Cryptography
CS 555

Week 2:

- Computational Security against Eavesdropper
- Constructing Secure Encryption Schemes against Eavesdropper
- Chosen Plaintext Attacks and CPA-Security

Readings: Katz and Lindell Chapter 3.1-3.4

Homework 1 Released: Due Feb 4 at 11:59 PM on Gradescope

Recap

- Historical Ciphers (and their weaknesses)
- Three Equivalent Definitions of Perfect Secrecy
- One-time-Pads
- Concrete vs Asymptotic Approach to Security
  - Probabilistic Polynomial Time (PPT)
  - Negligible Function

Private Key Encryption Syntax (Revisited)

- Message Space: $\mathcal{M}$
- Key Space: $\mathcal{K}$
- Three Algorithms
  - $\mathrm{Gen}(\mathbf{1}^{n}; R)$ (Key-generation algorithm)
    - **Input:** 1^n **(security parameter in unary)** + Random Bits
    - **Output:** Secret key $\mathrm{k} \in \mathcal{K}$
  - $\mathrm{Enc}_{\mathrm{k}}(m; \boldsymbol{R})$ (Encryption algorithm)
    - **Input:** Secret key $\mathrm{k} \in \mathcal{K}$ and message $\mathrm{m} \in \mathcal{M}$ + Ra
    - **Output:** ciphertext *c*
  - $\mathrm{Dec}_{\mathrm{k}}(c)$ (Decryption algorithm)
    - **Input:** Secret key $\mathrm{k} \in \mathcal{K}$ and a ciphertex c
    - **Output:** a plaintext message $\mathrm{m} \in \mathcal{M}$ *or* $\perp$ ***(i.e "Fail")***
- Invariant: $\mathrm{Dec}_{\mathrm{k}}(\mathrm{Enc}_{\mathrm{k}}(\mathrm{m}))=\mathrm{m}$

Requirement: all three algorithms run in probabilistic polynomial time

Quick Comment on Notation:
$\mathrm{K} = \mathrm{Gen}(\mathbf{1}^{n}; R)$ vs.
$\mathrm{K} \leftarrow \mathrm{Gen}(\mathbf{1}^{n})$

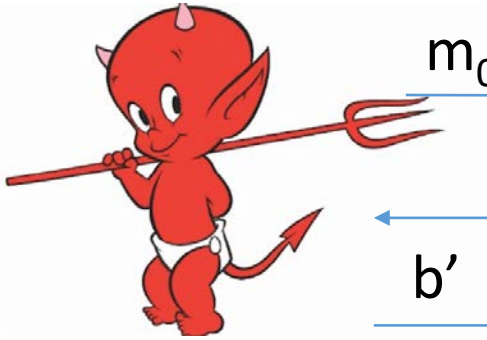
Private Key Encryption Syntax (Revisited)

- Message Space: $\mathcal{M}$
- Key Space: $\mathcal{K}$
- Three Algorithms
  - $\mathrm{Gen}(\mathbf{1}^{n}; R)$ (Key-generation algorithm)
    - **Input:** 1^n **(security parameter in unary)** + Random Bits
    - **Output:** Secret key $\mathrm{k} \in \mathcal{K}$
  - $\mathrm{Enc}_{\mathrm{k}}(m; \boldsymbol{R})$ (Encryption algorithm)
    - **Input:** Secret key $\mathrm{k} \in \mathcal{K}$ and message $\mathrm{m} \in \mathcal{M}$ + Ra
    - **Output:** ciphertext *c*
  - $\mathrm{Dec}_{\mathrm{k}}(c)$ (Decryption algorithm)
    - **Input:** Secret key $\mathrm{k} \in \mathcal{K}$ and a ciphertex c
    - **Output:** a plaintext message $\mathrm{m} \in \mathcal{M}$ *or* $\perp$ ***(i.e "Fail")***
- Invariant: $\mathrm{Dec}_{\mathrm{k}}(\mathrm{Enc}_{\mathrm{k}}(\mathrm{m}))=\mathrm{m}$

Requirement: all three algorithms run in probabilistic polynomial time

Quick Comment on Notation:
$\mathrm{K} = \mathrm{Gen}(\mathbf{1}^{n}; R)$ vs.
$\mathrm{K} \leftarrow \mathrm{Gen}(\mathbf{1}^{n})$

Adversarial Indistinguishability Experiment



m_0, m_1

c

b'

Random bit b

K ← Gen(1^n)

c ← $Enc_K(m_b)$

ppt attacker

negligible function

$$\forall \text{ devil } \exists \mu \text{ devil } \Pr\left[\text{ devil } Guesses\ b' = b\right] \leq \frac{1}{2} + \mu \text{ devil } (n)$$

Adversarial Indistinguishability Experiment

$Formally, let\ \Pi = (Gen, Enc, Dec)\ denote\ the\ encryption\ scheme,$
$call\ the\ game\ the\ adversarial\ indistinguishability\ experiment\ and$
$define\ a\ random\ variable\ PrivK^{eav}_{A,\Pi}(1^n)\ as\ follows$

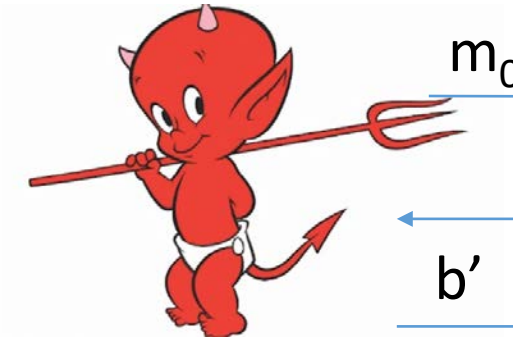
$$PrivK^{eav}_{A,\Pi}(1^n) = \begin{cases} 1 & \text{if } b = b' \\ 0 & \text{otherwise} \end{cases}$$

$\Pi\ has\ indistinguishable\ encryptions\ in\ the\ presence\ of$
$an\ eavesdropper\ if\ for\ all\ PPT\ adversary\ A, there\ exists\ a$
negligible function $\mu(.)$ such that

$$\Pr[PrivK^{eav}_{A,\Pi} = 1] \leq \frac{1}{2} + \mu(n)$$

bit b
(1^n)
(m_b)

EAV-Secure



m_0, m_1

c

b’

Random bit b

K ← Gen(1^n)

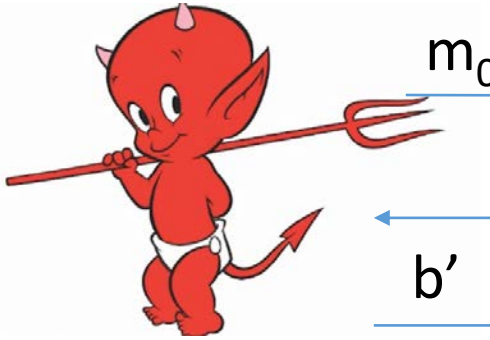
c ← $Enc_K(m_b)$

ppt attacker

negligible function

$$\forall \text{ppt } \exists \mu \quad \Pr\left[\text{Guesses } b' = b\right] \leq \frac{1}{2} + \mu(n)$$

$(t(n), \varepsilon(n))$-EAV-Secure (Concrete Version)



m_0, m_1

c

b’

Random bit b

K ← Gen(1^n)

c = $Enc_K(m_b)$

running in time at most $t(n)$

specific function
(same for all attackers)

$$\forall \text{ Pr}\left[\text{ } Guesses\ b' = b \right] \leq \frac{1}{2} + \varepsilon(n)$$

Aside: Message and Ciphertext Length

- In the previous game we typically require that $|m_0|=|m_1|$. Why?
- It is impossible to support arbitrary length messages while hiding all information about plaintext length
- **Limitation:** When could message length be sensitive?
  - Numeric data (5 figure vs 6 figure salary)
  - Database Searches: number of records returned can reveal information about the query
  - Compressed Data: Short compressed string indicates that original plaintext has a lot of redundancy (e.g., CRIME attack on session cookies in HTTPS)

Implications of Indistinguishability

i^{th} bit of message

Theorem 3.10: Let (Gen, Enc, Dec) be a fixed-length private-key encryption scheme for message of length ℓ that satisfies indistinguishability (prior definition) then for all PPT attackers A and any $i \leq \ell$ we have

$$\Pr\left[A\left(1^n, \mathrm{Enc}_K(m)\right) = m^i\right] \leq \frac{1}{2} + \mathrm{negl}(n)$$

Where the randomness is taken over $K \leftarrow \mathrm{Gen}(1^n)$, uniform $m \in \{0,1\}^\ell$ and the randomness of Enc and A.

