COS433/Math 473: Cryptography

Mark Zhandry

Princeton University

Spring 2018

Hash Functions

Let $\mathbf{h:\{0,1\}^l \rightarrow \{0,1\}^n}$ be a function, $\mathbf{n \ll l}$

MAC'(k,m) = MAC(k, h(m))
Ver'(k,m,σ) = Ver(k, h(m), σ)

Correctness is straightforward

Security?

- Pigeonhole principle: $\exists\, \mathbf{m_0 \neq m_1 \text{ s.t. } h(m_0)=h(m_1)}$
- But, hopefully such collisions are hard to find

Collision Resistant Hashing

Syntax:

- Key space **K** (typically $\{0,1\}^\lambda$)
- Domain **D** (typically $\{0,1\}^l$ or $\{0,1\}^*$)
- Range **R** (typically $\{0,1\}^n$)
- Function $H: K \times D \rightarrow R$

Correctness: $n \ll l$

Security

Definition: H is **(t,ε)-**collision resistant if, for all  running in time at most **t**,

$$\Pr[H(k,x_0) = H(k,x_1) \wedge x_0 \neq x_1 :$$
$(x_0,x_1) \leftarrow$  $(k), k \leftarrow K] < \varepsilon$

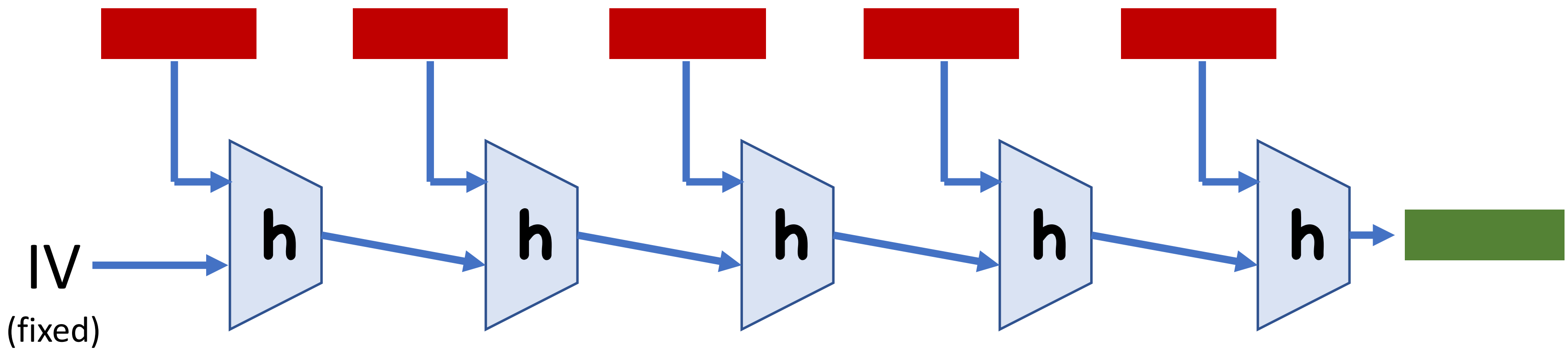
Collision Resistance and MACs

Let $\mathbf{h(m) = H(k,m)}$ for a random choice of $\mathbf{k}$

$$\mathrm{MAC'}(k_{MAC},m) = \mathrm{MAC}(k_{MAC},\ h(m))$$
$$\mathrm{Ver'}(k_{MAC},m,\sigma) = \mathrm{Ver}(k_{MAC},\ h(m),\ \sigma)$$

