안전성 증명

https://youtu.be/Af1iar6Ab5o

Cryptography

- Symmetric Encryption scheme (SE = Gen, Enc, Dec)
  - Gen(1^s)
  - Enc(k, m)
  - Dec(k, c)

- Chosen Plaintext Attack (CPA)
  - Have Oracle Enc.

Pseudorandom Ciphertext Experiment

Algorithm 1 Pseudorandom Ciphertext Experiment

procedure $\mathsf{PRC\_EXP}_{\mathsf{A}}^{\mathsf{SE}}(1^s)$
    $k \leftarrow \mathsf{Gen}(1^s)$
    $(m, S) \leftarrow \mathsf{A}_1^{\mathsf{Enc}_k}(1^s)$ ▷ S is internal state information of A
    $b \leftarrow U(\{0,1\})$
    if $b = 1$ **then**
        $c \leftarrow \mathsf{Enc}(k, m)$ ▷ Use the actual encryption
    else
        $c \leftarrow U(\{0,1\}^{|\mathsf{Enc}(k,m)|})$ ▷ Use a random string
    end if
    $b' \leftarrow \mathsf{A}_2^{\mathsf{Enc}_k}(c, S)$
    if $b = b'$ **then**
        return 1 ▷ A guessed correctly
    else
        return 0 ▷ A did not guess correctly
    end if
end procedure

Success advantage

IF

$$\mathbf{Adv}^{\mathsf{PRC}}_{\mathsf{A,SE}}(s) = \left| \frac{1}{2} - \Pr\left[\mathsf{PRC\_EXP}^{\mathsf{SE}}_{\mathsf{A}}(1^s) = 1 \right] \right|,$$

Is negligible for every adversary
Then, SE has pseudorandom ciphertext

Steganography

Hidden Text

$$b_i \in \{0, 1\}$$

$$t_i \leq t_{i+1}$$

t1	t2	...	tn
(b1, t1)	(b2, t2)	...	(bn, tn)

Definition 1

- Stegosystem $\Pi(embed, extract)$
- Security parameter s
- Channel C with two probabilistic algorithms
  - Embedding algorithm
    - Input : $k \in \{0,1\}^{n_k},\ m \in \{0,1\}^*,\ \mathrm{H} \in \{0,1\}^* \times \mathbb{R}$
    - Output : $\mathrm{c} \in \{0,1\}^*$
  - Extracting algorithm
    - Input : $k \in \{0,1\}^{n_k},\ \mathrm{c} \in \{0,1\}^*$
    - Output : $m \in \{0,1\}^*$

H : $channel\ History$

C_{H} : Distribution

Definition 2

- Reliability of a $\Pi\, with\, messages\, of\ length\, n$

$$min_{m\in\{0,1\}^n,\mathrm{H}}\ \Pr[Extract(k, Embed(k, m, \mathrm{H})],$$

- Reliability ?
  - embedding, extracting

Definition 2

- Security (chosen hiddentext attack)
  - $Adversaty\ \mathrm{A}\ can\ access\ to$
    - Channel C through sampling oracle M
    - Additional 2 oracles
      - Π_k : $outputs\ stegotext\ \boldsymbol{c}$
      - O : $outputs\ \boldsymbol{random\ components}\ of\ C_{\mathrm{H}}$

$$\left|\Pr[\mathrm{A}^{M,\Pi_k}(1^s) = 1] - \Pr[\mathrm{A}^{M,O}(1^s) = 1]\right|,$$

$$\boldsymbol{if\ negligible\ for\ any\ A, then\ secure}$$

Assumption

- Indistinguishibility in blockchain B
  - 공격자가 랜덤한 지불들로부터 내용이 담긴 지불을 분리해내는 것
- Assumption
  - Alice
    ${P_k}^A$로 구분, Bob과 $(\lambda, k), \mathrm{N}$ 공유, M_{H}에 따라 샘플링 가능
  - Bob
    ${P_k}^A$를 알고 있음, Alice과 $(\lambda, k), \mathrm{N}$ 공유
  - Adversary
    ${P_k}^A$를 알고 있음, B의 $\mathrm{Read}, \mathrm{Submit}\ Oracles$에 대한 접근 가능
    Distinguish secret communication payments from regular ones

Assumption

- 공격자 A
  - 숨겨진 메시지와 블록체인을 읽을 수 있는 권한 가짐
  - 공격자가 Alice의 Submit을 막거나 Bob의 Read를 막을 수 없음
  - Alice의 키 생성, (λ, k) 공유에 관여할 수 없음
  - 전자서명을 위조할 수 없음

Adversary (A_1, A_2)

- A_1
  - 전체 블록 접근 가능
  - Submit 가능
  - 메시지 m 을 찾아내는 것을 목표
- A_2
  - 메시지 m 이 포함된 지불을 랜덤한 지불로부터 찾아내는 것을 목표

Algorithm 4 Payment Distinguishing Experiment

procedure $\mathsf{PAY\_DIST\_EXP}_{\mathsf{A}}^{\Pi,\mathcal{B}}(1^s)$
    $(s_k, p_k) \leftarrow \mathsf{Gen}_{\Sigma}(1^s)$
    $(m, S) \leftarrow \mathsf{A}_1^{\mathsf{Read,Submit}}(p_k)$ ▷ S is internal state information of the adversary that can be passed to the second stage
    $(\lambda, k) \leftarrow \mathsf{Gen}_{\Pi}(1^s)$
    $b \leftarrow U(\{0,1\})$
    if $b = 1$ **then** ▷ Actual message is sent to the blockchain
        $\mathsf{Embed}((\lambda, k), m, \mathcal{B})$
    else ▷ Random payments are sent to the blockchain
        $n_\lambda \leftarrow |\lambda|$
        $N \leftarrow |\mathsf{Enc}(k, m)| + n_\lambda$ ▷ Enc is the encryption scheme used by Π
        Generate N random addresses a_i for $i \in \{1, 2, \ldots, N\}$
        Simulate Embed to generate payments to a_i
        Submit payments to blockchain one-by-one as Embed does
    end if
    $b' \leftarrow \mathsf{A}_2^{\mathsf{Read,Submit}}(p_k, S)$
    if $b = b'$ **then**
        return 1 ▷ A guessed correctly
    else
        return 0 ▷ A did not guess correctly
    end if
end procedure