Remark: A bit weaker than saying eavesdropping attacker obtains ``no additional" information about message m.

Semantic Security

Definition 3.12: Let $\Pi = (\text{Gen, Enc, Dec})$be a fixed-length private key encryption scheme for message of length ℓ. We say that the scheme is semantically secure if for all PPT attackers A there exists a PPT algorithm A' such that for any PPT algorithm Sample all any polynomial time computable functions f and h we have

$$|\Pr[A(1^n, \text{Enc}_K(m), h(m)) = f(m)]$$

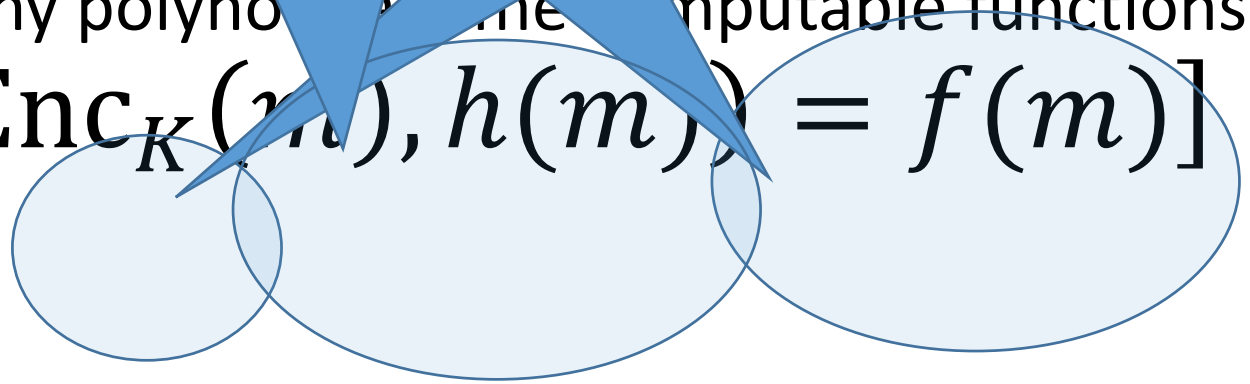
S

h(m) background knowledge the attacker might have about m.

A' doesn't even get to see an encryption of m! Just the length of m!

Definition 3.12: Let Π = (Ge… be a …length private key encryption scheme for message of length… …e scheme is semantically secure if for all PPT attackers A there e… a …gorithm A' such that for any PPT algorithm Sample all any polyno… …mputable functions f and h we have

$$\left|\Pr\left[A\left(1^n, \mathrm{Enc}_K(m), h(m)\right) = f(m)\right]\right.$$



Semantic Security

Definition 3.12: Let $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ be a fixed-length private key encryption scheme for message of length ℓ. We say that the scheme is semantically secure if for all PPT attackers A there exists a PPT algorithm A' such that for any PPT algorithm Sample all any polynomial time computable functions f and h we have

$$\left|\Pr\left[A\left(1^n, \text{Enc}_K(m), h(m)\right) = f(m)\right]\right.$$

Another Interpretation of Semantic Security

- World 2: Perfect Secrecy (Attacker doesn't even see ciphertext).
- For all attackers A' (even unbounded) with background knowledge h(m) we have

$$\Pr[A'(1^n, |m|, h(m)) = f(m)] = \Pr[f(m)|\ \ h(m),|m|]$$

- World 1: Attacker is PPT and sees ciphertext
  - Best World 1 attacker does no better than World 2 attacker
- $|\Pr[A(1^n, \mathrm{Enc}_K(m), h(m)) = f(m)] - \Pr[A'(1^n, |m|, h(m)) = f(m)]| \leq \mathrm{negl}(n)$

- What is probability over?

Week 2: Topic 2: Constructing Secure Encryption Schemes

New Goal

- ~~Define computational security~~

  ~~*If you don't understand what you want to achieve, how can you possibly know when (or if) you have achieved it?*~~

- Show how to build a symmetric encryption scheme with semantic security.

- ~~Define computational security against an attacker who sees multiple ciphertexts or attempts to modify the ciphertexts~~

Building Blocks

- Pseudorandom Generators
- Stream Ciphers

Pseudorandom Generator (PRG) G

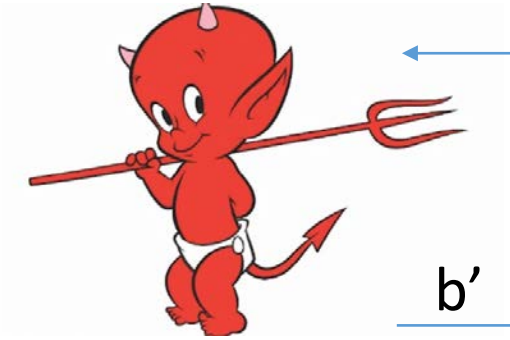
- **Input:** *Short* random seed $s \in \{0,1\}^n$
- **Output:** Longer "pseudorandom" string $G(s) \in \{0,1\}^{\ell(n)}$ with $\ell(n) > n$
  - $\ell(n)$ is called expansion factor
- **PRG Security**: For all PPT attacker A there is a negligible function $\mathrm{negl}(.)$ s.t

$$\left|\Pr_{s\in\{0,1\}^n}\left[A\left(G(s)\right)=1\right]-\Pr_{R\in\{0,1\}^{\ell(n)}}[A(R)=1]\right|\leq \mathrm{negl}(n)$$

- **Concrete Security**: We say that $G(.)$ is a $\left(t(n),\varepsilon(n)\right)$-secure PRG if for all attackers running in time at most $t(n)$ we have

$$\left|\Pr_{s\in\{0,1\}^n}\left[A\left(G(s)\right)=1\right]-\Pr_{R\in\{0,1\}^{\ell(n)}}[A(R)=1]\right|\leq \varepsilon(n)$$

PRG Security as a Game



R

b’

Random bit b

If b=1

$r \leftarrow \{0,1\}^n$

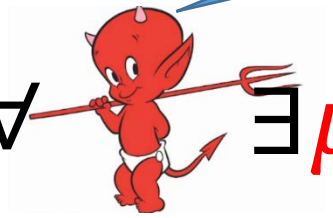
R = G(r)

Else

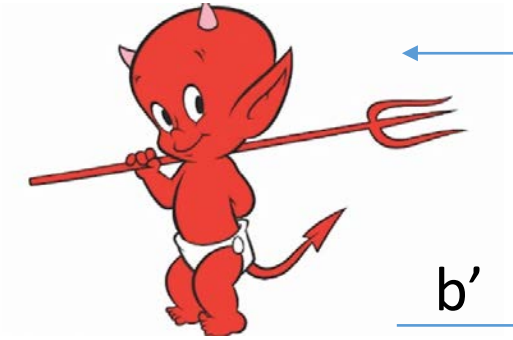
$\{0,1\}^{\ell(n)}$

ppt attacker

negligible function

$$\forall \text{  } \exists \mu \text{  } \Pr\left[\text{  } \; Guesses\ b' = b\right] \leq \frac{1}{2} + \mu \text{  }(n)$$

$(t(n), \varepsilon(n))$-Secure PRG (Concrete Version)



R

b’

Random bit b

If b=1

$r \leftarrow \{0,1\}^n$

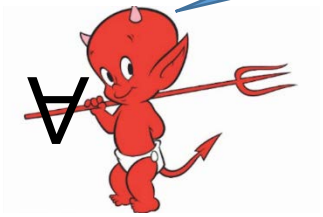
R = G(r)

