政大在職碩班「專題研討」

Deniable Encryption

Speaker: Po-Wen Chi

Department of Computer Science and Information Engineering,

National Taiwan Normal University

2020/11/16

Who am I?

- 紀博文 Po-Wen Chi
- email: neokent@gapps.ntnu.edu.tw
- Experience:
  - National Taiwan Normal University
  - Arcadyan Technologies
  - Institute for Information Industry
- Interests:
  - Applied Cryptography
  - Network Security
  - Next Generation Network

Outline

- Introduction to Deniable Encryption.
- Shared-Key Deniable Encryption.
- Public-Key Deniable Encryption.
- Bi-Deniable Public-Key Deniable Encryption.
- Block-wise Deniable Encryption.
- Reference.

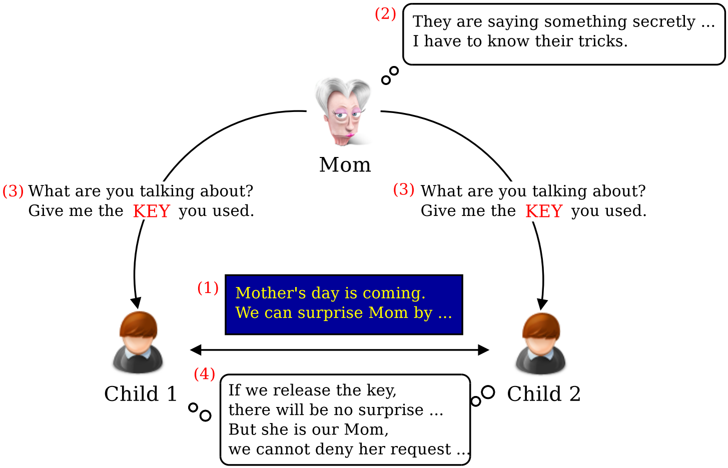
Don't Worry.

This is not a mathematic course.

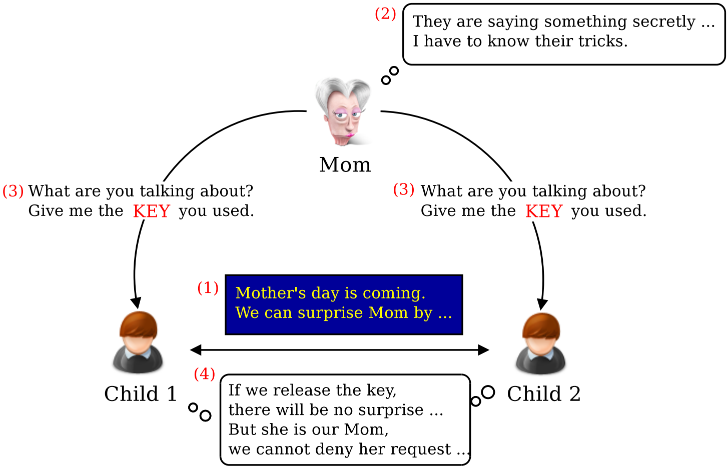
Everything will be described in plain Chinese.

Introduction to Deniable Encryption

Deniable Encryption



Deniable Encryption



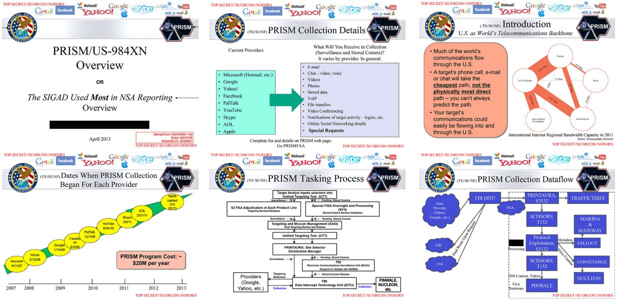
Deniable Encryption!!

This is a Stupid Scenario!!

Actually, this is a real story.

This is a Stupid Scenario!!

Actually, this is a real story.



Email Service Provider

The U.S. government commanded Snowden's email provider, Lavabit, to release the private key. Will you release or not? Why?

Email Service Provider

The U.S. government commanded Snowden's email provider, Lavabit, to release the private key. Will you release or not? Why?

Email Service Provider

The U.S. government commanded Snowden's email provider, Lavabit, to release the private key. Will you release or not? Why?

On January 20, 2017, Lavabit owner Ladar Levison relaunched the service.

Deniable Encryption

- An encryption scheme is *deniable* if the entities can generate *plausible* keys or random coins that will satisfy the authority.
- Usage: Protect people from subpoenas or legal coercion.
  - Ex: E-Voter, Journalist, Whistle-blowers.
- Theoretical Properties:
  - Non-committing.
  - Against selective-opening attacks.
  - Incoercible multi-party computation.

