정보보호이론 [과제-3]

목차

[과제-3][1] ------------------------------------ 1

[과제-3][2] ------------------------------------ 2

[과제-3][3] ------------------------------------ 3

[과제-3][4] ------------------------------------ 4

[과제-3][1]

[1] 3개의 OTP 표준 (Hash Chain, Counter, Time) 각각에 대한 작동방식을 설명하고 실용적인 측면에서 3개 중에서 어떤 방식이 우수한지에 대한 본인의 의견은?

〈answer〉

(Hash Chain)

1) Prover가 임의의 난수로 Seed값 생성
2) Seed에 연속적으로 암호해시함수를 n번 반복
3) n개의 OTP를 서버에 저장
4) 처음 사용할 때 Prover는 저장된 Password_n값이 아닌 Password_n-1값 사용
5) Verifier는 Password_n-1에 암호해시함수를 적용하여 Password_n값과 일치하는지 확인
6) 확인 후 Verifier는 Password_n-1 저장
7) n개의 Password를 모두 사용한 후에는 갱신

* Prover 이외에는 Password_n ~ Password_0(Seed)의 순서 즉 내림차순의 계산은 불가능하다.

(Counter / Time)

1) verifier가 prover에게 Nonce를 전송한다
2) prover는 verifier에게 받은 Nonce를 대칭키로 encryption하여 전송한다.
3) verifier는 prover에게 받은 response를 대칭키로 decryption하여 Nonce를 확인한다.

Counter - Nonce 대신 사전에 prover와 verifier 사이에 공유된 순번을 사용한다. 따라서 1번 과정을 생략할 수 있다.

Time - Nonce 대신 prover와 verifier 사이에 동기화된 Timestamp를 사용한다. 따라서 1번 과정을 생략할 수 있다. 하지만 시간이 정확히 동기화 되지 않는 경우도 많기 때문에 어느 정도의 시간 간격을 생각하고 적용한다.

(우수한 방식 선택)

저는 Time 방식은 시간의 오차를 해결하는 방법이 있다고 하더라도 오류의 발생가능성이 있다는 생각이 들기에 나머지 방식인 Hash Chain 방식과 Counter 방식이 더 낫다고 생각합니다. 또한 이 둘 중에 하나를 꼽자면 미리 많은 양의 Password를 저장하기 때문에 많은 공간 즉 메모리가 필요한 Hash Chain 방식 보다는 Counter 방식이 더 효율적이라는 생각입니다.

[과제-3][2]

[2] 인증기술 - Reflection Attack에 대해서 설명하고, 이를 무력화 시킬 수 있는 방법을 생각해 보시오.

〈answer〉

(Reflection Attack)

1) 공격자가 Alice로 가장하여 Bob에게 인증 시도
2) Bob은 Alice임을 인증하기 위한 challenge 전송
3) 공격자는 Bob이 보낸 challenge 그대로 Bob에게 전송
   → 이때 Bob은 공격자가 보낸 challenge를 Alice가 보낸 response로 인식

(무력화 시킬 수 있는 방법)

1) Alice가 Bob에게 인증 시도
2) Bob은 Nonce를 대칭키로 encryption하여 전송
3) Alice는 대칭키로 Bob이 전송한 request를 decryption하여
   Nonce를 확인하고 Nonce-1(사전에 약속)을 대칭키로 encryption하여 Bob에게 전송
4) Bob은 대칭키로 Alice에게 받은 response를 decryption하여 Nonce-1 확인 후 인증

[과제-3][3]

[3] 키 관리 기술 - Linear Feedback Shift Register에서 Connection Polynomial이 다음과 같고 Register의 초기값이 공통적으로 1 0 0 1 이라고 가정하자.

$\sigma(x) = 1 + x + x^2 + x^3 + x^4$
$\sigma(x) = 1 + x^3 + x^4$

각각의 Connection Polynomial에 대해서 도출되는 pseudo random bit들의 cycle(주기= 즉, 어떤 비트들이 반복되는지)를 구하시오.

〈answer〉

cycle: $2^4 - 1$의 약수 개

1001
→ 0100 → 0010 → 0001 → 1000 → 1100
→ 1110 → 1111 → 0111 → 1011 → 0101
→ 1010 → 1101 → 0110 → 0011 → 1001

[과제-3][4]

[4] 키 관리 기술 - Secret Sharing에서 Lagrange Interpolation이 어떻게 동작하는지를 웹에서 해당내용을 참조하여 Secret Sharing과 함께 설명하시오.

