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

Fall 2017

An Important Remark on Randomness

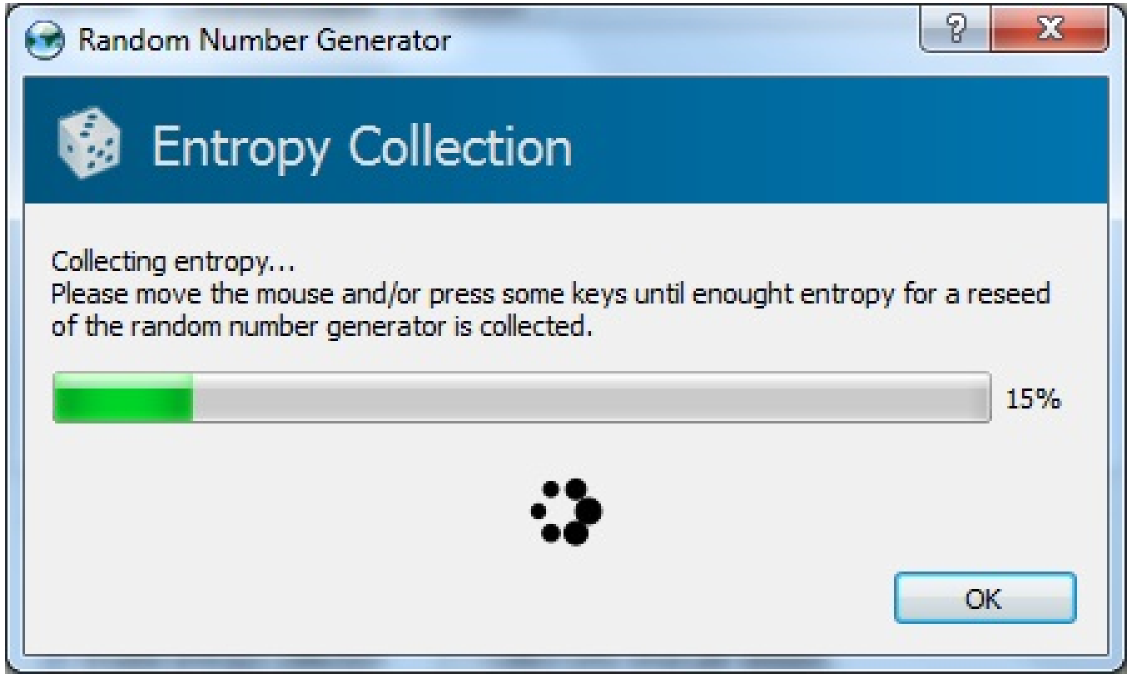
- In our analysis we have made (and will continue to make) a key assumption:

  We have access to true “randomness” to generate a secret key K (e.g. OTP)

- Independent Random Bits
  - Unbiased Coin flips
  - Radioactive decay?

In Practice

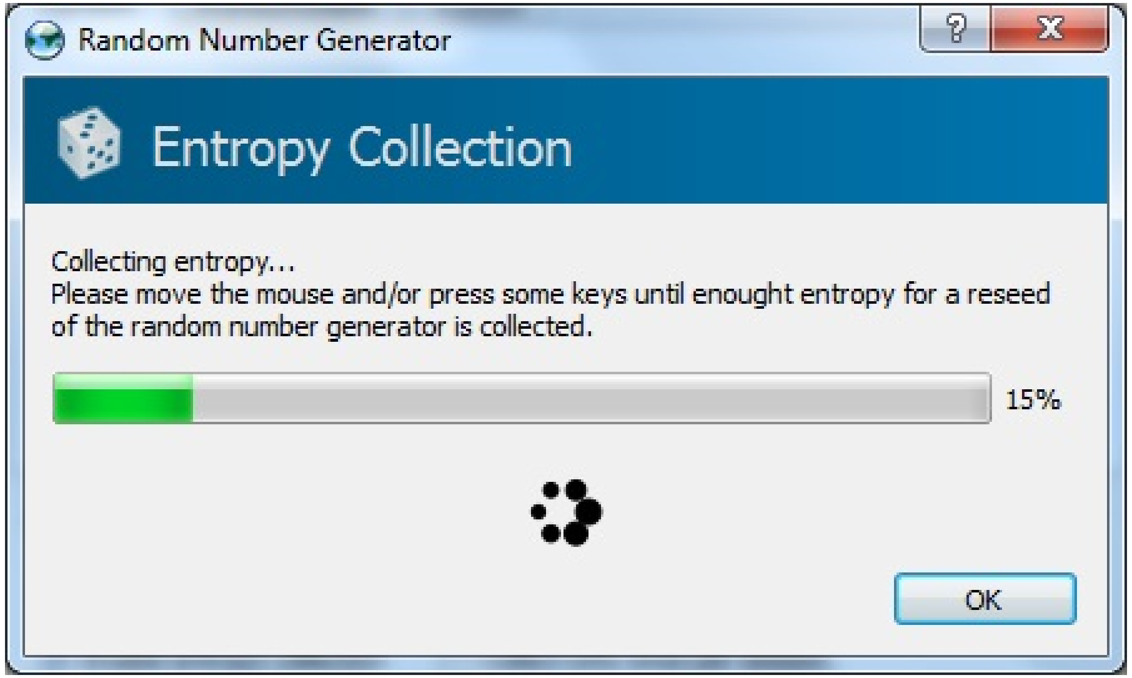
- Hard to flip thousands/millions of coins
- Mouse-movements/keys
  - Uniform bits?
  - Independent bits?
- Use Randomness Extractors
  - As long as input has high entropy, we can extract (almost) uniform/independent bits
  - Hot research topic in theory



Collecting entropy...
Please move the mouse and/or press some keys until enought entropy for a reseed of the random number generator is collected.

In Practice

- Hard to flip thousands/millions of coins
- Mouse-movements/keys
- Customized Randomness Chip?


Random Number Generator

Entropy Collection

Collecting entropy...
Please move the mouse and/or press some keys until enought entropy for a reseed of the random number generator is collected.

Caveat: Don’t do this!

- Rand() in C stdlib.h is no good for cryptographic applications

- Source of many real

world flaws

Week 2: Topic 1: Computational Security

Recap

- Perfect Secrecy, One-time-Pads

Theorem: If (Gen,Enc,Dec) is a perfectly secret encryption scheme then

$$|\mathcal{K}| \geq |\mathcal{M}|$$

What if we want to send a longer message?