Selective Opening Attack

- Given a public key encryption scheme:
  - $c = (c[1], c[2], c[3], \ldots, c[n])$.
  - $c[i] = E(pk, m[i], r[i]), 1 \leq i \leq n$.
  - All coins $r[i]$ are random and independent.
- The adversary is allowed to corrupt some subset I :
  - $r[i], i \in I$.
  - $m[i], i \in I$.
- The security requirement is that the privacy of the unopened messages is preserved.

Definition

Computational Indistinguishable:

Let $A = \{A_n\}_{n \in N}$ and $B = \{B_n\}_{n \in N}$ be two probability distributions and $\delta : N \rightarrow [0,1]$. A and B are $\delta(n)$-close if for every polytime distinguisher D and for all large enough n,

$$|Prob(D(A_n) = 1) - Prob(D(B_n) = 1)| < \delta(n).$$

If $\delta(n)$ is negligible, A and B are *computational indistinguishable* and write $A \approx^c B$.

Definition

Correctness:

The probability that R's output is different than S's input is negligible (as a function of n).

Some Definitions

Definition

Plan-Ahead:

A somewhat *weaker* deniability property allows the encryption algorithm to have the fake messages as part of its input.

Definition

Sender Deniable:

1. Correctness.
2. Security: $E[m_1] \approx^c E[m_2]$.
3. Deniability:
   - $c = E[m_1, r_S]$.
   - A faking algorithm ϕ that $r'_S = \phi(m_1, r_S, c, m_2)$.
   - $(m_2, r'_S, c) \approx^c (m_2, r''_S, E[m_2, r''_S])$.

Definition

Receiver Deniable:

1. Correctness.
2. Security: $E[m_1] \approx^c E[m_2]$.
3. Deniability:
   - $c = E[m_1, r_R]$.
   - A faking algorithm ϕ that $r'_R = \phi(m_1, r_R, c, m_2)$.
   - $(m_2, r'_R, c) \approx^c (m_2, r''_R, E[m_2, r''_R])$.

Shared-Key Deniable Encryption

Shared-Key Deniable Encryption

- The most trivial solution is: One Time Pad, Vernam Cipher.
  - $c \leftarrow m \oplus k$.
  - $k' \leftarrow c \oplus m'$.
- m' can be chosen as late as at time of coercion.
- This scheme is not practical for most cases.

Shared-Key Deniable Encryption based on Pseudorandom Generators

- The message will be encrypted:
  - $m_1 = m_1^{(1)}, m_1^{(2)}, m_1^{(3)}, \ldots$.
  - Each block $m_1^{(j)}$ is n-bit.
- The fake messages:
  - $m_2 = m_2^{(1)}, m_2^{(2)}, m_2^{(3)}, \ldots$.
  - $\ldots$
  - $m_l = m_l^{(1)}, m_l^{(2)}, m_l^{(3)}, \ldots$.
- The shared key:
  - k_1: n-bit random key.
  - $k_2, \ldots, k_l$: $l - 1$ independent n-bit *fake* keys.
- A pseudorandom number generator G:
  - Expand n-bit input to $3n$-bit output.
  - Using G iteratively: $G(k_i^{(j-1)}) = k_i^{(j)} | a_i^{(j)} | b_i^{(j)}$.

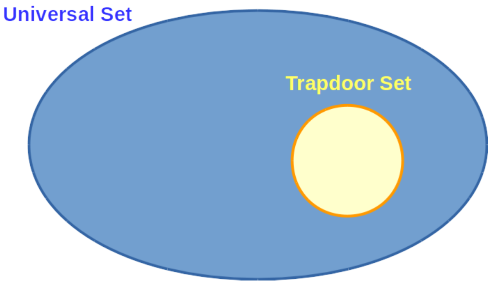
Shared-Key Deniable Encryption based on Pseudorandom Generators

- Encryption:
  - $c = c^{(1)}, c^{(2)}, c^{(3)}, \ldots$.
  - The sender finds the polynomial $Q^{(j)}$ of degree $l - 1$ such that $Q^{(j)}(a_i^{(j)}) = m_i^{(j)} + b_i^{(j)}, i = 1 \ldots l$.
  - $c^{(j)} = \langle j, Q^{(j)} \rangle$.
- Decryption:
  - $m_1^{(j)} = Q^{(j)}(a_1^{(j)}) - b_1^{(j)}$.
- Deniability:
  - Just select one of fake keys when coercion.