<answer>

Secret Spliting은 한 가지의 Secret에 대하여 n명의 인원에게 키를 부여하고 n명이 모두 Secret에 접근할 때 Secret을 획득할 수 있는 기법이다. 하지만 이 방법은 한 명이라도 키를 분실하거나 해당 인원이 사망하였을 경우 Secret에 접근할 수 없게 된다. 이에 Secret Spliting의 문제점을 해결할 수 있는 기법이 Secret Sharing 이다.

Secret Sharing은 Secret에 대한 키를 가진 n명의 인원 중 k명만 Secret에 접근하더라도 Secret을 획득할 수 있는 기법이다. Lagrange Interpolation을 활용한 방식이 Secret Sharing의 대표적인 기법이다.

이 기법은 1979년 Shamir에 의해 발견되었다. 이 기법은 크게 초기화 과정, 공유값 분배과정, 비밀키 복원과정으로 나누어진다. 여기서 보안관리자가 존재하는데 보안관리자는 하나의 비밀키로부터 공유값을 생성, 분배, 복원하는 역할을 수행한다.

(초기화 과정)

보안관리자는 Z_p상의 0이 아닌 서로 다른 원소 n개를 선택하고 이를 x_i로 표기한다. i 는 참가자들의 순서(index)를 의미하고 $1 \le i \le n$을 만족한다. 임의의 I에 대해서 보안관리자는 x_i를 i 번째 참가자 P_i에 대응시켜준다.

(공유값 분배과정)

1) 보안관리자가 공유하려는 비밀키(K)는 Z_p상의 임의의 원소(즉, $K \in Z_p$)로 가정한다. 딜러는 Z_p상에서 $k-1$개의 원소들을 선택하고 이를 각각 $a_1, a_2, \dots, a_{k-1}$로 표기한다.

2) 임의의 i $(1 \le i \le n)$에 대해 보안관리자는 다음의 식을 이용해 공유값 $y_i = a(x_i)$값을 계산한다.

$$a(x) = K + \sum_{j=1}^{k-1} a_j x^j \pmod p$$

3) 임의의 i $(1 \le i \le n)$에 대해 보안관리자는 i 번째 참가자 P_i에 대응하는 공유값(y_i)을 분배한다.

(비밀 복원과정)

1) 보안관리자는 n명의 인원 중 k명 또는 그 이상의 인원을 모집한 후 그 인원으로부터 임의의 참가자 P_i가 소유한 공유값 y_i를 수집한다.

2) 수집한 $k(k \leq n)$ 또는 그 이상의 (i, y_i)쌍들과 Lagrange Interpolation을 이용하여 공유값 분배과정에서 사용했던 식을 복원한다. Lagrange Interpolation에서 사용되는 식은 다음과 같다.

$$a(x) = \sum_{j=1}^{k} (y_j \prod_{1 \leq o \leq k, o \neq j} \frac{x - x_o}{x_j - x_o}) \bmod p$$

단, x_j와 x_o는 각각 순번이 j번과 o번째인 인원의 고유값을 의미하고 y_j는 $a(x_j)$에 대응되는 값이며 p는 소수이다.

3) 보안관리자에 의해 복원된 식을 통해 비밀키(K)를 계산하고 비밀키의 복원이 완료된다.

* Lagrange Interpolation

– 개요: Lagrange Interpolation(라그랑주 보간법)이란 서로 다른 $x_1, x_2, ..., x_{n+1}$에 대하여 $n+1$개의 점 $(x_1, y_1), (x_2, y_2), ..., (x_{n+1}, y_{n+1})$이 주어져 있을 때 이 점을 모두 지나는 n차 이하의 다항식을 구하는 공식을 말한다.

– 공식: 서로 다른 $x_1, x_2, ..., x_{n+1}$에 대하여 아래의 다항식

$$p_{i(x)} = \prod_{j \neq i} \frac{x - x_j}{x_j - x_j} = \frac{(x - x_i) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots (x - x_{n+1})}{(x_i - x_1) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_{n+1})}$$

을 라그랑주 기저 다항식이라 한다. 또한

$$p(x) = y_1 p_1(x) + \cdots + y_{n+1} p_{n+1}(x)$$

을 라그랑주 보간 다항식이라 하며, 이 다항식은 점 $(x_1, y_1), (x_2, y_2), ..., (x_{n+1}, y_{n+1})$을 모두 지나는 유일한 n차 이하의 다항식이다.