Else

$n)$

running in time $t(n)$

specific function (usually negligible)



$$\forall \Pr\left[\text{Guesses } b' = b\right] \leq \frac{1}{2} + \varepsilon(n)$$

A Bad PRG

$$G(s) = s|1.$$

- What is the expansion factor?
  - Answer: $\ell(n)$=n+1

- Task: Construct a distinguisher D which breaks PRG security for G

  - One Answer: D(x|1)=1 and D(x|0)=0 for all x.
  - Analysis: Pr[D(G(s)) = 1] = ?
  - Analysis: Pr[D(R) = 1] = ?
  - $\left|\Pr_{s\in\{0,1\}^n}\left[D\left(G(s)\right)=1\right]-\Pr_{R\in\{0,1\}^{\ell(n)}}[D(R)=1]\right|=\frac{1}{2}$

One-Time-Pads + PRGs

- Encryption:
  - Secret key is the seed (K=s)

$$\mathrm{Enc}_s(\mathrm{m}) = \mathrm{G}(\mathrm{s}) \oplus m$$
$$\mathrm{Dec}_s(\mathrm{c}) = \mathrm{G}(\mathrm{s}) \oplus c$$

  - **Advantage**: $|\mathrm{m}| = \ell(n) \gg |s| = n$
  - Computational Security vs Information Theoretic (Perfect) Security
  - **Disadvantage**: Still can only send one message

Theorem 3.18: If G is a pseudorandom generator then the above encryption scheme has indistinguishable encryptions in the presence of an eavesdropper.

One-Time-Pads + PRGs

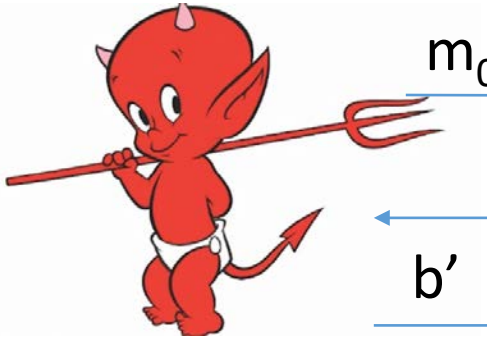
$$\mathrm{Enc}_s(\mathrm{m}) = \mathrm{G}(\mathrm{s}) \oplus m$$
$$\mathrm{Dec}_s(\mathrm{c}) = \mathrm{G}(\mathrm{s}) \oplus c$$

Theorem 3.18: If G is a pseudorandom generator then the above encryption scheme has indistinguishable encryptions in the presence of an eavesdropper.

Proof by Reduction: Start with and attacker A that breaks security of encryption scheme and transform A into distinguisher D that breaks PRG security of G.

Why is this sufficient?

Breaking Semantic Security



m_0, m_1

$c = G(s) \oplus m_b$

b'

Random bit b

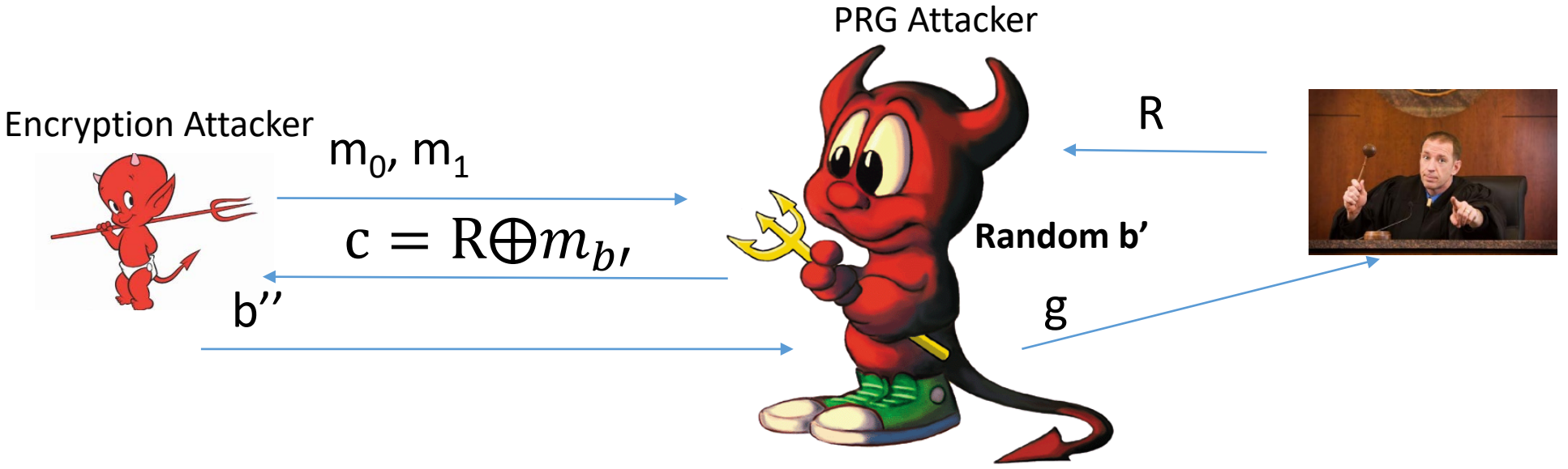
Random seed s

ppt attacker

non − negligible function
(possibly still small)

$$\Pr\left[\text{  } \; Guesses\ b' = b\right] \geq \frac{1}{2} + f(n)$$

The Reduction



Random bit b

If b=1

$r \leftarrow \{0,1\}^n$

R = G(r)

Else

$R \leftarrow \{0,1\}^{\ell(n)}$

- What is $\Pr[b'' \neq b' | b=0]$?
  - Hint: What encryption scheme is used?
- What is $\Pr[b'' = b' | b=1]$?

g = 1 if b”=b’

0 otherwise

Analysis

$$\left|\Pr_{s\in\{0,1\}^n}\left[D\left(G(s)\right)=1\right]-\Pr_{R\in\{0,1\}^{\ell(n)}}\left[D(R)=1\right]\right|$$
$$=\left|\Pr[\text{b''}=\text{b'}|\text{b=1}]-\Pr[\text{b''}\neq\text{b'}|\text{b=0}]\right|$$
$$=\left|\Pr[\text{b''}=\text{b'}|\text{b=1}]-½\right|$$
$$\geq ½ + \text{f(n)} - ½ \geq \text{f(n)}$$

Recall: f(n) was (non-negligible) advantage of encryption attacker.

Implication: PRG G is also insecure (contrary to assumption).

QED

One-Time-Pads + PRGs

- Encryption:
  - Secret key is the seed (K=s)

$$\mathrm{Enc}_s(\mathrm{m}) = \mathrm{G}(\mathrm{s}) \oplus m$$
$$\mathrm{Dec}_s(\mathrm{c}) = \mathrm{G}(\mathrm{s}) \oplus c$$

  - **Advantage**: $|\mathrm{m}| = \ell(n) \gg |s| = n$
  - Computational Security vs Information Theoretic (Perfect) Security
  - **Disadvantages**: can only send one message, no message integrity vs. active attacker

Theorem (Concrete Security): If G is a $\left(t(n), \varepsilon(n)\right)$-secure PRG then the above encryption scheme is $\left(t'(n) = t(n) - O(n), \varepsilon(n)\right)$-semantically secure.

Proof Idea: Use the same reduction. If encryption attacker runs in time $t'(n)$ then our PRG attacker runs in time $t(n)$. If encryption attacker wins with probability $\varepsilon(n)$ then our PRG attacker wins the PRG game with the same probability.