Success advantage

$$\mathbf{Adv}^{\mathsf{PAY\_DETECT}}_{\mathsf{A},\Pi,\mathcal{B}}(s) \leq \epsilon(s)$$

IF

$$\mathbf{Adv}^{\mathsf{PAY\_DETECT}}_{\mathsf{A},\Pi,\mathcal{B}}(s) = \left| \frac{1}{2} - \Pr\left[\mathsf{PAY\_DIST\_EXP}^{\Pi,\mathcal{B}}_{\mathsf{A}}(1^s) = 1\right] \right| .$$

Is negligible for every adversary
Then, SE has pseudorandom ciphertext

Security proof of BLOCCE

Proposition 3. BLOCCE *securely embeds into a simplified ideal blockchain* $\mathcal{B}$. *For every probabilistic polynomial time adversary* A *there is a probabilistic polynomial time adversary* A′ *such that*

$$\mathbf{Adv}^{\mathsf{PRC}}_{\mathsf{A}',\mathsf{SE}}(s) = \mathbf{Adv}^{\mathsf{PAY\_DETECT}}_{\mathsf{A},\mathsf{BLOCCE},\mathcal{B}}(s) \leq \epsilon(s),$$

where SE *is the encryption scheme used in* BLOCCE *and* ϵ *is a negligible function.*

$$\mathbf{Adv}^{\mathsf{PAY\_DETECT}}_{\mathsf{A},\mathsf{BLOCCE},\mathcal{B}}(s) \leq \epsilon(s).$$

Security proof of BLOCCE

Algorithm 5 First Stage of the Adversary A'

procedure $\mathsf{A}'^{\mathsf{Enc}_k}_1(1^s)$
    Initialize a blockchain $\mathcal{B}$
    $(s_k, p_k) \leftarrow \mathsf{Gen}_\Sigma(1^s)$
    $(m, S) \leftarrow \mathsf{A}_1^{\mathsf{Read},\mathsf{Submit}}(p_k)$ ▷ Answers the queries according to the specification of $\mathcal{B}$
    $S' \leftarrow$ state and internal information of $\mathcal{B}$
    output $(m, (S, S', p_k, s_k))$
end procedure

Algorithm 6 Second Stage of the Adversary A'

procedure $\mathsf{A}'^{\mathsf{Enc}_k}_2(c, (S, S', p_k, s_k))$
    Initialize a blockchain $\mathcal{B}$ according to the state S'
    $(\lambda, k) \leftarrow \mathsf{Gen}_{\mathsf{BLOCCE}}(1^s)$
    Embed $\lambda || c$ into $\mathcal{B}$ by simulating Embed
    $b' \leftarrow \mathsf{A}_2^{\mathsf{Read},\mathsf{Submit}}(p_k, S)$
    output b'
end procedure

Security proof of BLOCCE

1. Suppose first that $D = 1$ and A' was given the correct $c \leftarrow \mathsf{Enc}(k, m)$. Then A' embeds $\lambda || c$ into the blockchain which follows the payment distinguishing experiment for A for the case $b = 1$. Since A'_2 outputs the same bit b' as A_2 we have

$$\Pr\left[\mathsf{A}' \text{ succeeds in } \mathsf{PRC\_EXP} | D = 1\right] = \Pr\left[\mathsf{A} \text{ succeeds} | D = 1\right].$$

2. Suppose now that $D = 0$ and c is a uniformly random string. By the description of $\mathsf{Gen}_{\mathsf{BLOCCE}}$, λ is also uniformly random, meaning that a uniformly random string $\lambda || c$ gets embedded into the blockchain. By the description of the payment distinguishing experiment, this is equal to the case $b = 0$ and

$$\Pr\left[\mathsf{A}' \text{ succeeds in } \mathsf{PRC\_EXP} | D = 0\right] = \Pr\left[\mathsf{A} \text{ succeeds} | D = 0\right].$$

Security proof of BLOCCE

$$\Pr\left[\mathsf{PRC\_EXP}_{\mathsf{A}'}^{\mathsf{SE}}(1^s) = 1\right] = \Pr\left[\mathsf{PAY\_DIST\_EXP}_{\mathsf{A}}^{\mathsf{BLOCCE},\mathcal{B}}(1^s) = 1\right].$$

$$\mathbf{Adv}_{\mathsf{A}',\mathsf{SE}}^{\mathsf{PRC}}(s) = \mathbf{Adv}_{\mathsf{A},\mathsf{BLOCCE},\mathcal{B}}^{\mathsf{PAY\_DETECT}}(s).$$

$$\mathbf{Adv}_{\mathsf{A},\mathsf{BLOCCE},\mathcal{B}}^{\mathsf{PAY\_DETECT}}(s) = \mathbf{Adv}_{\mathsf{A}',\mathsf{SE}}^{\mathsf{PRC}}(s) \leq \epsilon(s),$$