K1,K2,K3

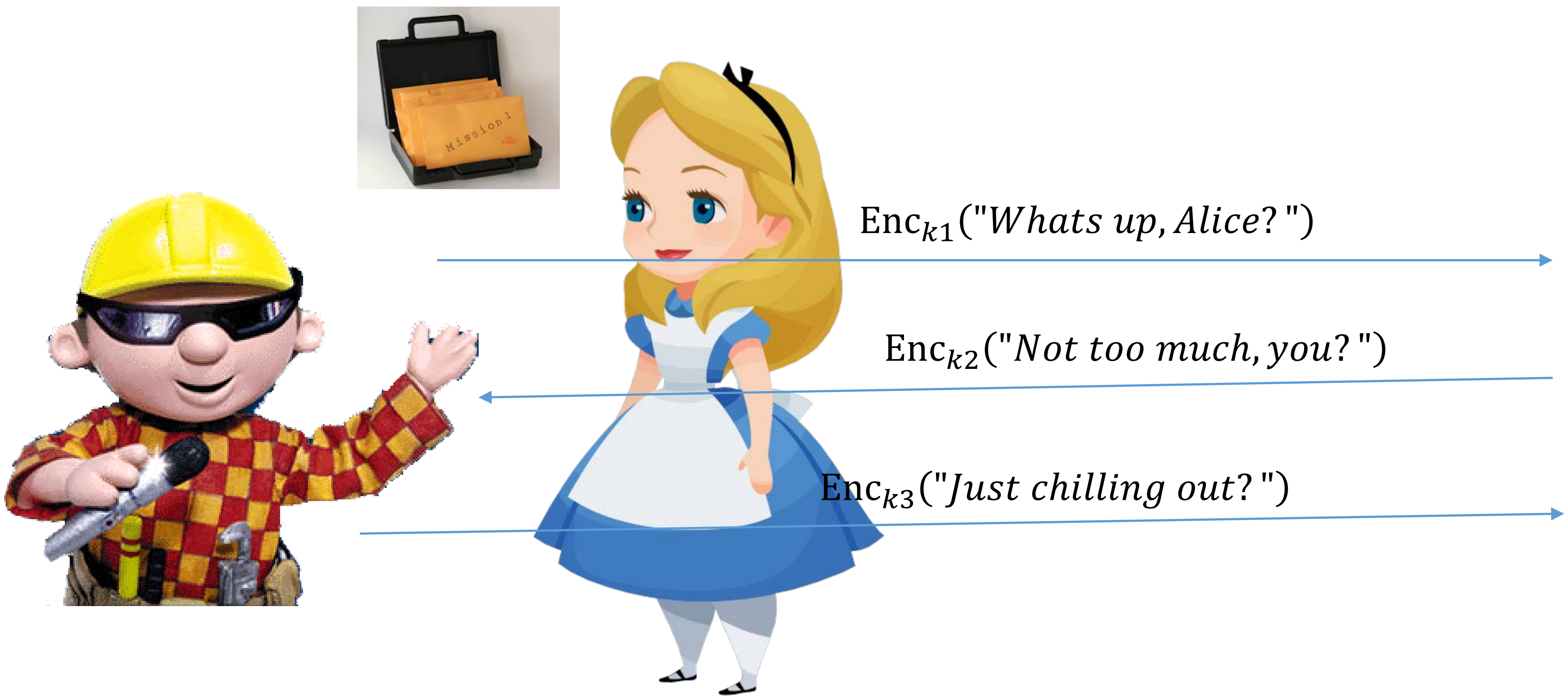
K1,K2,K3

$\text{Enc}_{k1}(\text{"}Dear\ Alice, I\ wrote\ this\ poem\ for\ you\text{"})$

$\text{Enc}_{k2}(\text{"}Roses\ are\ red, \ldots.\text{"})$

$\text{Enc}_{k3}(\text{"}I\ am\ out\ of\ space, but\ the\ rest\ was\ awesome\text{"})$

What if we want to send many messages?



K1,K2,K3

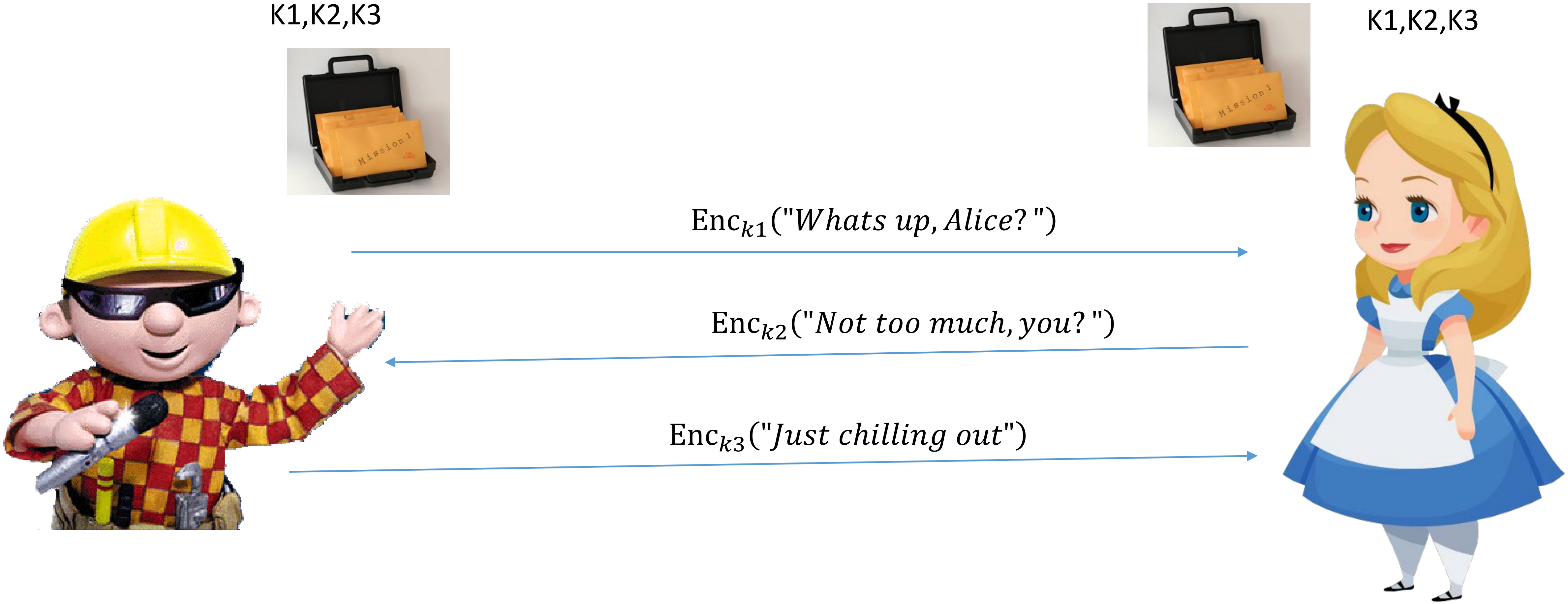
K1,K2,K3

$\mathrm{Enc}_{k1}("Whats\ up, Alice?")$

$\mathrm{Enc}_{k2}("Not\ too\ much, you?")$

$\mathrm{Enc}_{k3}("Just\ chilling\ out?")$

Can we save their relationship?