Public-Key Deniable Encryption

Translucent Set

- This scheme is based on the trapdoor *SPARSE sets*.
  1. A small set $S \subset \{0,1\}^t$, $|S| \leq 2^{t-k}$ for some k.
  2. It is easy to generate random element $x \in S$.
  3. Without the trapdoor d, it is infeasible to decide whether $x \in \{0,1\}^t$ was chosen from S or uniformly from $\{0,1\}^t$.



How to Construct Sparse Sets

- A trapdoor permutation $f\colon \{0,1\}^s \to \{0,1\}^s$.
- A hard-core bit function $B : \{0,1\}^s \to \{0,1\}$.
- Construction I:
  - $t = sk$.
  - Represent $x \in \{0,1\}^t$ as a vector $x = x_1 x_2 \ldots x_k$, where $x_i \in \{0,1\}^s$.
  - $S = \{x \in \{0,1\}^{sk} | \forall i = 1 \ldots k, B(f^{-1}(x_i)) = 0\}$.
  - $|S| = 2^{(s-1)k} = 2^{t-k}$.

How to Construct Sparse Sets

- A trapdoor permutation $f\colon \{0,1\}^s \to \{0,1\}^s$.
- A hard-core bit function $B : \{0,1\}^s \to \{0,1\}$.
- Construction I:
  - $t = sk$.
  - Represent $x \in \{0,1\}^t$ as a vector $x = x_1 x_2 \dots x_k$, where $x_i \in \{0,1\}^s$.
  - $S = \{x \in \{0,1\}^{sk} | \forall i = 1 \dots k, B(f^{-1}(x_i)) = 0\}$.
  - $|S| = 2^{(s-1)k} = 2^{t-k}$.
- Construction II:
  - $t = s + k$.
  - Represent $x \in \{0,1\}^t$ as a vector $x = x_0, b_1 b_2 \dots b_k$, where $x_0 \in \{0,1\}^s$ and $b_i \in \{0,1\}$.
  - $S = \{x \in \{0,1\}^{s+k} | \forall i = 1 \dots k, B(f^{-i}(x_0)) = b_i\}$.
  - $|S| = 2^s = 2^{t-k}$.

Public-Key Sender-Deniable Encryption Scheme 01

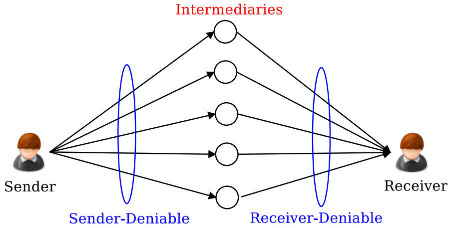
- The Basic Scheme:
  - Bitwise encryption.
  - Public key: $S \subset \{0,1\}^t$; Private key: the trapdoor d.
  - Encryption:
    - To encrypt 1, send a random element from S.
    - To encrypt 0, send a random element from $\{0,1\}^t$.
  - Decryption: Check if the cipher c is in S.
  - Dinability: If the encrypted bit is 1, claim that the cipher is chosen from $\{0,1\}^t$ instead from S.
  - Only half deniablity.
  - The probability of decryption error is $\frac{2^{t-k}}{2^t} = 2^{-k}$.

Public-Key Sender-Deniable Encryption Scheme 02

- The Parity Scheme:
  - Public key: $S \subset \{0,1\}^t$, $R = \{0,1\}^t$; Private key: the trapdoor d.
  - Use $V \in \{S,R\}^n$ to denote a length n vector.
  - Encryption:
    - To encrypt 1, send a $V \in_R \{S,R\}^n$ where V randomly contains odd S-elements.
    - To encrypt 0, send a $V \in_R \{S,R\}^n$ where V randomly contains even S-elements.
  - Decryption: Reveal the number of elements in V that belongs to S.
  - Deniability: The sender can claim V has $i-1$ S-elements rather than i.
  - The probability of decryption error is at most $n2^{-k}$.

Receiver-Deniability and Bi-Deniability

- Receiver-Deniability from Sender-Deniability:
  - If there is a Sender-Deniable scheme, the receiver first sends a deniable message r to the sender.
  - The sender sends $m \oplus r$ to the receiver.
- Bi-Deniability:
  - Sender-and-Receiver-Deniability.
  - $\oplus_i b_i = b$.
  - As long as one intermediary node is uncoerced, the sender and the receiver can deny their messages.



What is the Problem?

- We do not like bitwise encryption.
- We do not like interactive encryption.
- We do not like third-party.
- We do not like decryption error.