Think of $\mathbf{k}$ as part of key for $\mathbf{MAC'}$

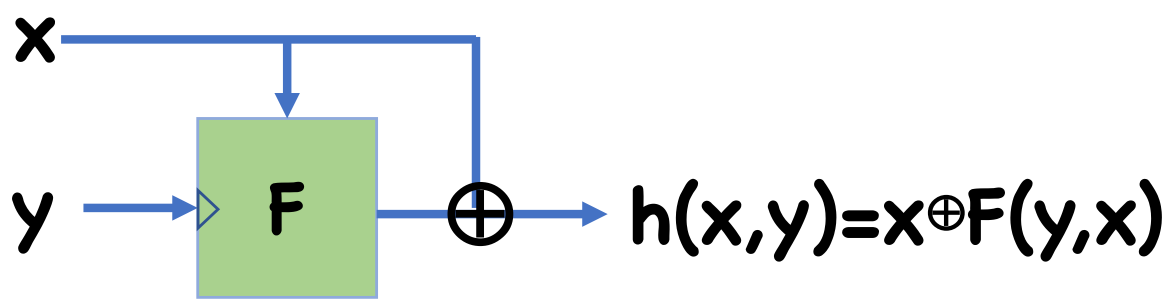
Merkle-Damgard



Constructing **h**

Common approach: use block cipher

Davies-Meyer



Birthday Attack

If the range of a hash function is **R**, a collision can be found in time $T=O(|R|^{1/2})$

Attack:

- Given key **k** for **H**
- For $i=1,\ldots,T$,
  - Choose random x_i in **D**
  - Let $t_i \leftarrow H(k,x_i)$
  - Store pair (x_i, t_i)
- Look for collision amongst stored pairs

Today: Applications of Hashing

Basing MACs on Hash functions

Commitment Schemes

Basing MACs on Hash Functions