K1,K2,K3

K1,K2,K3

$\mathrm{Enc}_{k1}("Whats\ up, Alice?")$

$\mathrm{Enc}_{k2}("Not\ too\ much, you?")$

$\mathrm{Enc}_{k3}("Just\ chilling\ out")$

Perfect Secrecy vs Computational Security

- Perfect Secrecy is Information Theoretic
  - Guarantee is independent of attacker resources

- Computational Security
  - Security against computationally bounded attacker
    - An attacker with infinite resources might break security
  - Attacker might succeed with very small probability
    - Example: Lucky guess reveals secret key
    - Very Small Probability: 2^{-100}, 2^{-1000}, …

Current Goal

- Define computational security in presence of eavesdropper who intercepts a single (long) message

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 computational security in the presence of an eavesdropper.~~
- ~~Define computational security against an active attacker who might modify the message~~
- ~~Define computational security for multiple messages in presence of an eavesdropper~~

Concrete Security

> "A scheme is (t,ε)-secure if **every** adversary running for time at most t succeeds in breaking the scheme with probability at most ε"

- Example: $t = 2^{60}$ CPU cycles
  - 9 years on a 4GHz processor
  - < 1 minute on fastest supercomputer (in parallel)
- Full formal definition needs to specify "break"
- Important Metric in Practice
  - **Caveat 1**: difficult to provide/prove such precise statements
  - **Caveat 2**: hardware improves over time

Asymptotic Approach to Security

A scheme is secure if every *probabilistic polynomial time* (ppt) adversary “succeeds” with *negligible* probability.

- Two Key Concepts
  - Polynomial time algorithm
  - Negligible Function

Definition: A function $f\colon \mathbb{N} \longrightarrow \mathbb{R}_{\geq 0}$ is negligible if for every positive polynomial p there is an integer N>0 such that for all n > N we have

$$f(n) < \frac{1}{p(n)}$$

Asymptotic Approach to Security

Definition: A function $f\colon \mathbb{N} \longrightarrow \mathbb{R}_{\geq 0}$ is negligible if for every positive polynomial p there is an integer N>0 such that for all n > N we have

$$f(n) < \frac{1}{p(n)}$$

Intuition: If we choose the security parameter n to be sufficiently large then we can make the adversaries success probability very small (negligibly small).

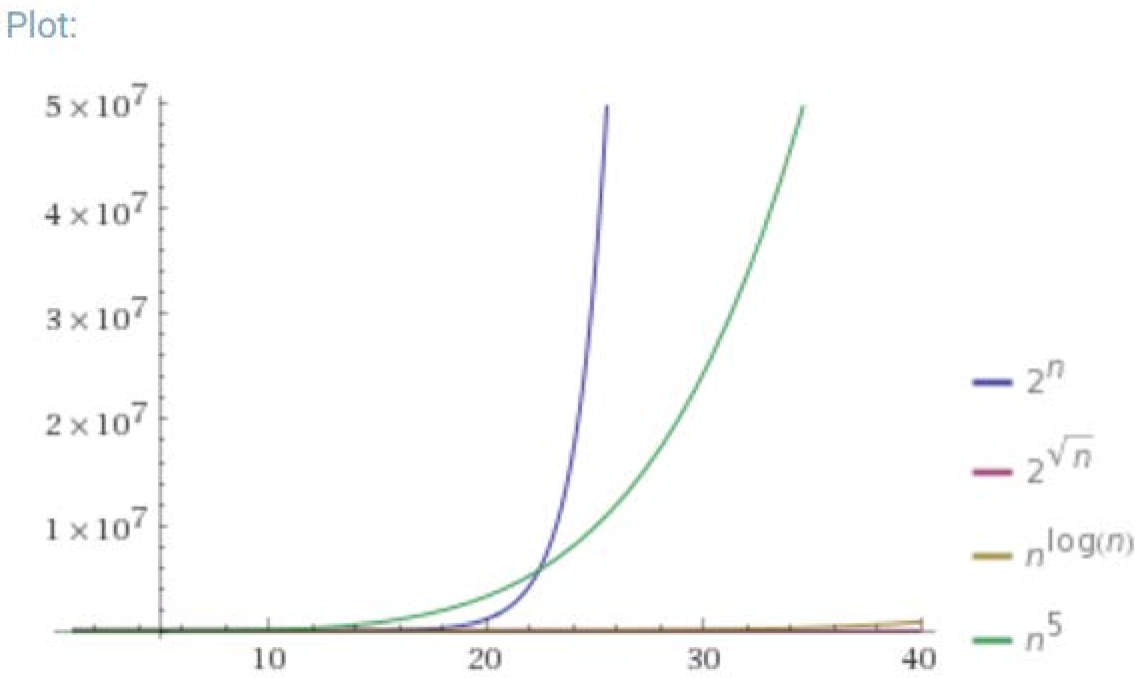
Asymptotic Approach to Security

Definition: A function $f\colon\ \mathbb{N} \longrightarrow \mathbb{R}_{\geq 0}$ is negligible if for every positive polynomial p there is an integer N>0 such that for all n > N we have

$$f(n) < \frac{1}{p(n)}$$

Which functions below are negligible?

- $f(n) = 2^{-n}$
- $f(n) = n^{-3}$
- $f(n) = 2^{-1000}1000n^{1000}$
- $f(n) = 2^{100}2^{-\sqrt{n}}$
- $f(n) = 2^{-\log n}$