Bi-Deniable Public-Key Deniable Encryption

Multi-Distributional Bi-Deniable Scheme

- A. O'Neil, C. Peikert and B. Waters proposed Multi-Distributional Bi-Deniable Scheme based on Simulatable Public-Key Encryption.

Definition

Multi-Distributional:

- Multi-Distributional means the parties run alternative key-generation and encryption algorithms for equivocable communication, but claim under coercion to have run the prescribed algorithms.
- Multi-Distributional means the scheme contains normal and deniable encryption at the same time.

A Philosophical Question

- Why would anyone ever choose to send a message according to the non-deniable encryption algorithm?
- It is impossible to eliminate this option because the coercer would *know* that the sender is lying.
- The purpose of deniability is not at all to convince the coercer, but to *preempt coercion* in the first place.

Bi-Deniable Scheme

Sender-Deniable	Receiver-Deniable
$pk \leftarrow \mathsf{Gen}(1^n, r_R)$	$(pk, fk) \leftarrow \mathsf{DenGen}(1^n)$
$c \leftarrow \mathsf{DenEnc}(pk, m, r_S)$	$c \leftarrow \mathsf{Enc}(pk, m, r_S)$
	$r_R^* \leftarrow \mathsf{RecFake}(pk, fk, c, m')$
$r_S^* \leftarrow \mathsf{SendFake}(pk, r_S, m, m')$	
Return (pk, c, r_S^*)	(pk, c, r_R^*)

Bi-Deniable
$(pk, fk) \leftarrow \mathsf{DenGen}(1^n)$
$c \leftarrow \mathsf{DenEnc}(pk, m, r_S)$
$r_R^* \leftarrow \mathsf{RecFake}(pk, fk, c, b)$
$r_S^* \leftarrow \mathsf{SendFake}(pk, r_S, m, m')$
Return (pk, c, r_S^*, r_R^*)

Simulatable Public-Key System

Definition

Given a public-key system (K, E, D, M), where

- K: key generation algorithm; E: encryption algorithm;
- D: decryption algorithm; M: message space generator.

(K, E, D, M) is a simulatable public key system if $(\tilde{K}, \tilde{K}^{-1}, C, C^{-1})$ exists:

- **Oblivious public key generation**:

$$r \leftarrow R, (P, S) \leftarrow K(r), r' \leftarrow \tilde{K}^{-1}(P).$$

$$r'' \leftarrow R, (P'', S'') \leftarrow \tilde{K}(r'').$$

(r', P) and (r'', P'') are computationally indistinguishable.

Simulatable Public-Key System

Definition

- **Oblivious ciphertext generation**:

$$(P, S) \leftarrow K, r_1 \leftarrow R, C_1 \leftarrow C(P, r_1).$$

$$r_2 \leftarrow R, C_2 \leftarrow E_P(M, r_2), r_2' \leftarrow C^{-1}(C_2, P).$$

(P, r_1, C_1) and (P, r_2', C_2) are computationally indistinguishable.

Definition

- **Semantic security**:

$$r \leftarrow R, (P, S) \leftarrow K(r).$$

$$r_0 \leftarrow R, C_0 \leftarrow E_P(M_0, r_0);$$

$$r_1 \leftarrow R, C_1 \leftarrow E_P(M_1, r_1).$$

(P, M_0, M_1, C_0) and (P, M_0, M_1, C_1) are computationally indistinguishable.

ElGamal is Simulatable under DDH Assumption

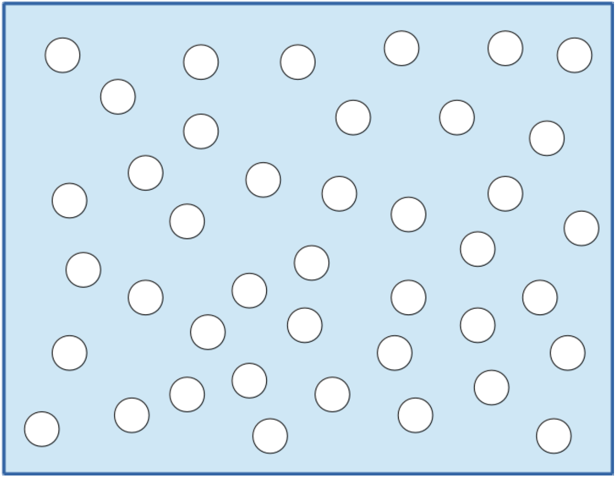
- ElGamal Encryption:
  - Public key: $h = g^x, p, g$.
  - Private key: x.
  - Encryption: (g^y, mh^y).
- Oblivious:
  - $\tilde{K} = h$.
  - $\tilde{K}^{-1}(p, g, h) = (p, g, h)$.
  - $C = (y_1, y_2)$, where $y_1 \leftarrow R, y_2 \leftarrow R$.