Idea: **MAC(k,m) = H(k || m)**

Thought: if **H** is a "good" hash function and **k** is random, should be hard to predict **H(k || m)** without knowing **k**

Unfortunately, cannot prove secure based on just collision resistance of **H**

Random Oracle Model

Pretend **H** is a truly random function

Everyone can query **H** on inputs of their choice
- Any protocol using **H**
- The adversary (since he knows the key)

A query to **H** has a time cost of 1

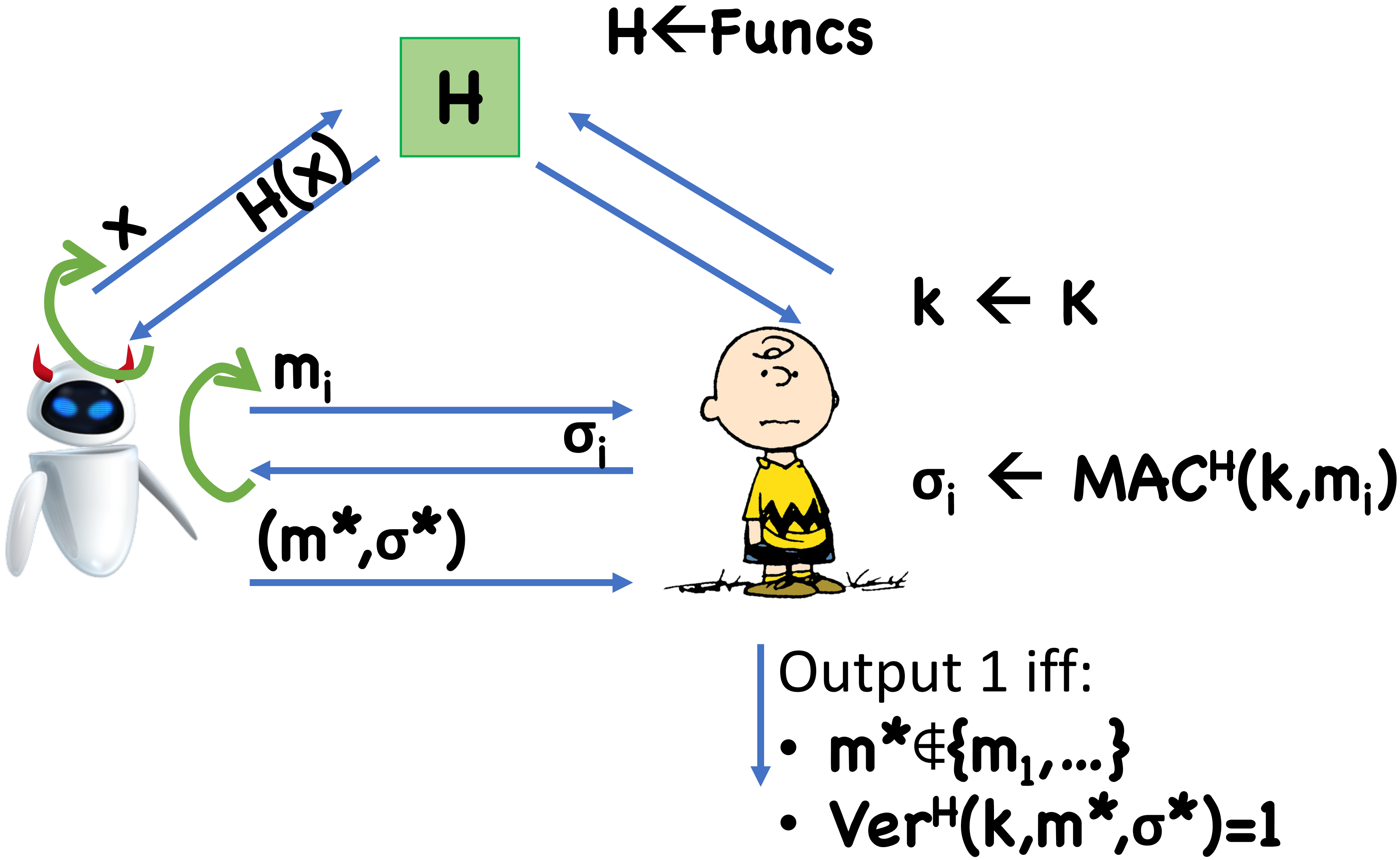
Intuitively captures adversaries that simple query **H**, but don't take advantage of any structure