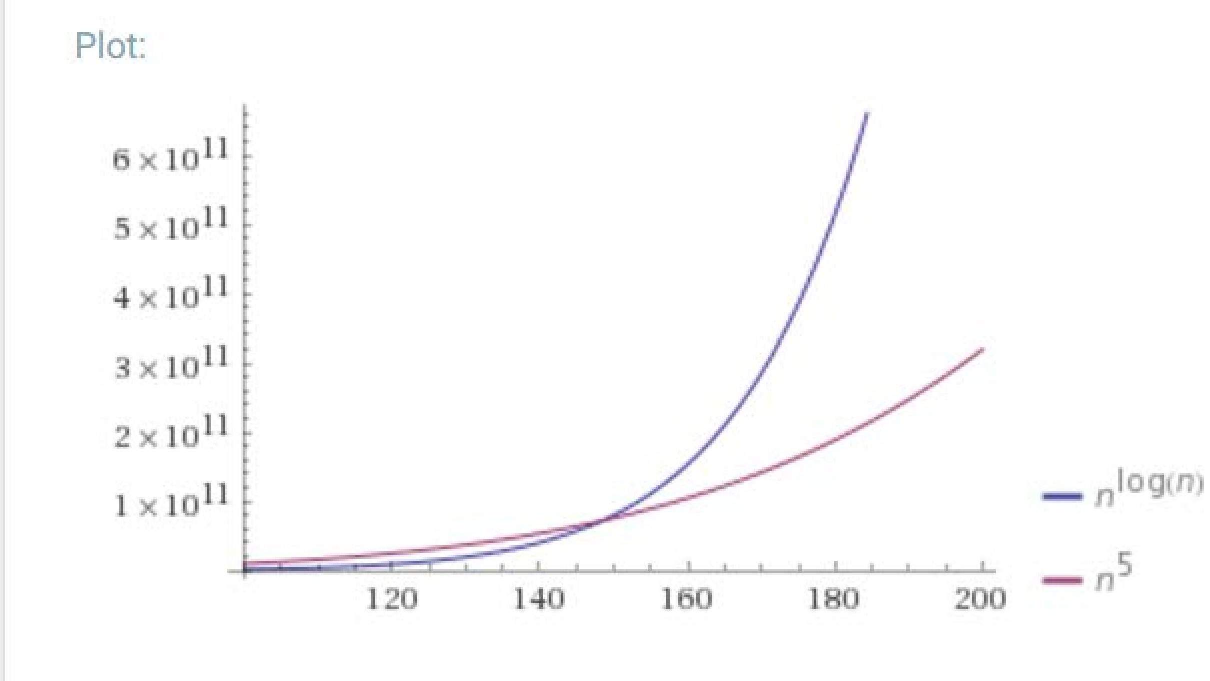
Asymptotic Approach to Security

Definition: A function $f\colon \mathbb{N} \longrightarrow \mathbb{R}_{\geq 0}$ is negligible if for every positive polynomial p there is an integer N>0 such that for all n > N we have

$$f(n) < \frac{1}{p(n)}$$

Which functions below are negligible?

- $f(n) = 2^{-n}$
- $f(n) = n^{-3}$
- $f(n) = 2^{-1000}1000n^{1000}$
- $f(n) = 2^{100}2^{-\sqrt{n}}$
- $f(n) = 2^{-\log n}$



Asymptotic Approach to Security

Definition: An (randomized) algorithm A runs in polynomial time if there exists a polynomial p such that for every n-bit input x, A(x) terminates in at most p(n) steps in expectation.

Intuition: If an algorithm A does not run in polynomial time then, for sufficiently large n, it will quickly become impractical for any attacker to run the algorithm A.

Asymptotic Approach to Security

A scheme is secure if every *probabilistic polynomial time* (ppt) adversary "succeeds" with *negligible* probability.

- **General Attack 1:** Test all possible secret keys $\mathrm{k}' \in \mathcal{K}$
  - Doesn't run in polynomial time, since $|\mathcal{K}| = 2^n$
- **General Attack 2:** Select random key $\mathrm{k}' \in \mathcal{K}$, check if it is correct (otherwise output $\perp$ for "fail").
  - Only successful with negligible probability 2^{-n}

Advantages of Asymptotic Approach

- **Closure**
  - If subroutine B runs in polynomial time and algorithm A makes p(n) queries to B then A also runs in polynomial time.
  - If f and g are negligible functions then h(n) = f(n)+g(n) is a negligible function
  - If p is a positive polynomial, and f is a negligible function then the function g(n)=f(n)p(n) is also negligible.
- **Church-Turing Thesis**: “reasonable” model of computations are all polynomially equivalent.
- **Implication**: No need to worry about different models of computation (circuits, random access machines, etc…)
- **Disadvantage:** Limited guidance on how big to make security parameter n in practice.

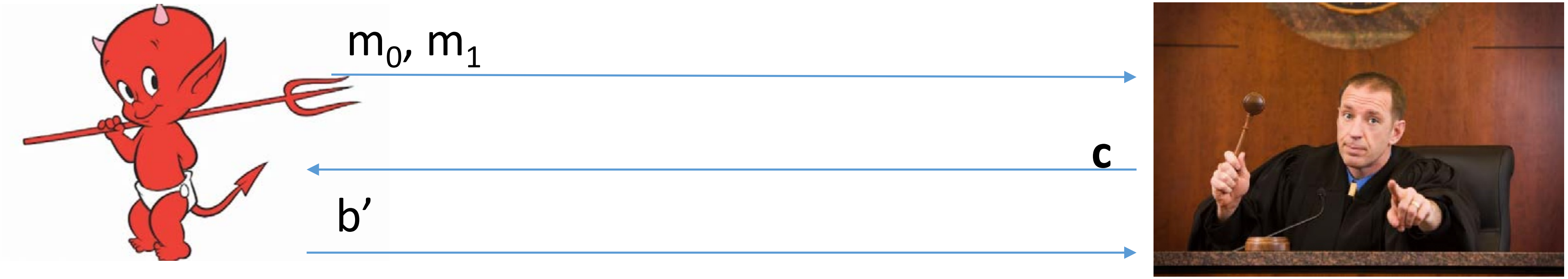
Private Key Encryption Syntax (Revisited)

- Message Space: $\mathcal{M}$
- Key Space: $\mathcal{K}$
- Three Algorithms
  - $\mathrm{Gen}(\mathbf{1^{n}}; R)$ (Key-generation algorithm)
    - Input: **1ⁿ (security parameter in unary)** + Random Bits
    - Output: Secret key $\mathrm{k} \in \mathcal{K}$
  - $\mathrm{Enc_k}(m; \boldsymbol{R})$ (Encryption algorithm)
    - Input: Secret key $\mathrm{k} \in \mathcal{K}$ and message $\mathrm{m} \in \mathcal{M}$ + Randon
    - Output: ciphertext *c*
  - $\mathrm{Dec_k}(c)$ (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”)***

Requirement: all three algorithms run in probabilistic polynomial time

- Invariant: $\mathrm{Dec_k(Enc_k(m))=m}$

Adversarial Indistinguishability Experiment



Random bit b
K = Gen(.)
$c = Enc_K(m_b)$

ppt attacker

negligible function

$\forall$ $\Pr\Big[$ $Guesses\ b' = b\Big] \le \frac{1}{2} + \mu(n)$

Adversarial Indistinguishability Experiment

$Formally, let\ \Pi =\ (Gen, Enc, Dec)\ denote\ the\ encryption\ scheme,$
$call\ the\ experiment\ PrivK_{A,\Pi}^{eav}\ and\ define\ a\ random\ variable$

$$PrivK_{A,\Pi}^{eav} = 1 \quad if\ b = b'$$
$$PrivK_{A,\Pi}^{eav} = 0 \quad otherwise$$

$\Pi\ has\ indistinguishable\ encryptions\ in\ the\ presence\ of$
$an\ eavesdropper\ if\ for\ all\ PPT\ adversary\ A, there\ is\ a$
Negligible function μ such that $\Pr[PrivK_{A,\Pi}^{eav} = 1] \leq \frac{1}{2} + \mu(n)$

m bit b

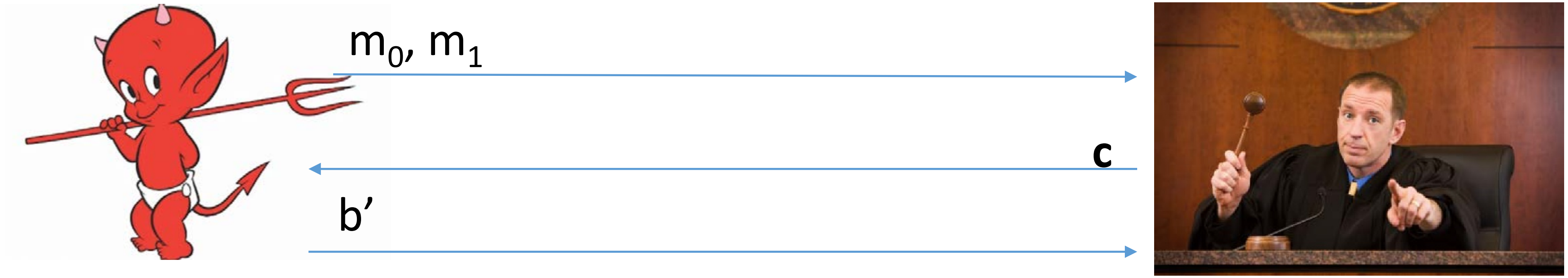
en(.)