Bideniable Encryption from Simulatable System (1/3)

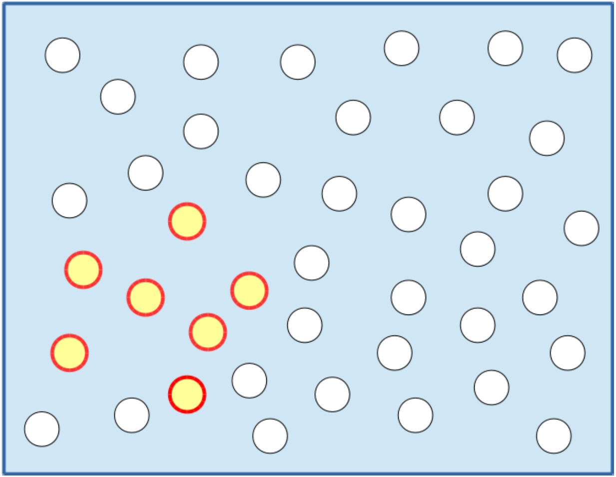
BI-DEN.Gen(1^n)	BI-DEN.Enc(pk, b)
$R \leftarrow P_n([5n])$	$S \leftarrow P_n([5n])$
For $i = 1$ to $5n$ do:	For $i = 1$ to $5n$ do:
If $i \in R$ then	If $i \in S$ then
$pk_i \leftarrow \mathrm{Gen}(1^n, r_{R,i})$	$c_i \leftarrow \mathrm{Enc}(pk_i, b, r_{S,i})$
Else	Else
$pk_i \leftarrow \mathrm{OGen}(1^n, r_{R,i})$	$c_i \leftarrow \mathrm{OEnc}(pk_i, r_{S,i})$
$pk \leftarrow pk_1 \| pk_2 \| \ldots \| pk_{5n}$	$c \leftarrow c_1 \| c_2 \| \ldots \| c_{5n}$
Return pk	Return c

BI-DEN.Dec$((R, r_R), c)$
For $i \in R$ do:
$d_i \leftarrow \mathrm{Dec}(r_{R,i}, c_i)$
If most d_i's are 1 then Return 1
Else Return 0

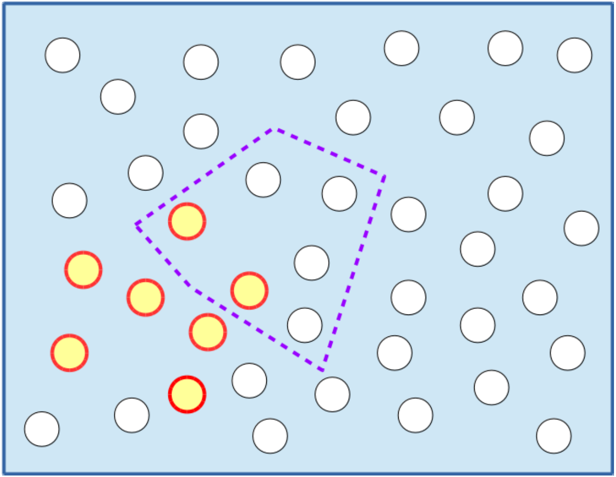
Encryption.



Encryption.



Decryption.



Proof of Correctness

- BI-DEN.Enc should be correct.
- The tail of the hypergeometric distribution:

$$Pr[X \leq E[X] - ty = y(\frac{M}{N} - t)] \leq e^{-2t^2 y}$$

$$Pr[X \leq E[X] + ty = y(\frac{M}{N} + t)] \leq e^{-2t^2 y}$$

- BI-DEN.Enc:
  - Let I be $|S \cap R|$ and D be $R \setminus S$ and $d_i = b$.
  - Decryption error: $D + I < \frac{n}{2}$.
  - If $\frac{n}{10} < I \leq \frac{n}{2}$,

$$Pr[D \leq \frac{n - I}{2} - \frac{I}{2}] \leq Pr[D \leq (1 - \frac{1}{9})E[D]] \leq e^{-\frac{n-I}{324}} \leq e^{-\frac{n}{648}}$$

Bideniable Encryption from Simulatable System (2/3)