Candidate PRG

- **Notation:** Given string $x \in \{0,1\}^n$ and a subset $S \subset \{1, \ldots, n\}$ let $x_S \in \{0,1\}^{|S|}$ denote the substring formed by concatenating bits at the positions in S.
- **Example**: x=10110 and S = {1,4,5} x_S=110

$$P(x_1, x_2, x_3, x_4, x_5) = x_1 + x_2 + x_3 + x_4 x_5 \bmod 2$$

- Select random subsets $\mathbb{S} = S_1, \ldots, S_{\ell(n)} \subset \{1, \ldots, n\}$ of size $|S_i|=5$ and with $\ell(n) = n^{1.4}$

$$G_{\mathbb{S}}(x) = P\left(x_{S_1}\right) \circ \cdots \circ P\left(x_{S_{\ell(n)}}\right)$$

Stream Cipher vs PRG

- PRG pseudorandom bits output all at once

- Stream Cipher
  - Pseudorandom bits can be output as a stream
  - RC4, RC5 (Ron's Code)

st_0 := Init(s)

For i=1 to ℓ:

$(y_i, st_i) := GetBits(st_{i-1})$

Output: $y_1, \ldots, y_\ell$

The RC4 Stream Cipher

- A proprietary cipher owned by RSA, designed by Ron Rivest in 1987.
- Became public in 1994.
- Simple and effective design.
- Variable key size (typical 40 to 256 bits),
- Output unbounded number of bytes.
- Widely used (web SSL/TLS, wireless WEP).
- Extensively studied, not a completely secure PRNG when used correctly, ~~no known attacks exist~~
- **Newer Versions**: RC5 and RC6
- **Rijndael** selected by NIST as AES in 2000

The RC4 Cipher

- The cipher internal state consists of
  - a 256-byte array S, which contains a permutation of 0 to 255
    - total number of possible states is 256! ≈ 2^{1700}
  - two indexes: i, j

```
i = j = 0
Loop
   i = (i + 1) (mod 256)
   j = (j + S[i]) (mod 256)
   swap(S[i], S[j])
   output S[S[i] + S[j]] (mod 256)
End Loop
```

Limitations of Current Security Definition

- Assumes adversary observes just one ciphertext

- What if adversary observes two ciphertexts?

$$c_1 = \mathrm{Enc}_s(m_1) = \mathrm{G}(s) \oplus m_1$$
$$c_2 = \mathrm{Enc}_s(m_2) = \mathrm{G}(s) \oplus m_2$$

- How could the adversary (Joe) attempt to modify $c=\mathrm{Enc}_k(m)$ below?

m = “Pay Joe the following amount (USD): 000000101”

Limitations of Current Security Definition

- Assumes adversary observes just one ciphertext

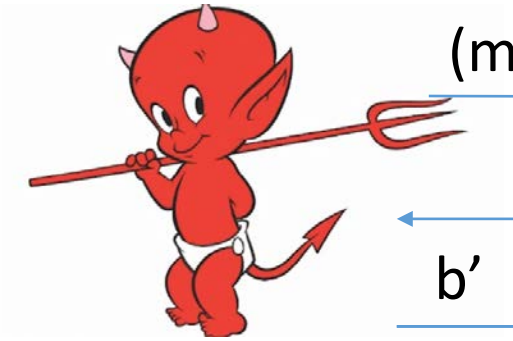
- What if adversary observes two ciphertexts?

$$c_1 = \mathrm{Enc}_s(m_1) = \mathrm{G}(s) \oplus m_1$$
$$c_2 = \mathrm{Enc}_s(m_2) = \mathrm{G}(s) \oplus m_2$$

- How could the adversary (Joe) attempt to modify $c=\mathrm{Enc}_k(m)$ below?

m = “Pay Joe the following amount (USD): **1**00000101”

Multiple Message Eavesdropping Experiment



$(m_{0,1},\ldots,m_{0,t})$, $(m_{1,1},\ldots,m_{1,t})$

$(c_1,\ldots,c_t)$

b’

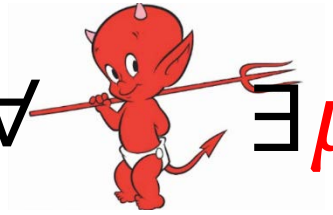
Random bit b

K = Gen(.)

$c_i = Enc_K(m_{b,i})$

ppt attacker

negligible function

$$\forall \text{  } \exists \mu \text{  } \Pr\left[\text{  } \; Guesses\ b' = b\right] \leq \frac{1}{2} + \mu \text{  }(n)$$

Multiple Message Eavesdropping Experimen

$Formally, let\ \Pi = (Gen, Enc, Dec)\ denote\ the\ encryption\ scheme\ and\ define\ a\ random\ variable$

$$PrivK_{A,\Pi}^{mult}(1^n) = \begin{cases} 1 & \text{if } b = b' \\ 0 & \text{otherwise} \end{cases}$$

$\Pi\ has\ indistinguishable\ multiple\ encryptions\ in\ the\ presence\ of\ an\ eavesdropper\ if\ for\ all\ PPT\ adversary\ A, there\ is\ a$

Negligible function μ such that $\Pr[PrivK_{A,\Pi}^{mult}(1^n) = 1] \leq \frac{1}{2} + \mu(n)$

m bit b
n(.)
$c_K(m_b)$

Multiple vs Single Encryptions

If Π has *indistinguishable multiple encryptions* in the presence of an eavesdropper

then

Π also has *indistinguishable encryptions* in the presence of an eavesdropper.

Question: Are the definitions equivalent?

- **Answer:** No, *indistinguishable multiple encryptions* is a strictly stronger security notion.

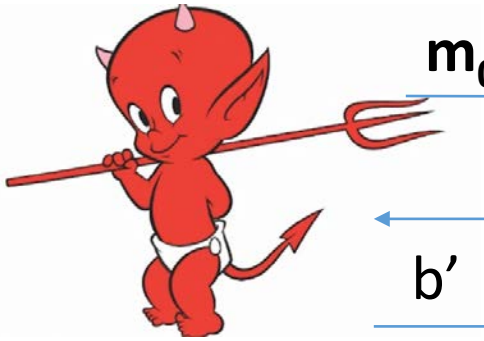
Example

$$\mathrm{Enc}_s(\mathrm{m}) = \mathrm{G}(\mathrm{s}) \oplus m$$
$$\mathrm{Dec}_s(\mathrm{c}) = \mathrm{G}(\mathrm{s}) \oplus c$$

Recall: $\Pi = (Gen, Enc, Dec)$ has **indistinguishable encryptions** in the presence of an eavesdropper.

Claim: $\Pi = (Gen, Enc, Dec)$ does **not** have **indistinguishable multiple encryptions** in the presence of an eavesdropper.

Multiple Message Eavesdropping Attack



$\mathbf{m_0}=(0^{\ell(n)}, 0^{\ell(n)})$, $\mathbf{m_1}=(0^{\ell(n)}, 1^{\ell(n)})$

$(c_1 = \mathrm{G}(s)\oplus \mathbf{m_{b,1}}, c_2 = \mathrm{G}(s)\oplus \mathbf{m_{b,2}})$

b'

Random bit b

$s \leftarrow \mathrm{Gen}(1^n)$

$c_i = \mathrm{Enc}_K(m_{b,i})$

$$b' = \begin{cases} 1 & \text{if } c_1 \neq c_2 \\ 0 & otherwise \end{cases}$$

Analysis: If b=0 then $c_1 = \mathrm{G}(s)\oplus 0^{\ell(n)} = c_2$

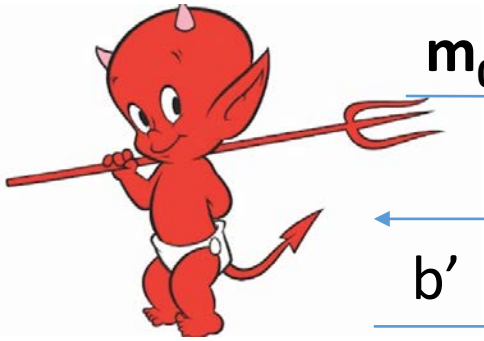
Analysis: If b=1 then $c_1 = \mathrm{G}(s)\oplus 0^{\ell(n)} \neq \mathrm{G}(s)\oplus 1^{\ell(n)} = c_2$

Did We Cheat?

- Attack specifically exploited the fact that we can ask to see multiple encryptions of the same message…
- The above argument might appear to show that no encryption scheme provides secure **indistinguishable multiple encryptions** in the presence of an eavesdropper.