$c_K(m_b)$

$[$

$] - 2 \quad (\quad)$

Semantic Security



m_0, m_1

c

b’

Random bit b

K = Gen(.)

c = Enc$_K$(m$_b$)

ppt attacker

negligible function

$\forall$ $\Pr\Big[$ $\mathit{Guesses}\ b' = b\Big] \le \frac{1}{2} + \mu(n)$

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. (CRIME attack on session cookies in HTTPS)

Implications of Indistinguishability

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 $\mathrm{i} \leq \ell$ 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 $\mathrm{K} \leftarrow \mathrm{Gen}(1^n)$, <u>uniform</u> $\mathrm{m} \in \{0,1\}^{\ell}$ and the randomness of Enc and A.

S

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

Definition 3.12: Let (Gen, En fixed-length private key encryption scheme for message of le antically secure if for all PPT attackers A t t for any PPT algorithm Sample all any f and h we have

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

A en get to see an encr on ! Just the length of !

Homework 1 Released Thursday

- Due in class on Thursday, September 14th (2 weeks)

- Solutions should be typeset (preferably in Latex)

- You may collaborate with classmates, but you must write up your own solution and you *must understand* this solution

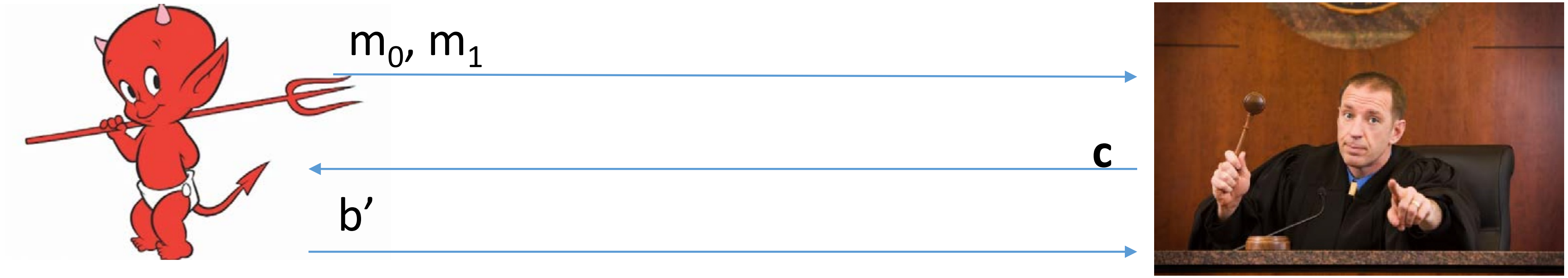
- Clarification questions: fall-2017-cs-55500-le1@lists.purdue.edu

Week 2: Topic 2: Constructing Secure Encryption Schemes

Recap

- Sematic Security/Indistinguishable Encryptions
- Concrete vs Asymptotic Security
  - Negligible Functions
  - Probabilistic Polynomial Time Algorithm

Recap: Semantic Security



m_0, m_1

c

b'

Random bit b

K = Gen(.)

$c = Enc_K(m_b)$

ppt attacker

negligible function

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

S

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

Definition 3.12: Let (Gen, En fixed-length private key encryption scheme for message of le antically secure if for all PPT attackers A t t for any PPT algorithm Sample all any f and h we have

A ven get to see an encry on ! Just the length of !

$$|\Pr[A(1^n, \mathrm{Enc}_K($$

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

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

- 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?

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

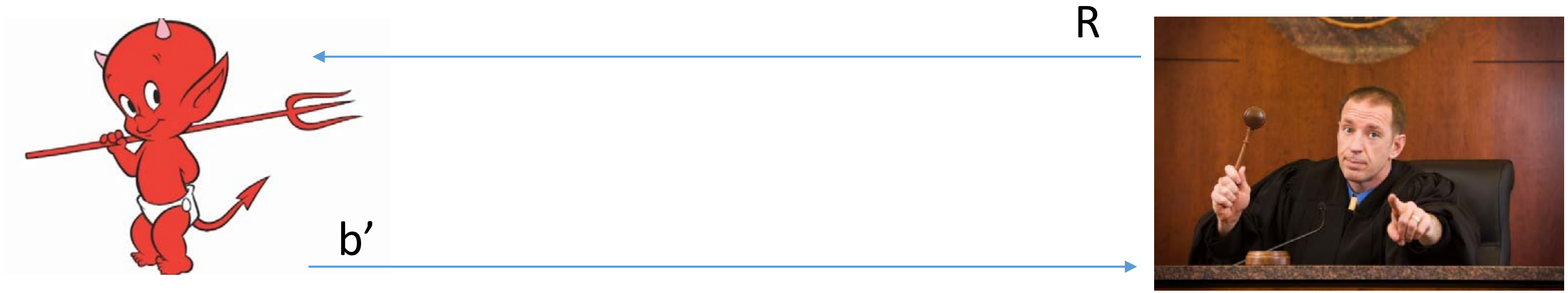
Pseudorandom Generator G

- **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 negl s.t

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

PRG Security as a Game



Random bit b

If b=1

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

R = G(r)

Else

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

ppt attacker

negligible function

$\forall$ $\Pr\Big[$ $Guesses\ b' = b\Big] \le \frac{1}{2} + \mu(n)$

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)

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

    - **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

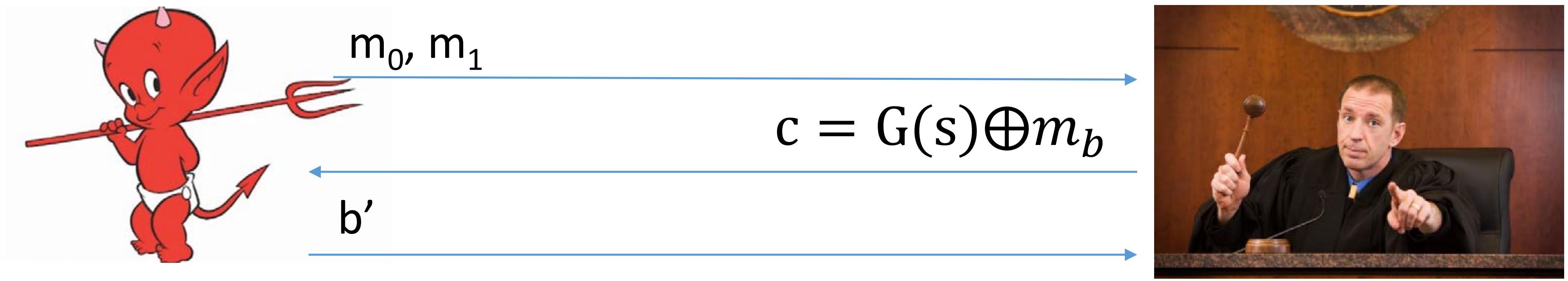
$$\mathrm{Enc}_{\mathrm{s}}(\mathrm{m}) = \mathrm{G}(\mathrm{s}) \oplus m$$
$$\mathrm{Dec}_{\mathrm{s}}(\mathrm{c}) = \mathrm{G}(\mathrm{s}) \oplus c$$