BI-DEN.DenGen(1^n)	BI-DEN.DenEnc(pk, b)
$R \leftarrow P_n([5n])$	$S_0 \leftarrow P_n([5n])$
For $i = 1$ to $5n$ do:	$S_1 \leftarrow P_n([5n] \setminus S_0)$
$\quad pk_i \leftarrow \mathsf{Gen}(1^n, r_{R,i})$	$Y \leftarrow P_n([5n] \setminus (S_0 \cup S_1))$
$pk \leftarrow pk_1 \Vert pk_2 \Vert \ldots \Vert pk_{5n}$	For $i = 1$ to $5n$ do:
$r \leftarrow r_{R,1} \Vert r_{R,2} \Vert \ldots \Vert r_{R,5n}$	$\quad$ If $i \in S_0$ then $c_i \leftarrow \mathsf{Enc}(pk_i, 0, r_{S,i})$
	$\quad$ If $i \in S_1$ then $c_i \leftarrow \mathsf{Enc}(pk_i, 1, r_{S,i})$
	$\quad$ If $i \in Y$ then $c_i \leftarrow \mathsf{Enc}(pk_i, b, r_{S,i})$
Return $(pk, (R, r))$	$\quad$ Else $c_i \leftarrow \mathsf{OEnc}(pk_i, r_{S,i})$
	$c \leftarrow c_1 \Vert c_2 \Vert \ldots \Vert c_{5n}$
	Return c

Bideniable Encryption from Simulatable System (3/3)

$\underline{\text{BI-DEN.FakeCoins}(pk, fk, r_S, b, b')}$

$c \leftarrow \text{BI-DEN.Enc}(pk, b, r_S)$

$z \leftarrow \text{HGD}(5n, n, n)$

$Z \leftarrow P_z(S_{b'})$

$Z' \leftarrow P_{n-z}([5n] \setminus (S_0 \cup S_1 \cup Y))$

$R^* \leftarrow Z \cup Z'$

$S^* \leftarrow S_{b'}$

For $i = 1$ to $5n$ do:

  If $i \in S^*$, then $r^*_{S,i} \leftarrow r_{S,i}$

  Else $r^*_{S,i} \leftarrow I_{OEnc}(pk_i, c_i)$

  If $i \in R^*$, then $r^*_{R,i} \leftarrow r_{R,i}$

  Else $r^*_{R,i} \leftarrow I_{OGen}(pk_i)$

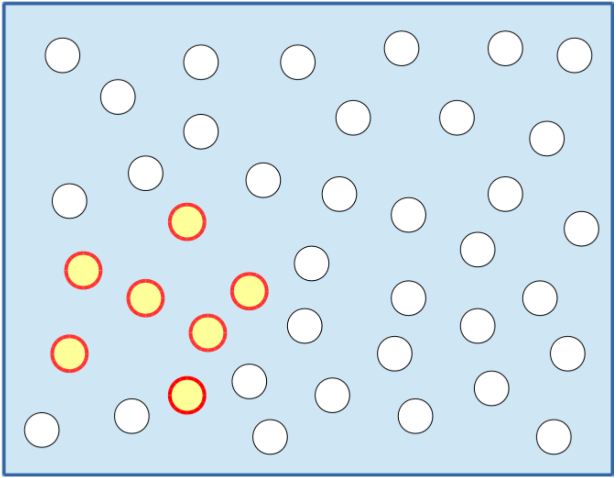
$r^*_S \leftarrow r^*_{S,1} \| r^*_{S,2} \| \ldots \| r^*_{S,5n}$

$r^*_R \leftarrow r^*_{R,1} \| r^*_{R,2} \| \ldots \| r^*_{R,5n}$

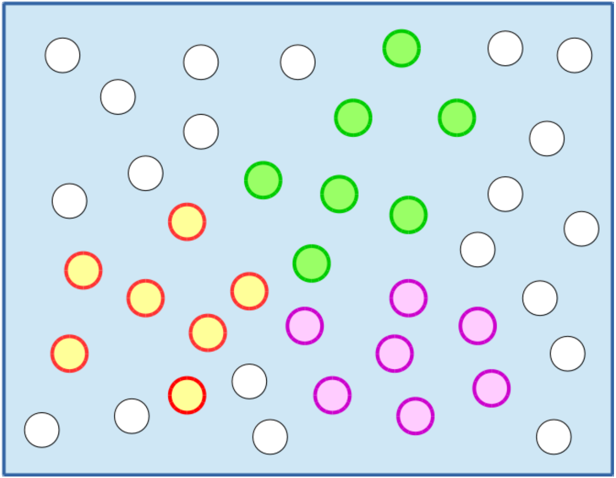
Return (r^*_S, r^*_R)

- Hypergeometric Distribution:
  - $P_{\text{HGD}}(x, N, M, y) = \frac{C_x^M C_{y-x}^{N-M}}{C_y^N}$.
  - $\text{HGD}(N, M, y)$ is the expectation.