MAC in ROM

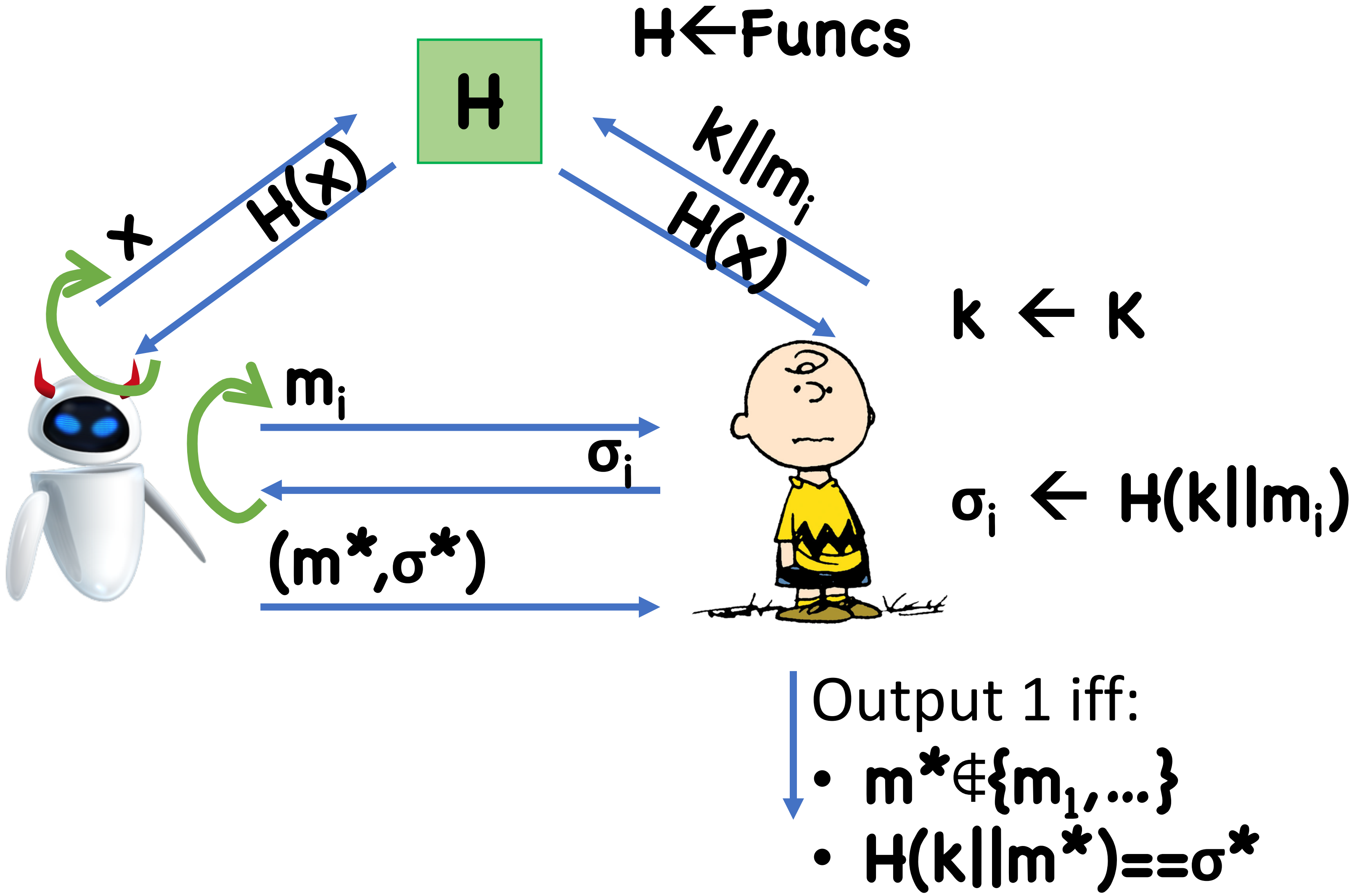
$MAC^H(k,m) = H(k||m)$
$Ver^H(k,m,\sigma) = (H(k||m) == \sigma)$

Theorem: $H(k \,||\, m)$ is a $(t, q, qt/2^n)$–CMA-secure MAC in the random oracle model

Meaning



Meaning



Proof Idea

Value of **H(k||m*)** independent of adversary's view unless she queries **H** on **k||m***

- Only way to forge better than random guessing is to learn **k**

Adversary only sees truly rand and indep **H** values and MACs, unless she queries **H** on $\mathbf{k||m_i}$ for some **i**

- Only way to learn **k** is to query **H** on $\mathbf{k||m_i}$

However, this is very unlikely without knowing **k** in the first place

The ROM

A random oracle is a good

- PRF: $F(k,x) = H(k||x)$

- PRG (assuming H is expanding):
  - Given a random x, $H(x)$ is pseudorandom since adv is unlikely to query H on x

- Collision-resistant hash function:
  - Given poly-many queries, unlikely for find two that map to same output

The ROM

The ROM is very different from security properties like collision resistance

What does it mean that “SHA-2 behaves like a random oracle”?