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'

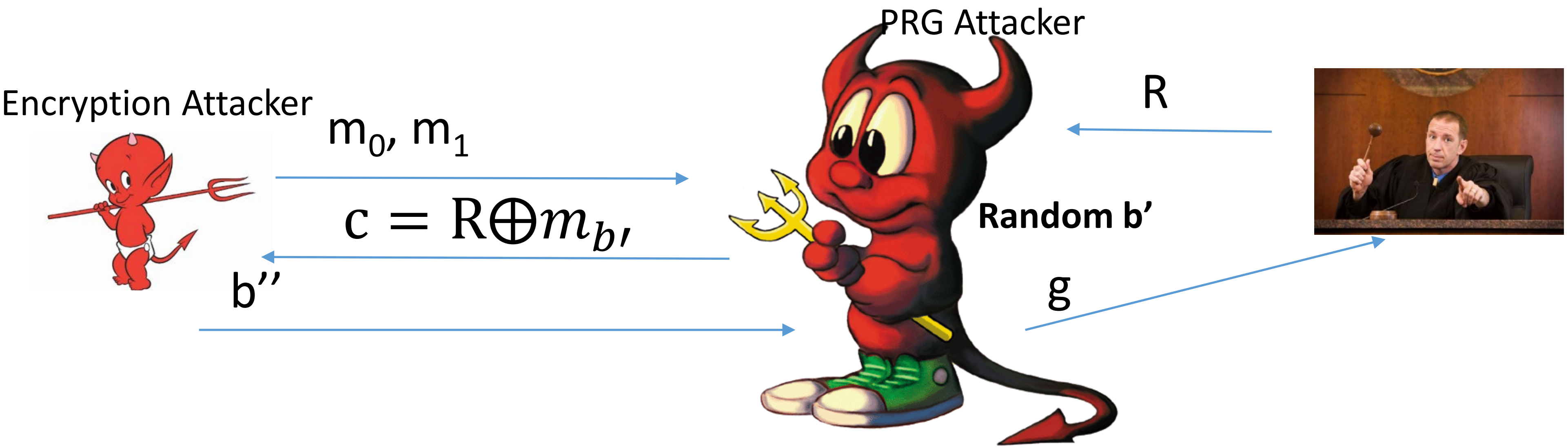
Random bit b
Random seed s

ppt attacker

non − negligible function
(possibly still small)

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

The Reduction



PRG Attacker

Encryption Attacker

m_0, m_1

$c = R \oplus m_{b'}$

b''

R

Random b'

g

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

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

Analysis

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

$$= |\Pr[\text{b''} = \text{b'}|\text{b=1}] - \Pr[\text{b''} \neq \text{b'}|\text{b=0}]|$$

$$= |\Pr[\text{b''} = \text{b'}|\text{b=1}] - ½|$$

$$\geq ½ \ + \text{f(n)} \ - ½ \geq \text{f(n)}$$

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

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

QED

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 $\ell(n) = n^{1.4}$

$$G_{\mathbb{S}}(x) = P\left(x_{S_1}\right) \middle| \ldots \middle| P\left(x_{S_{\ell(n)}}\right)$$

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 $\ell$:
    $(y_i, st_i)$:=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

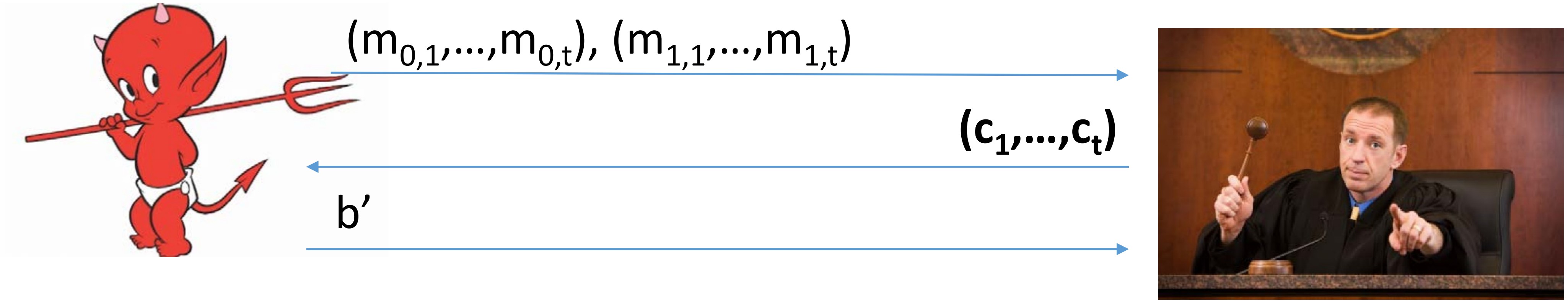
- What if adversary observes two ciphertexts?

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

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

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

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

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

Multiple Message Eavesdropping Experimen

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

$$PrivK^{mult}_{A,\Pi} = 1 \quad if\ b = b'$$
$$PrivK^{mult}_{A,\Pi} = 0 \quad otherwise$$

Π *has indistinguishable multiple encryptions in the presence of an eavesdropper if for all PPT adversary A, there is a* Negligible function μ such that $\Pr[PrivK^{mult}_{A,\Pi} = 1] \leq \frac{1}{2} + \mu(n)$

om bit b

en(.)

$c_K(m_b)$

$[$

$] = 2$

$( )$

A Simple Observation

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

then

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

- In fact ***indistinguishable multiple encryptions*** *is a strictly stronger security notion.*

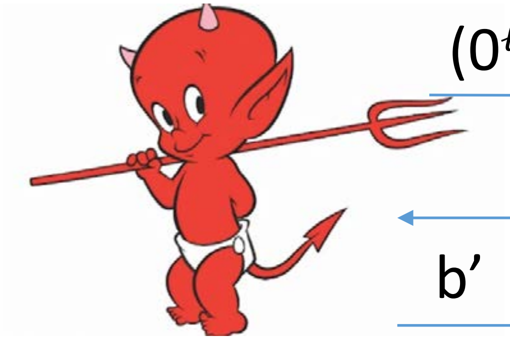
Example

$$\mathrm{Enc_s(m) = G(s)} \oplus m$$
$$\mathrm{Dec_s(c) = G(s)} \oplus c$$

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



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

$(\mathrm{G}(\mathrm{s}) \oplus \mathbf{m}_{\mathbf{b},1}, \mathrm{G}(\mathrm{s}) \oplus \mathbf{m}_{\mathbf{b},2})$

b'

Random bit b

s = Gen(.)