Claim.



In fact.



Proof of Deniability (1/3)

Experiment G_0	Experiment G_1
$S_b \leftarrow P_n([5n])$	$S_b \leftarrow P_n([5n])$
$R \leftarrow P_n([5n])$	$z \leftarrow HGD(5n, n, n)$
$S_{1-b} \leftarrow P_n([5n] \setminus (S_b \cup R))$	$Z \leftarrow P_z(S_b)$
$Y \leftarrow P_n([5n] \setminus (S_b \cup S_{1-b} \cup R))$	$Z' \leftarrow P_{n-z}([5n] \setminus S_b)$
	$R \leftarrow Z \cup Z'$
	$S_{1-b} \leftarrow P_n([5n] \setminus (S_b \cup R))$
	$Y \leftarrow P_n([5n] \setminus (S_b \cup S_{1-b} \cup R))$
For $i = 1$ to $5n$ do:	For $i = 1$ to $5n$ do:
If $i \in R$, $pk_i \leftarrow \mathrm{Gen}(1^n, r_{R,i})$	If $i \in R$, $pk_i \leftarrow \mathrm{Gen}(1^n, r_{R,i})$
Else $pk_i \leftarrow \mathrm{OGen}(1^n, r_{R,i})$	Else $pk_i \leftarrow \mathrm{OGen}(1^n, r_{R,i})$
If $i \in S_b$, $c_i \leftarrow \mathrm{Enc}(pk_i, b, r_{S,i})$	If $i \in S_b$, $c_i \leftarrow \mathrm{Enc}(pk_i, b, r_{S,i})$
Else $c_i \leftarrow \mathrm{OEnc}(pk_i, r_{S,i})$	Else $c_i \leftarrow \mathrm{OEnc}(pk_i, r_{S,i})$
Return $(pk, c, (R, r_R), (S_b, r_S))$	Return $(pk, c, (R, r_R), (S_b, r_S))$

Proof of Deniability (2/3)

Experiment G_1	Experiment G_2
$S_b \leftarrow P_n([5n])$	$S_b \leftarrow P_n([5n])$
$z \leftarrow HGD(5n, n, n)$	$S_{1-b} \leftarrow P_n([5n] \setminus S_b)$
$Z \leftarrow P_z(S_b)$	$Y \leftarrow P_n([5n] \setminus (S_b \cup S_{1-b}))$
$Z' \leftarrow P_{n-z}([5n] \setminus S_b)$	$z \leftarrow HGD(5n, n, n)$
$R \leftarrow Z \cup Z'$	$Z \leftarrow P_z(S_b)$
$S_{1-b} \leftarrow P_n([5n] \setminus (S_b \cup R))$	$Z' \leftarrow P_{n-z}([5n] \setminus (S_b \cup S_{1-b} \cup Y)$
$Y \leftarrow P_n([5n] \setminus (S_b \cup S_{1-b} \cup R))$	$R \leftarrow Z \cup Z'$
For $i = 1$ to $5n$ do:	For $i = 1$ to $5n$ do:
If $i \in R$, $pk_i \leftarrow \mathrm{Gen}(1^n, r_{R,i})$	If $i \in R$, $pk_i \leftarrow \mathrm{Gen}(1^n, r_{R,i})$
Else $pk_i \leftarrow \mathrm{OGen}(1^n, r_{R,i})$	Else $pk_i \leftarrow \mathrm{OGen}(1^n, r_{R,i})$
If $i \in S_b$, $c_i \leftarrow \mathrm{Enc}(pk_i, b, r_{S,i})$	If $i \in S_b$, $c_i \leftarrow \mathrm{Enc}(pk_i, b, r_{S,i})$
Else $c_i \leftarrow \mathrm{OEnc}(pk_i, r_{S,i})$	Else $c_i \leftarrow \mathrm{OEnc}(pk_i, r_{S,i})$
Return $(pk, c, (R, r_R), (S_b, r_S))$	Return $(pk, c, (R, r_R), (S_b, r_S))$