Theorem: If Π is (stateless) encryption scheme and Enc is deterministic then Π does **not provide** secure **indistinguishable multiple encryptions**

Multiple Message Eavesdropping



$\mathbf{m_0}=(0^{\ell(n)}, 0^{\ell(n)})$, $\mathbf{m_1}=(0^{\ell(n)}, 1^{\ell(n)})$

$\left(c_1 = \mathrm{Enc}_K\ (\mathbf{m_{b,1}}), c_2 = \mathrm{Enc}_K\ (\mathbf{m_{b,2}})\right)$

b'

Random bit b

$s \leftarrow \mathrm{Gen}(1^n)$

$c_i = \mathrm{Enc}_K(m_{b,i})$

$$b' = \begin{cases} 1 & \text{if } c_1 \neq c_2 \\ 0 & otherwise \end{cases}$$

Analysis: If b=0 then $c_1 = \mathrm{Enc}_K(0^{\ell(n)}) = c_2$

Analysis: If b=1 then $c_1 = \mathrm{Enc}_K(0^{\ell(n)}) \neq \mathrm{Enc}_K(1^{\ell(n)}) = c_2$

Where to go from here?

Option 1: Weaken the security definition so that attacker cannot request two encryptions of the same message.

- Undesirable!
- **Example:** Dataset in which many people have the last name “Smith”
- We will actually want to strengthen the definition later…

Option 2: Consider randomized encryption algorithms

Week 2: Topic 3: CPA-Security

Chosen-Plaintext Attacks

- Model ability of adversary to control or influence what the honest parties encrypt.

- During World War 2 the British placed mines at specific locations, knowing that the Germans, upon finding the mines, would encrypt the location and send them back to headquarters. The encrypted messages helped cryptanalysts at Bletchley Park to break the German encryption scheme.

Chosen-Plaintext Attacks

- Model ability of adversary to control or influence what the honest parties encrypt.

- Battle of Midway (WWII). US Navy cryptanalysts intercept and encrypted message which they are able to partially decode (May 1942).
  - The message stated that the Japanese were planning an attack on AF?
  - Cryptanalysts could not decode ciphertext fragment AF.
  - Best Guess: AF = “Midway Island.”

Battle of Midway

From Wikipedia, the free encyclopedia

Coordinates: 28°12′N 177°21′W

This article is about the 1942 battle. For other uses, see The Battle of Midway (disambiguation).

The **Battle of Midway** was a decisive naval battle in the Pacific Theater of World War II.[6][7][8] Between 4 and 7 June 1942, only six months after Japan's attack on Pearl Harbor and one month after the Battle of the Coral Sea, the United States Navy under Admirals Chester Nimitz, Frank Jack Fletcher, and Raymond A. Spruance decisively defeated an attacking fleet of the Imperial Japanese Navy under Admirals Isoroku Yamamoto, Chuichi Nagumo, and Nobutake Kondo near Midway Atoll, inflicting devastating damage on the Japanese fleet that proved irreparable. Military historian John Keegan called it "the most stunning and decisive blow in the history of naval warfare."[9]

Battle of Midway

Part of the Pacific Theater of World War II

U.S. Douglas SBD-3 Dauntless dive bombers from USS *Hornet* about to attack the burning Japanese

Battle of Midway

From Wikipedia, the free encyclopedia

Coordinates: 28°12′N 177°21′W

This article is about the 1942 battle. For other uses, see The Battle of Midway (disambiguation).

The **Battle of Midway** was a decisive naval battle in the Pacific Theater of World War II.[6][7][8] Between 4 and 7 June 1942, only six months after Japan's attack on Pearl Harbor and one month after the Battle of the Coral Sea, the United States Navy under Admirals Chester Nimitz, Frank Jack Fletcher, and Raymond A. Spruance decisively defeated an attacking fleet of the Imperial Japanese Navy under Admirals Isoroku Yamamoto, Chuichi Nagumo, and Nobutake Kondo near Midway Atoll, inflicting devastating damage on the Japanese fleet that proved irreparable. Military historian John Keegan called it "the most stunning and decisive blow in the history of naval warfare."[9]

Battle of Midway

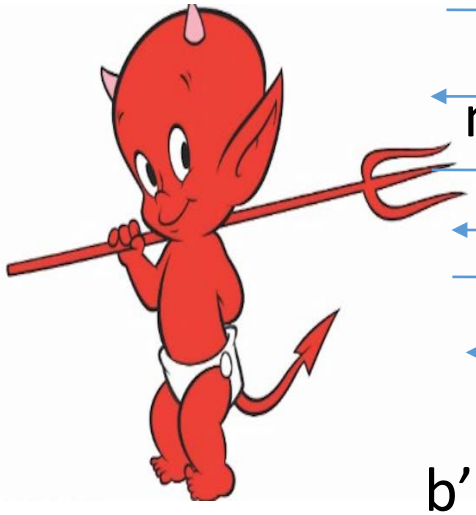
Part of the Pacific Theater of World War II

U.S. Douglas SBD-3 Dauntless dive bombers from USS *Hornet* about to attack the burning Japanese

Multiple Message Security and CPA-Attacks

- Multiple Message Security
  - Attacker must select all messages at the same time.
  - Significant Limitation!
- In the WWII attacks cryptanalysts selected the message adaptively
  - Selected message(s) to encrypt *after* observing target ciphertext

CPA-Security (Single Message)



m_0, m_1

$c = Enc_K(m_b)$

m_2

$c_2 = Enc_K(m_2)$

m_3

$c_3 = Enc_K(m_3)$

⋮

b’

Random bit b

K ← Gen(1ⁿ)

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr[A\ \ Guesses\ b' = b] \leq \frac{1}{2} + \mu(n)$$

CPA-Security (Single Message)

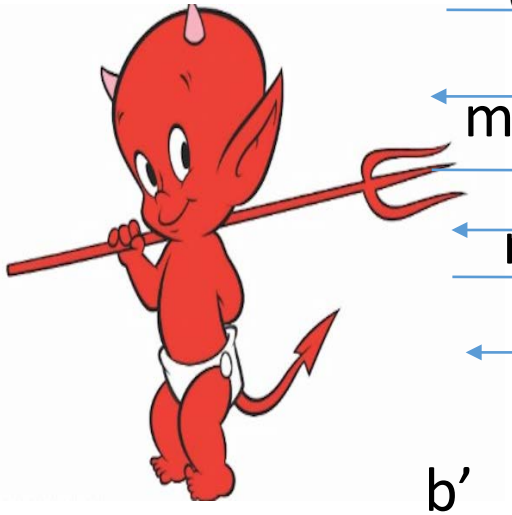
Formally, let $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ denote the encryption scheme, and define a random variable $\text{PrivK}^{\text{cpa}}_{\text{A},\Pi}(1^n)$

$$\text{PrivK}^{\text{cpa}}_{\text{A},\Pi}(1^n) = \begin{cases} 1 & \text{if } b = b' \\ 0 & \text{otherwise} \end{cases}$$

Π *has* indistinguishable encryptions under a chosen plaintext attack if for all PPT adversaries A, there is a negligible function μ such that

$$\Pr\left[\text{PrivK}^{\text{cpa}}_{\text{A},\Pi}(1^n) = 1\right] \leq \frac{1}{2} + \mu(n)$$

CPA-Security (Multiple Messages)



$m_{0,1}, m_{1,1}$

$c_1 = Enc_K(m_{b,1})$

$m_{0,2}, m_{1,2}$

$c_2 = Enc_K(m_{b,2})$

$m_{0,3}, m_{1,3}$

$c_3 = Enc_K(m_{b,3})$

⋮

b’

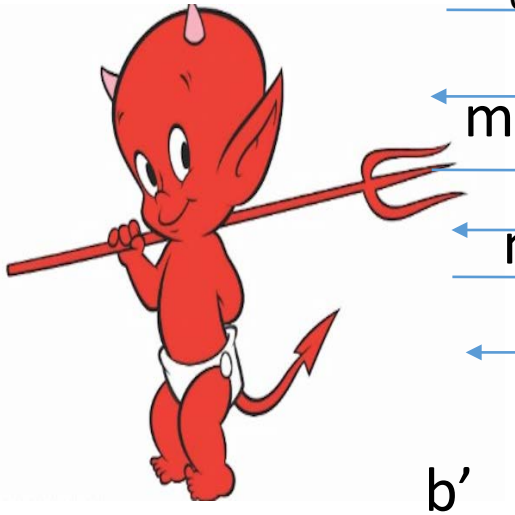
Random bit b

K ← Gen(1^n)