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

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

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

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

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**

Did We Cheat?

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

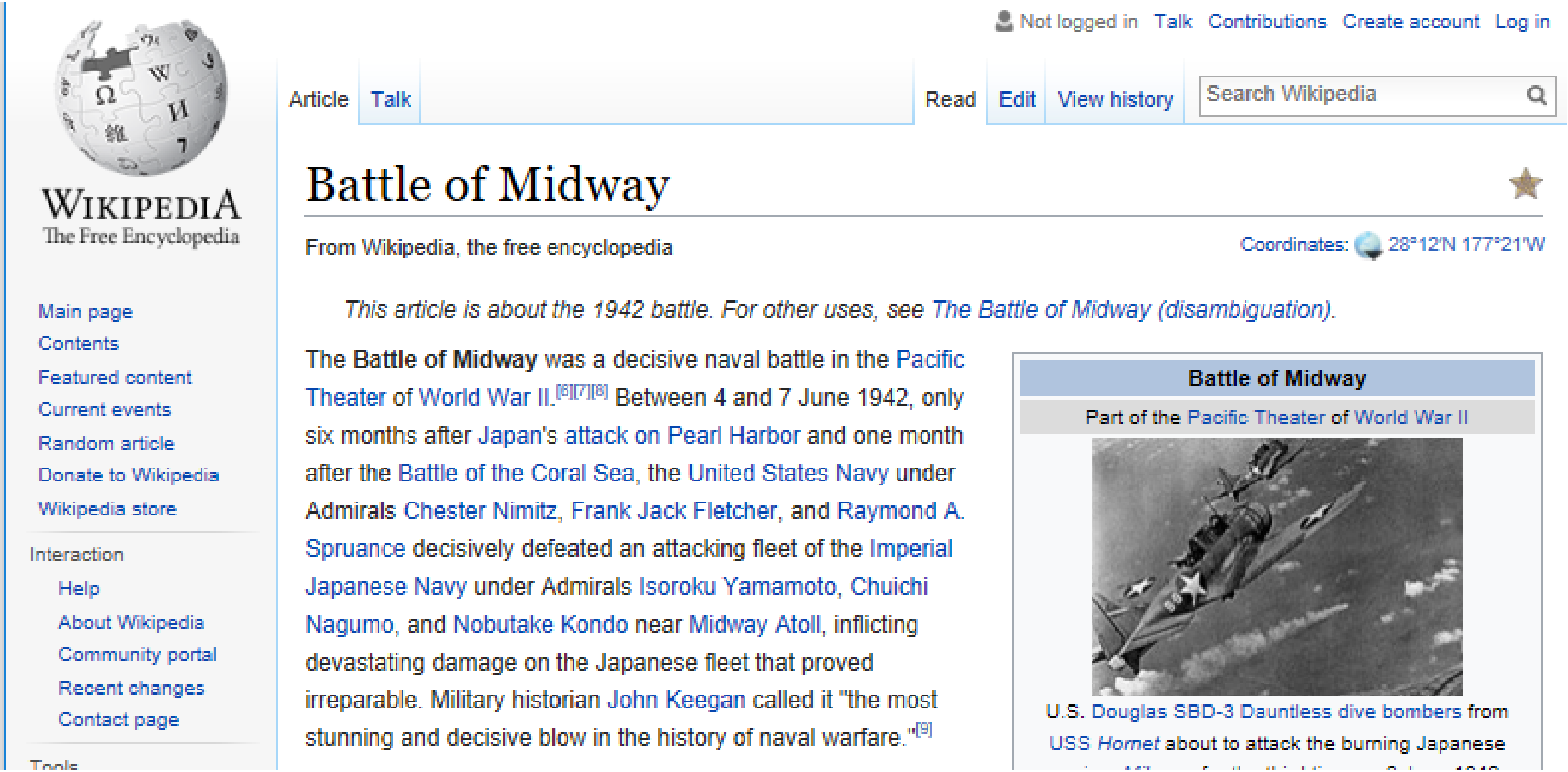
- Model ability of adversary to control or influence what the honest parties encrypt.

- 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

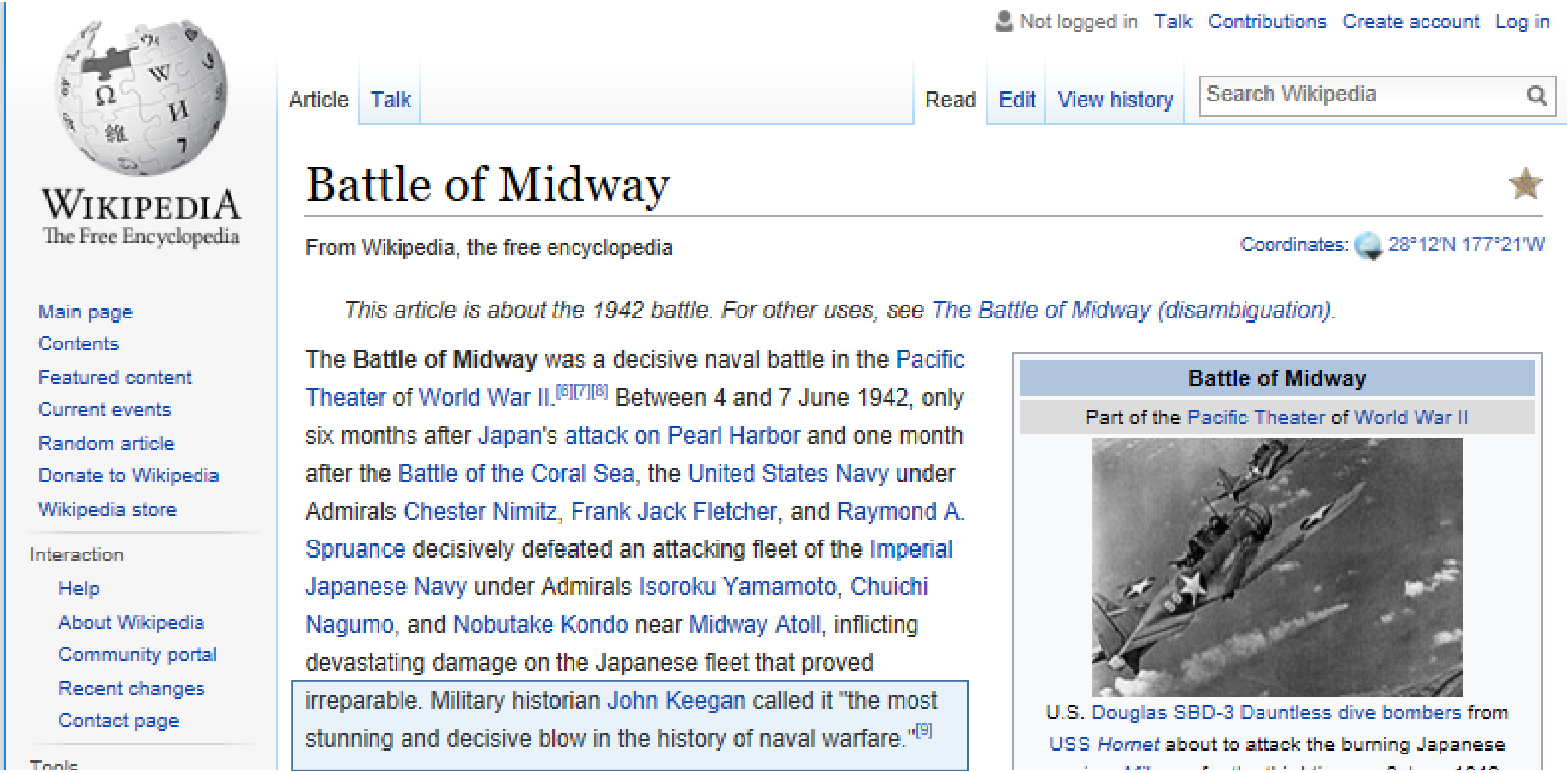
This article is about the 1942 battle. For other uses, see The Battle of Midway (disambiguation).