- No satisfactory definition

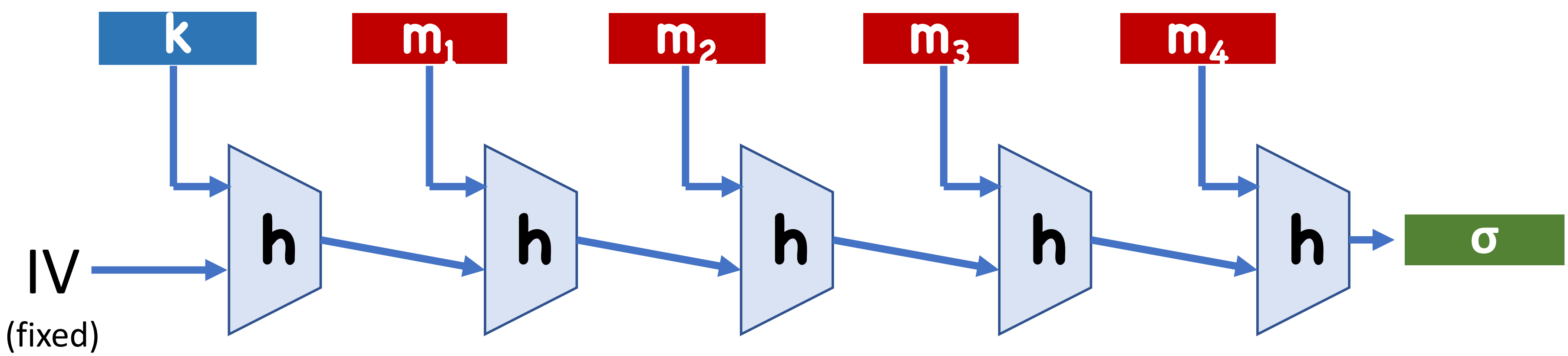
Therefore, a ROM proof is a heuristic argument for security

- If insecure, adversary must be taking advantage of structural weaknesses in H

When the ROM Fails

$MAC^H(k,m) = H(k||m)$
$Ver^H(k,m,\sigma) = (H(k||m) == \sigma)$

Instantiate with Merkle-Damgard (variable length)?



When the ROM Fails

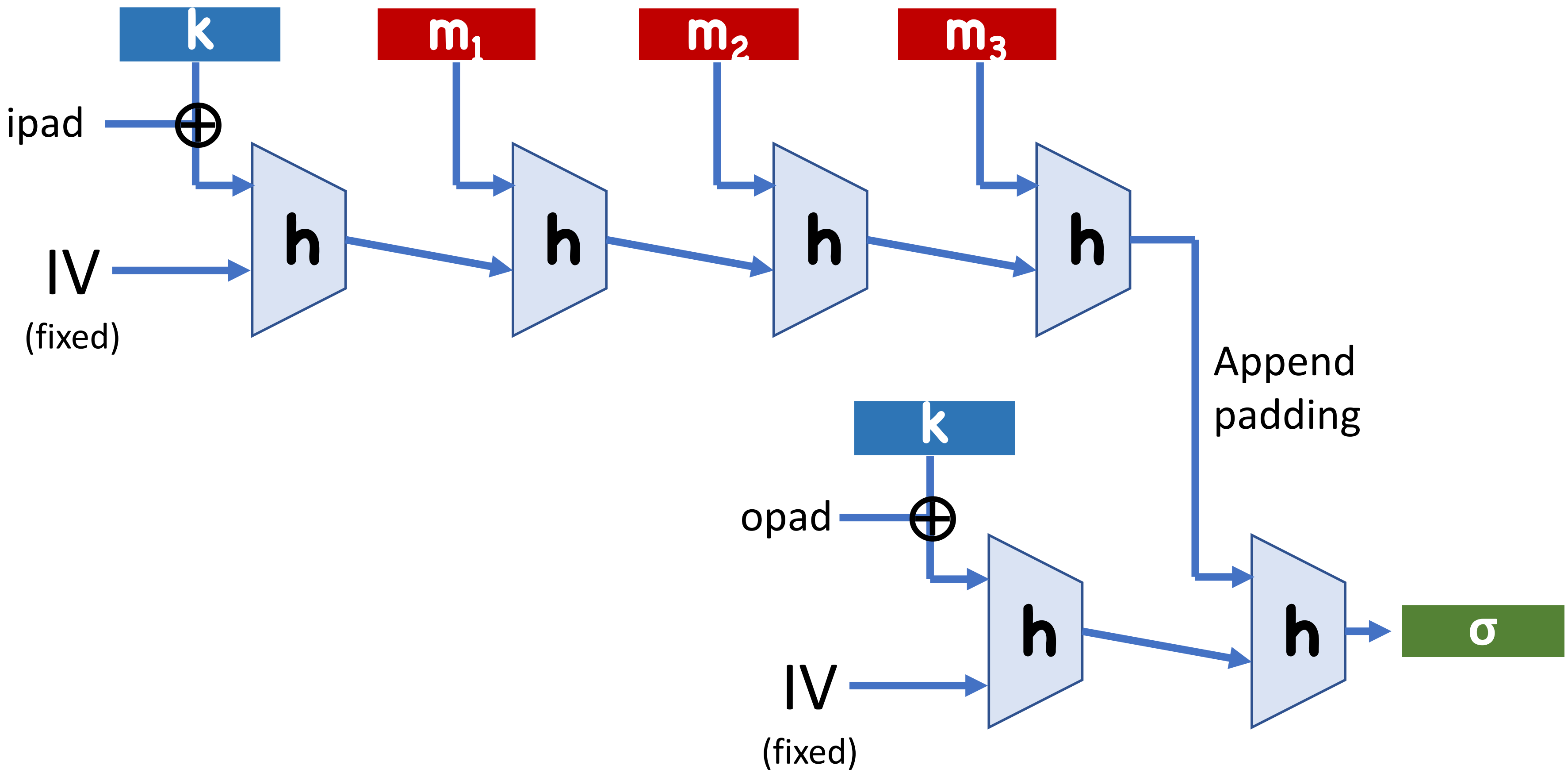
ROM does not apply to regular Merkle-Damgard

