ii) Is the above scheme resilient against sensor node capture attack? If not, why? Provide your argument very clearly.

(b) Explain the need for secure data delivery and collection in an Internet of Drones (IoD) environment. In such scenario, discuss how the blockchain technology can be useful.

[(3 + 3) + 4 = 10]

******************* End of Question Paper **********************