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr\left[PrivK_{A,\Pi}^{LR-CPA}(1^n)\right] \leq \frac{1}{2} + \mu(n)$$

CPA-Security (Multiple Messages)



$m_{0,1}, m_{1,1}$

$c_1 = Enc_K(m_{b,1})$

$m_{0,2}, m_{1,2}$

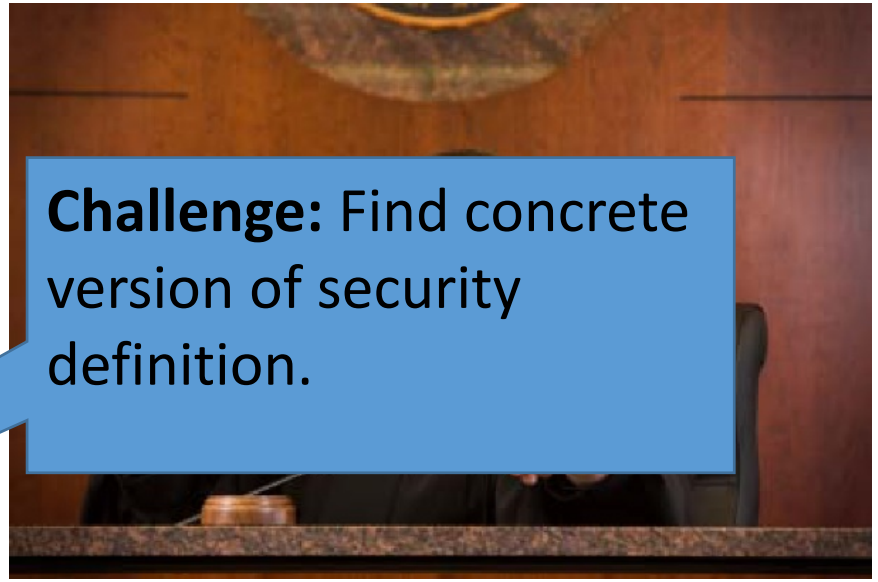
$c_2 = Enc_K(m_{b,2})$

$m_{0,3}, m_{1,3}$

$c_3 = Enc_K(m_{b,3})$

⋮

b’



Challenge: Find concrete version of security definition.

Random bit b

K ← Gen(1ⁿ)

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr\left[PrivK_{A,\Pi}^{LR-CPA}(1^n)\right] \leq \frac{1}{2} + \mu(n)$$

CPA-Security

Theorem: An encryption scheme $\Pi = (Gen, Enc, Dec)$ that is CPA-Secure for single encryptions is also CPA-secure for multiple encryptions.

- We will simply say CPA-security for simplicity
- To show CPA-Security it suffices to show CPA-security for single encryptions.
- To reason about security of a protocol using Π we can use game with multiple encryptions.

CPA-Security

- CPA-security vs Multiple Message Encryption
  - CPA-security is stronger guarantee
  - Attacker can select messages adaptively
- CPA-security: minimal security notion for a modern cryptosystem

- Limitations of CPA-Security: Does not model and adversary who
  - Attempts to modify messages
  - Can get honest party to (partially) decrypt some messages

CPA-Security and Message Length

Observation: Given a CPA-secure encryption scheme $\Pi = (Gen, Enc, Dec)$ that supports single bit messages ($\mathcal{M} = \{0,1\}$) it is easy to build a CPA-secure scheme $\Pi' = (Gen', Enc', Dec')$ that supports messages $m = m_1,\ldots,m_n \in \{0,1\}^n$ of length n.

$$\mathrm{Enc}'_k(m) = \langle \mathrm{Enc}_k(m_1), \ldots, \mathrm{Enc}_k(m_n) \rangle$$

Exercise: How would you prove Π' is CPA-secure?

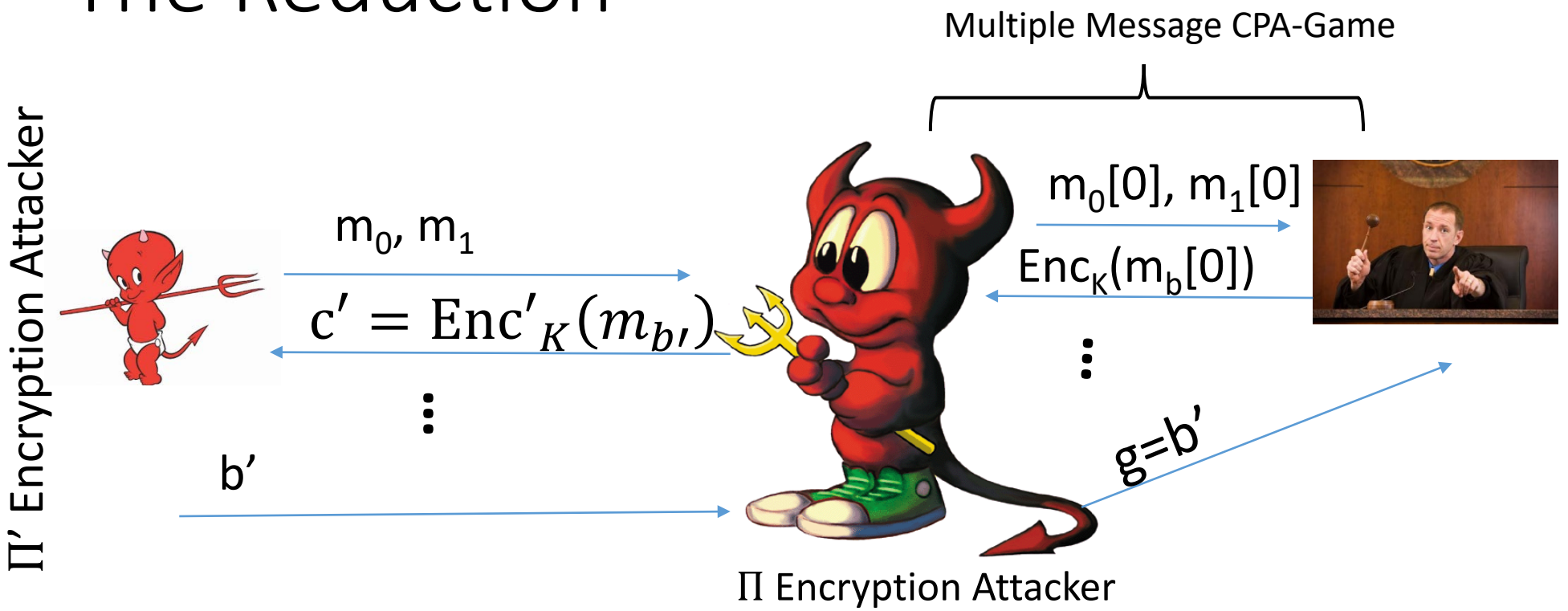
Security Reduction

- **Step 1:** Assume for contraction that we have a PPT attacker A that breaks CPA-Security.
- **Step 2:** Construct a PPT distinguisher D which breaks PRF security.

The Reduction

Random bit b

K ← Gen(1^n)



$$\mathrm{Enc}'_{\mathrm{k}}(m) = \langle \mathrm{Enc}_{\mathrm{k}}\ (m_1), \ldots, \mathrm{Enc}_{\mathrm{k}}\ (m_n) \rangle$$

Week 2: Topic 4: Pseudorandom Functions and CPA-Security

Pseudorandom Function (PRF)

A keyed function $\mathrm{F}: \{0,1\}^{\ell_{key}(n)} \times \{0,1\}^{\ell_{in}(n)} \rightarrow \{0,1\}^{\ell_{out}(n)}$, which "looks random" without the secret key k.