- Even if **h** is an ideal hash function

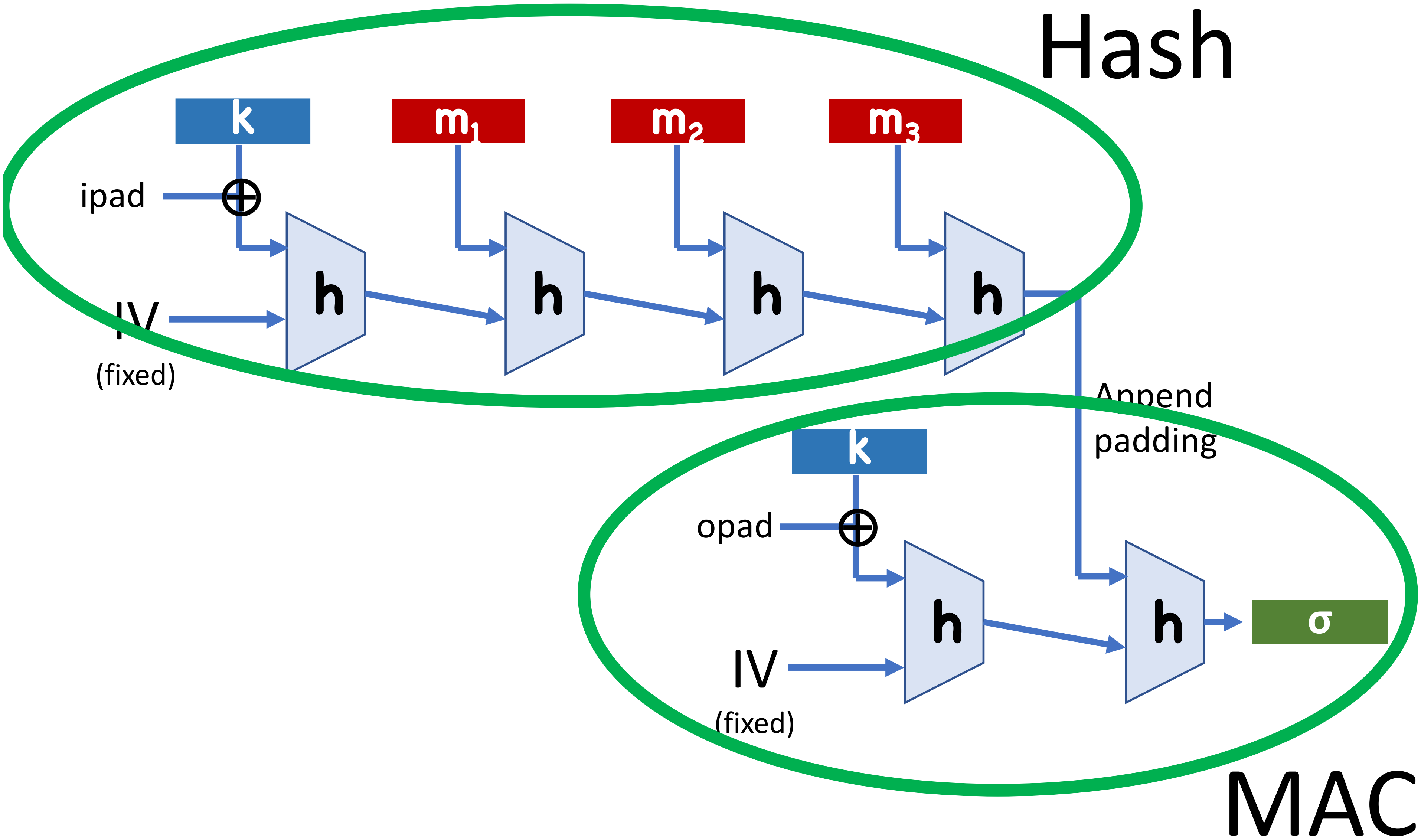
Takeaway: be careful about using ROM for non-"monolithic" hash functions