Experiment G_2	Experiment G_3
For $i = 1$ to $5n$ do:	For $i = 1$ to $5n$ do:
	$pk_i \leftarrow \mathsf{Gen}(1^n, r_{R,i})$
If $i \in R$, $pk_i \leftarrow \mathsf{Gen}(1^n, r_{R,i})$	If $i \in R$, $r^*_{R,i} \leftarrow r_{R,i}$
Else $pk_i \leftarrow \mathsf{OGen}(1^n, r_{R,i})$	Else $r^*_{R,i} \leftarrow \mathsf{I}_{\mathsf{OGen}}(pk_i)$
If $i \in S_b$, $c_i \leftarrow \mathsf{Enc}(pk_i, b, r_{S,i})$	If $i \in S_b$
Else $c_i \leftarrow \mathsf{OEnc}(pk_i, r_{S,i})$	$c_i \leftarrow \mathsf{Enc}(pk_i, b, r_{S,i})$, $r^*_{S,i} \leftarrow r_{S,i}$
	Else if $i \in S_{1-b}$
	$c_i \leftarrow \mathsf{Enc}(pk_i, 1-b, r_{S,i})$
	$r^*_{S,i} \leftarrow \mathsf{I}_{\mathsf{OEnc}}(pk_i, c_i)$
	Else if $i \in Y$
	$c_i \leftarrow \mathsf{Enc}(pk_i, b', r_{S,i})$
	$r^*_{S,i} \leftarrow \mathsf{I}_{\mathsf{OEnc}}(pk_i, c_i)$
	Else
	$c_i \leftarrow \mathsf{OEnc}(pk_i, r_{S,i})$, $r^*_{S,i} \leftarrow r_{S,i}$

Question

How does the receiver know S_0, S_1, Y ?

Block-wise Deniable Encryption

Problems

- The proposed schemes are bitwise.
- Cost too much.
- Consistency issue.

Plan-Ahead Bi-Deniable Encryption (1/3)

$\mathsf{Gen}(1^n)$:	$\mathsf{Enc}(pk, m)$:
$(pk, sk) \leftarrow \mathsf{Gen}'(1^n)$	$K_0 \leftarrow \{0,1\}^n, b \leftarrow \{0,1\}$
Return (pk, sk)	$c_{asym} \leftarrow \mathsf{Enc}'(pk, K_0 \parallel 0^n \parallel b)$
	$c_0 \leftarrow E(K_0, m)$
	$c_1 \leftarrow \{0,1\}^{\lvert c_b \rvert}$
	Return $c_{asym} \parallel c_b \parallel c_{1-b}$

Plan-Ahead Bi-Deniable Encryption (2/3)

DenGen(1^n):	PADenEnc(pk, m_0, m_1):
$(pk, sk, fk) \leftarrow \mathsf{Gen}'(1^n)$	$K_0, K_1 \leftarrow \{0,1\}^n, b \leftarrow \{0,1\}$
Return (pk, sk, fk)	$c_{asym} \leftarrow \mathsf{DenEnc}'(pk, K_0 \parallel K_1 \parallel b)$
	$c_0 \leftarrow E(K_0, m_0)$
	$c_1 \leftarrow E(K_1, m_1)$
	Return $c_{asym} \parallel c_b \parallel c_{1-b}$

Plan-Ahead Bi-Deniable Encryption (3/3)

$\mathsf{PARecFake}(fk, c, K_0 \parallel K_1 \parallel b, b')$:	$\mathsf{PASendFake}(pk, c, r_S, b')$:
$c \leftarrow c_{asym} \parallel c_0 \parallel c_1$	$c \leftarrow c_{asym} \parallel c_0 \parallel c_1$
	$K_0 \parallel K_1 \parallel b \parallel r \leftarrow r_S$
$x \leftarrow K_0 \parallel K_1 \parallel b$	$x \leftarrow K_0 \parallel K_1 \parallel b$
$y \leftarrow K_{b'} \parallel 0^n \parallel b'$	$y \leftarrow K_{b'} \parallel 0^n \parallel b'$
$r_R^* \leftarrow \mathsf{RecFake}'(fk, c_{asym}, x, y)$	$r_S^* \leftarrow \mathsf{SendFake}'(pk, c_{asym}, r, x, y)$
Return r_R^*	Return r_S^*

Chameleon Hash

Chameleon Hash is a trapdoor one-way function with three requirements:

1. Semantic Security.
2. Collision Resistance.
3. Collision Forgery with the trapdoor.

Most trapdoor pseudo random permutation functions can be used as chameleon hash functions.

Ciphertext Pattern

- Normal Ciphertext: $V = CH(t_b, M)$.
- Deniable Ciphertext: $V = CH(t_b, M) = CH(t_{1-b}, M^*)$.

Note: b can be used as a sender proof.

Reference

Reference

1. R. Canetti, C. Dwork, M. Naor and R. Ostrovsky. Deniable Encryption. Crypto 1997.
2. A. O'Neil, C. Peikert and B. Waters. Bi-Deniable Public-Key Encryption. Crypto 2011.
3. P. Chi and C. Lei. Audit-Free Cloud Storage via Deniable Attribute-Based Encryption. IEEE TCC 2018.

Q and A