- $\ell_{key}(n)$ - length of secret key k
- $\ell_{in}(n)$ - length of input
- $\ell_{out}(n)$ - length of output

- Typically, $\ell_{key}(n)$=$\ell_{in}(n)$=$\ell_{out}(n)$ =n (unless otherwise specified)

- Computing $F_K(x)$ is efficient (polynomial-time)

PRF vs. PRG

- Pseudorandom Generator G is not a keyed function

- PRG Security Model: Attacker sees only output G(r)
  - Attacker who sees r can easily distinguish G(r) from random
- PRF Security Model: Attacker sees both inputs and outputs $(r_i, F_k(r_i))$
  - In fact, attacker can select inputs r_i
  - Attacker Goal: distinguish F from a truly random function

Truly Random Function

- Let **Func$_n$** denote the set of all functions $f: \{0,1\}^n \to \{0,1\}^n$.

- **Question**: How big is the set **Func$_n$**?
- **Hint**: Consider the lookup table.
  - 2^n entries in lookup table
  - n bits per entry ($f(x)$)
  - $n2^n$ bits to encode f∈**Func$_n$**

2^n

x	$f(x)$
$0 \dots 00$	$f(0 \dots 00)$
$0 \dots 01$	$f(0 \dots 01)$
$0 \dots 10$	$f(0 \dots 10)$
…	…
$1 \dots 11$	$f(1 \dots 11)$

- **Answer**: $|\mathbf{Func_n}| = 2^{n2^n}$ (by comparison only $|\mathcal{K}| = 2^n$ n-bit keys)

Truly Random Function

- Let $\mathbf{Func_n}$ denote the set of all functions $f: \{0,1\}^n \to \{0,1\}^n$.

- Can view entries in lookup table as populated in advance (uniformly)
  - **Space:** $n2^n$ bits to encode $f \in \mathbf{Func_n}$
- Alternatively, can view entries as populated uniformly "on-the-fly"
  - **Space:** $2n \times q(n)$ bits after $q(n)$ queries to store prior responses
- Alternate view is often useful in security reductions
  - Doesn't require time to fully specify $f \in \mathbf{Func_n}$

Oracle Notation

- We use $A^{f(.)}$ to denote an algorithm A with oracle access to a function f.
- A may adaptively query f(.) on multiple different inputs $x_1, x_2, \ldots$ and A receives the answers $f(x_1), f(x_2), \ldots$
- However, A can only use f(.) as a blackbox (no peaking at the source code in the box)

PRF Security

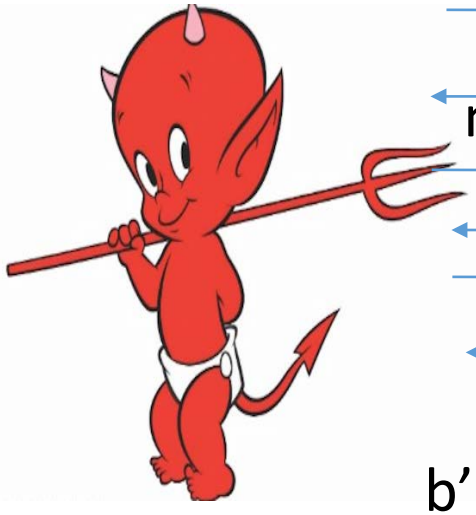
Definition 3.25: A keyed function $\mathrm{F}: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a pseudorandom function if for all PPT distinguishers D there is a negligible function μ s.t.

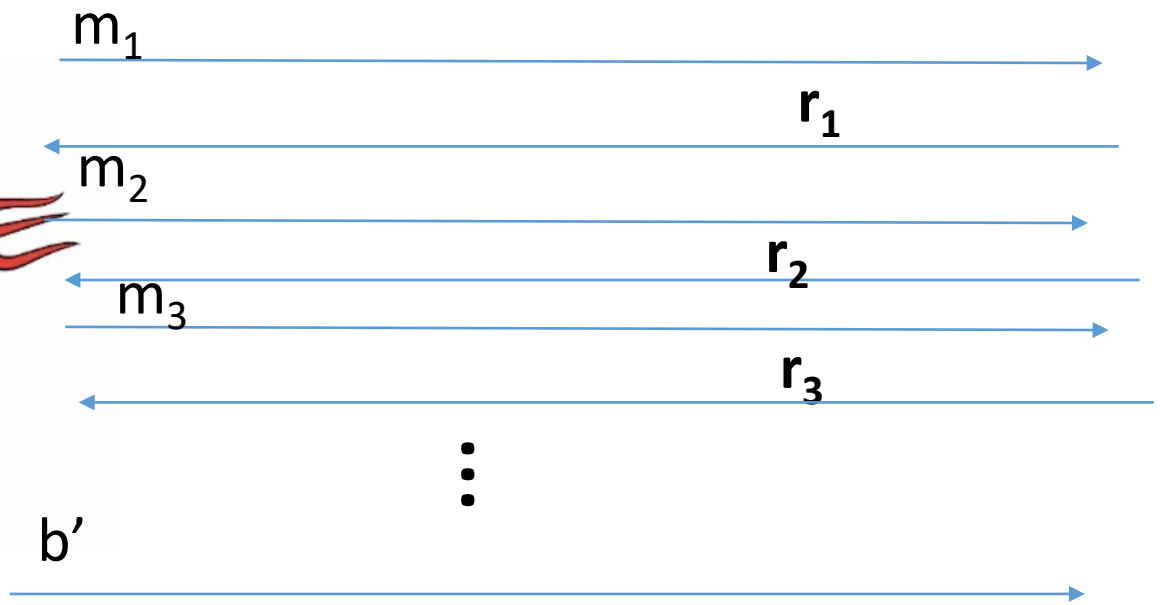
$$\left|Pr\left[D^{F_k(\cdot)}(1^n)\right] - Pr\left[D^{f(\cdot)}(1^n)\right]\right| \leq \mu(n)$$

Notes:

- the first probability is taken over the uniform choice of $k \in \{0,1\}^n$ as well as the randomness of D.
- the second probability is taken over uniform choice of f ∈**Func**$_n$as well as the randomness of D.
- D is *not* given the secret k in the first probability (otherwise easy to distinguish…how?)

PRF-Security as a Game





$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr[A\ \ Guesses\ b' = b] \leq \frac{1}{2} + \mu(n)$$

Random bit b

$\mathbf{K} \leftarrow \mathbf{Gen}(1^n)$

Truly random func R

$r_i = F_K(m_i)$ if b=1

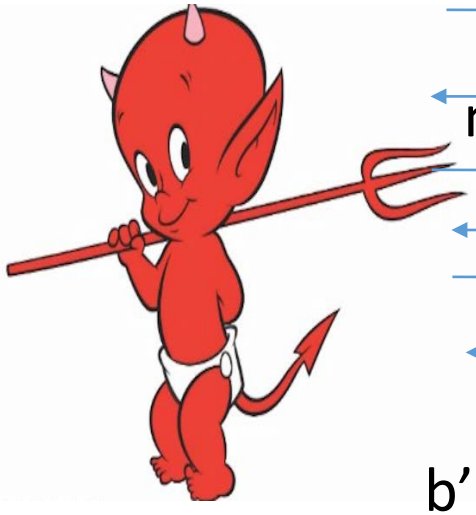
$R(m_i)$ o.w

PRF Security Concrete Version

Definition 3.25: A keyed function $\mathrm{F}: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a $\big(t(n), q(n), \varepsilon(n)\big)$-*secure pseudorandom function* if **for all** distinguishers D **running in time at most $\boldsymbol{t(n)}$** and **making at most $\boldsymbol{q(n)}$ queries** we have

$$\left|Pr\left[D^{F_k(\cdot)}(1^n)\right] - Pr\left[D^{f(\cdot)}(1^n)\right]\right| \leq \varepsilon(n)$$

Reminder: CPA-Security (Single Message)



m_0, m_1

$c = Enc_K(m_b)$

m_2

$c_2 = Enc_K(m_2)$

m_3

$c_3 = Enc_K(m_3)$

⋮

b’