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

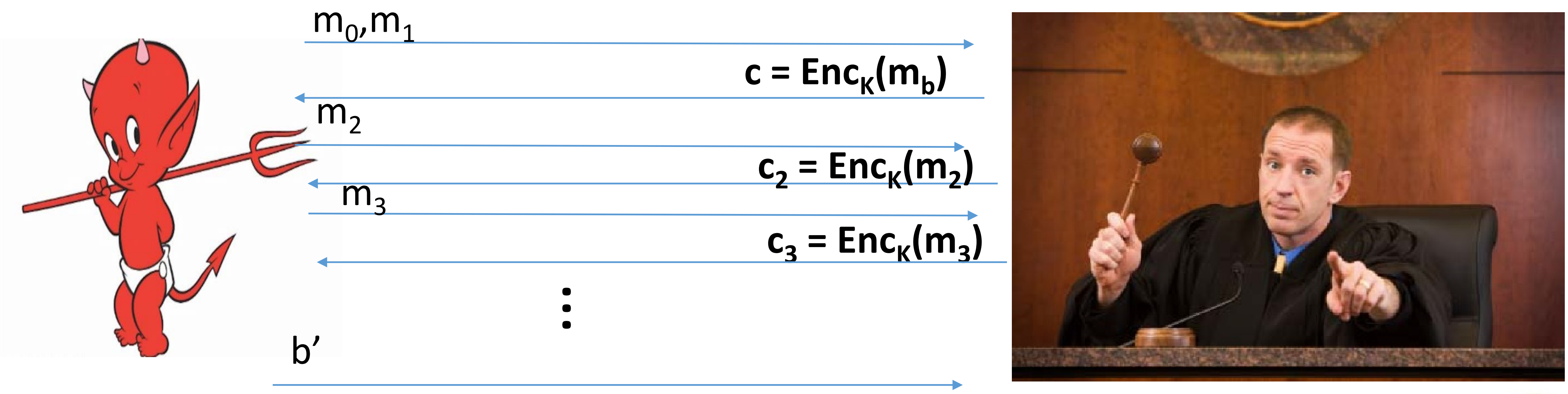
Part of the Pacific Theater of World War II

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(.)

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

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

CPA-Security (Single Message)

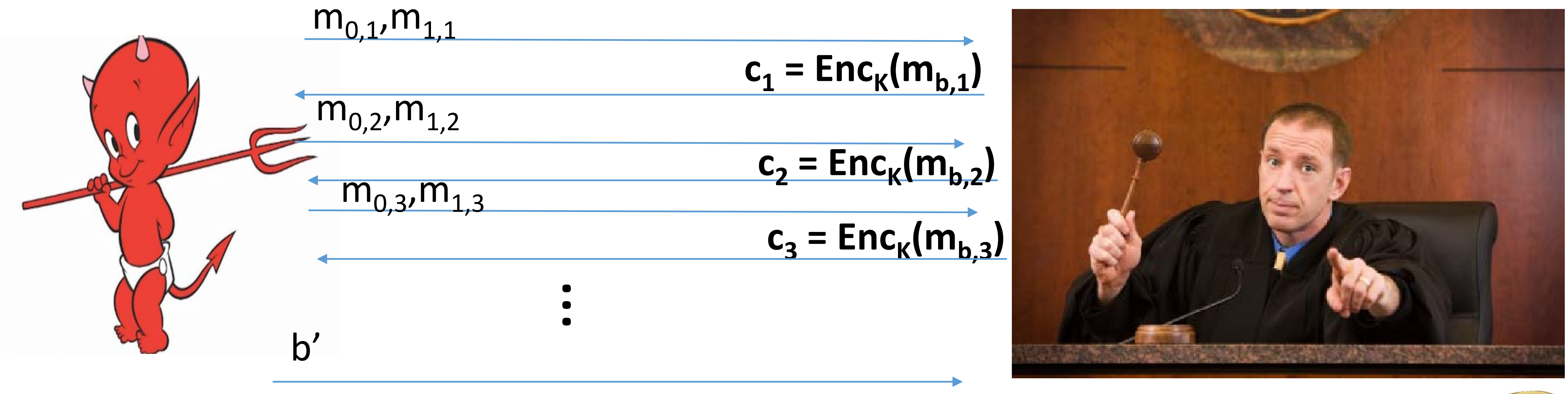
Formally, let $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ denote the encryption scheme, call the experiment $\text{PrivK}_{A,\Pi}^{\text{cpa}}$ and define a random variable

$$\text{PrivK}_{A,\Pi}^{\text{cpa}} = 1 \quad \text{if } b = b'$$
$$\text{PrivK}_{A,\Pi}^{\text{cpa}} = 0 \quad \text{otherwise}$$

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

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

CPA-Security (Multiple Messages)



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

$c_1 = \mathrm{Enc}_K(m_{b,1})$

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

$c_2 = \mathrm{Enc}_K(m_{b,2})$

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

$c_3 = \mathrm{Enc}_K(m_{b,3})$

⋮

b’

Random bit b

K = Gen(.)

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

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

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 messages of a single bit ($\mathcal{M} = \{0,1\}$) it is easy to build a CPA-secure scheme Π'$= (Gen', Enc', Dec')$ that supports messages m = m_1,…,m_n∈ $\{0,1\}^n$ of length n.

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

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.

Next Week

- Read Katz and Lindell 3.5-3.7
- Constructing CPA-Security with Pseudorandom Functions
- Block Cipher Modes of Operation
- CCA-Security (Chosen Ciphertext Attacks)