Random bit b

$\mathbf{K \leftarrow Gen(1^n)}$

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$

$$\Pr[A\ \ Guesses\ b' = b] \leq \frac{1}{2} + \mu(n)$$

CPA-Secure Encryption

- Gen: on input 1^n pick uniform $k \in \{0,1\}^n$
- Enc: Input $k \in \{0,1\}^n$ and $m \in \{0,1\}^n$

  Output $c = \langle r, F_k(r) \oplus m \rangle$ for uniform $r \in \{0,1\}^n$

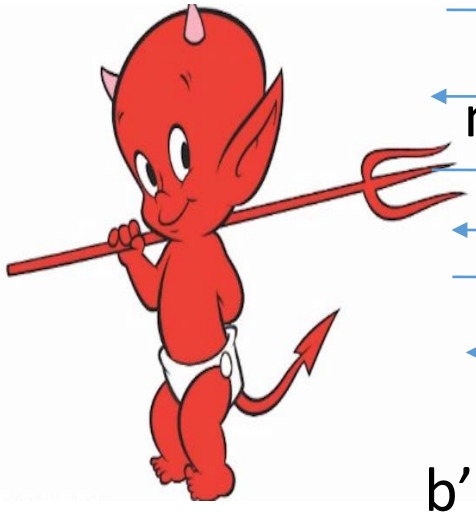
- Dec: Input $k \in \{0,1\}^n$ and $c = \langle r, s \rangle$

  Output $m = F_k(r) \oplus s$

How to begin proof?

Theorem: If F is a pseudorandom function, then (Gen,Enc,Dec) is a CPA-secure encryption scheme for messages of length n.

Breaking CPA-Security (Single Message)



m_0, m_1

$\langle r, F_k(r) \oplus m_b \rangle$

m_2

$\langle r_2, F_k(r_2) \oplus m_2 \rangle$

m_3

$\langle r_3, F_k(r_3) \oplus m_3 \rangle$

$\vdots$

b'

Random bit b

$\mathbf{K} \leftarrow \mathbf{Gen}(\mathbf{1}^n)$

Assumption: $\exists$ PPT A, P (non $-$ negligible) s. t

$$\Pr[A \ \ Guesses\ b' = b] \geq \frac{1}{2} + P(n)$$

Security Reduction

- **Step 1:** Assume for contraction that we have a PPT attacker A that breaks CPA-Security.
- **Step 2:** Construct a PPT distinguisher D which breaks PRF security.
- Distinguisher D^O (oracle O --- either f or F_k)
  - Simulate A
  - Whenever A queries its encryption oracle on a message m
    - Select random r
    - Return $c = \langle r, O(r) \oplus m \rangle$
  - Whenever A outputs messages m_0, m_1
    - Select random r and bit b
    - Return $c = \langle r, O(r) \oplus m_b \rangle$
  - Whenever A outputs b'
    - Output 1 if b=b'
    - Output 0 otherwise

Analysis: Suppose that O = f then

$$\Pr[D^{F_k} = 1] = \Pr\left[PrivK_{A,\Pi}^{cpa} = 1\right]$$

Suppose that O = f then

$$\Pr\left[D^{f} = 1\right] = \Pr\left[PrivK_{A,\widetilde{\Pi}}^{cpa} = 1\right]$$

where $\widetilde{\Pi}$ denotes the encryption scheme in which F_k is replaced by truly random f.

Security Reduction

- **Step 1:** Assume for contraction that we have a PPT attacker A that breaks CPA-Security.
- **Step 2:** Construct a PPT distinguisher D which breaks PRF security.
- Distinguisher D^O (oracle O --- either f or F_k)
  - Simulate A
  - Whenever A queries its encryption oracle on a message m
    - Select random r
    - Return $c = \langle r, O(r) \oplus m \rangle$
  - Whenever A outputs messages m_0, m_1
    - Select random r and bit b
    - Return $c = \langle r, O(r) \oplus m_b \rangle$
  - Whenever A outputs b'
    - Output 1 if b=b'
    - Output 0 otherwise

Analysis: Suppose that O = F_k then by PRF security, for some negligible function μ, we have

$$\left|\Pr\left[PrivK^{cpa}_{A,\Pi} = 1\right] - \Pr\left[PrivK^{cpa}_{A,\tilde{\Pi}} = 1\right]\right| = \left|\Pr[D^{F_k} = 1] - \Pr[D^{f} = 1]\right| \leq \mu(n)$$

Implies: $\Pr\left[PrivK^{cpa}_{A,\tilde{\Pi}} = 1\right] \geq \Pr\left[PrivK^{cpa}_{A,\Pi} = 1\right] - \mu(n)$

Security Reduction

- **Fact:** $\Pr\left[PrivK^{cpa}_{A,\widetilde{\Pi}} = 1\right] \geq \Pr\left[PrivK^{cpa}_{A,\Pi} = 1\right] - \mu(n)$

- **Claim**: For any attacker A making at most q(n) queries we have

$$\Pr\left[PrivK^{cpa}_{A,\widetilde{\Pi}} = 1\right] \leq \frac{1}{2} + \frac{q(n)}{2^n}$$

Conclusion: For any attacker A making at most q(n) queries we have

$$\Pr\left[PrivK^{cpa}_{A,\Pi} = 1\right] \leq \frac{1}{2} + \frac{q(n)}{2^n} + \mu(n)$$

where $\frac{q(n)}{2^n} + \mu(n)$ is negligible.

Finishing Up

Claim: For any attacker A making at most q(n) queries we have

$$\Pr\left[\mathrm{PrivK}^{cpa}_{A,\tilde{\Pi}} = 1\right] \leq \frac{1}{2} + \frac{q(n)}{2^n}$$

Proof: Let m_0,m_1 denote the challenge messages and let r* denote the random string used to produce the challenge ciphertext

$$c = \langle r^*, f(r^*) \oplus m_b \rangle$$

And let r_1,…,r_q denote the random strings used to produce the other ciphertexts $c_i = \langle r_i, f(r_i) \oplus m_i \rangle$.

If $r^* \neq r_1$,…,$r_q$then then c leaks no information about b (information theoretically).

Finishing Up

Claim: For any attacker A making at most q(n) queries we have

$$\Pr\left[PrivK^{cpa}_{A,\widetilde{\Pi}} = 1\right] \leq \frac{1}{2} + \frac{q(n)}{2^n}$$

Proof: If $r^* \neq r_1,\ldots,r_q$ then then c leaks no information about b (information theoretically). We have

$$\Pr\left[PrivK^{cpa}_{A,\widetilde{\Pi}} = 1\right]$$
$$\leq \Pr\left[PrivK^{cpa}_{A,\widetilde{\Pi}} = 1 \middle| r^* \neq r_1,\ldots,r_q\right] + \Pr\left[r* \in \{r_1,\ldots,r_q\}\right]$$
$$\leq \frac{1}{2} + \frac{q(n)}{2^n}$$

Conclusion

$$\text{Enc}_k(m) = \langle r, F_k(r) \oplus m \rangle$$

$$\text{Dec}_k(\langle r, s \rangle) = F_k(r) \oplus s$$

PRF Security

For any attacker A making at most q(n) queries we have

$$\Pr\left[PrivK^{cpa}_{A,\Pi} = 1\right] \leq \frac{1}{2} + \frac{q(n)}{2^n} + \mu(n)$$

Suggested Exercise: Work out concrete version of security proof

Are PRFs or PRGs more Powerful?

- Easy to construct a secure PRG from a PRF

$$G(s) = F_s(1)|\ldots|F_s(\ell)$$

- Construct a PRF from a PRG?
  - Tricky, but possible… (Katz and Lindell Section 7.5)

PRFs from PRGs

Theorem: Suppose that there is a PRG G with expansion factor $\ell(n) = 2n$. Then there is a secure PRF.

Let $G(x) = G_0(x)||G_1(x)$ (first/last n bits of output)

$$F_K(x_1, \ldots, x_n) = G_{x_n}\left(\ldots\left(G_{x_2}\left(G_{x_1}(K)\right)\right)\ldots\right)$$

Theorem: If G is a PRG then F_k is a PRF

PRFs from PRGs

Theorem: Suppose that there is a PRG G with expansion factor $\ell(n) = 2n$. Then there is a secure PRF.