- Though still possible to pad MD in a way that makes it an ideal hash function if **h** is ideal

HMAC


k
m_1
m_2
m_3
ipad
h
h
h
h
IV
(fixed)
Append
padding
k
opad
h
h
σ
IV
(fixed)

HMAC



Hash
k
m1
m2
m3
ipad
h
h
h
h
IV
(fixed)
Append
padding
k
opad
h
h
σ
IV
(fixed)
MAC

HMAC

ipad,opad?

- Two different (but related) keys for hash and MAC
- ipad makes hash a “secret key” hash function
- Even if not collision resistant, maybe still impossible to find collisions when hash key is secret
- Turned out to be useful after collisions found in MD5

Commitment Schemes

Anagrams and Astronomy

<u>Galileo and the Rings of Saturn</u>

- Galileo observed the rings of Saturn, but mistook them for two moons

- Galileo wanted extra time for verification, but not to get scooped

- Circulates anagram
  SMAISMRMILMEPOETALEUMIBUNENUGTTAUIRAS

- When ready, tell everyone the solution:
  altissimum planetam tergeminum observavi
  ("I have observed the highest planet tri-form")

Anagrams and Astronomy

Enter Huygens

- Realizes Galileo actually saw rings
- Circulates

AAAAAAA CCCCC D EEEEE G H IIIIIII LLLL MM NNNNNNNNN OOOO PP Q RR S TTTTT UUUUU

- Solution:

annulo cingitur, tenui, plano, nusquam cohaerente, ad eclipticam inclinato

(“it is surrounded by a thin flat ring, nowhere touching, and inclined to the ecliptic”)

Commitment Scheme

Different than encryption

- No need for a decryption procedure
- No secret key
- But still need secrecy (“hiding”)
- Should only be one possible opening (“binding”)
- (Sometimes other properties needed as well)

Anagrams are Bad Commitments

If too short (e.g. one, two, three words), possible to reconstruct answer

If too long, multiple possible solutions

- Kepler tries to solve Galileo's anagram as

`salue umbistineum geminatum martia proles`

(hail, twin companionship, children of Mars)

(Non-interactive) Commitment Syntax

Message space **M**
Ciphertext Space **C**
(suppressing security parameter)

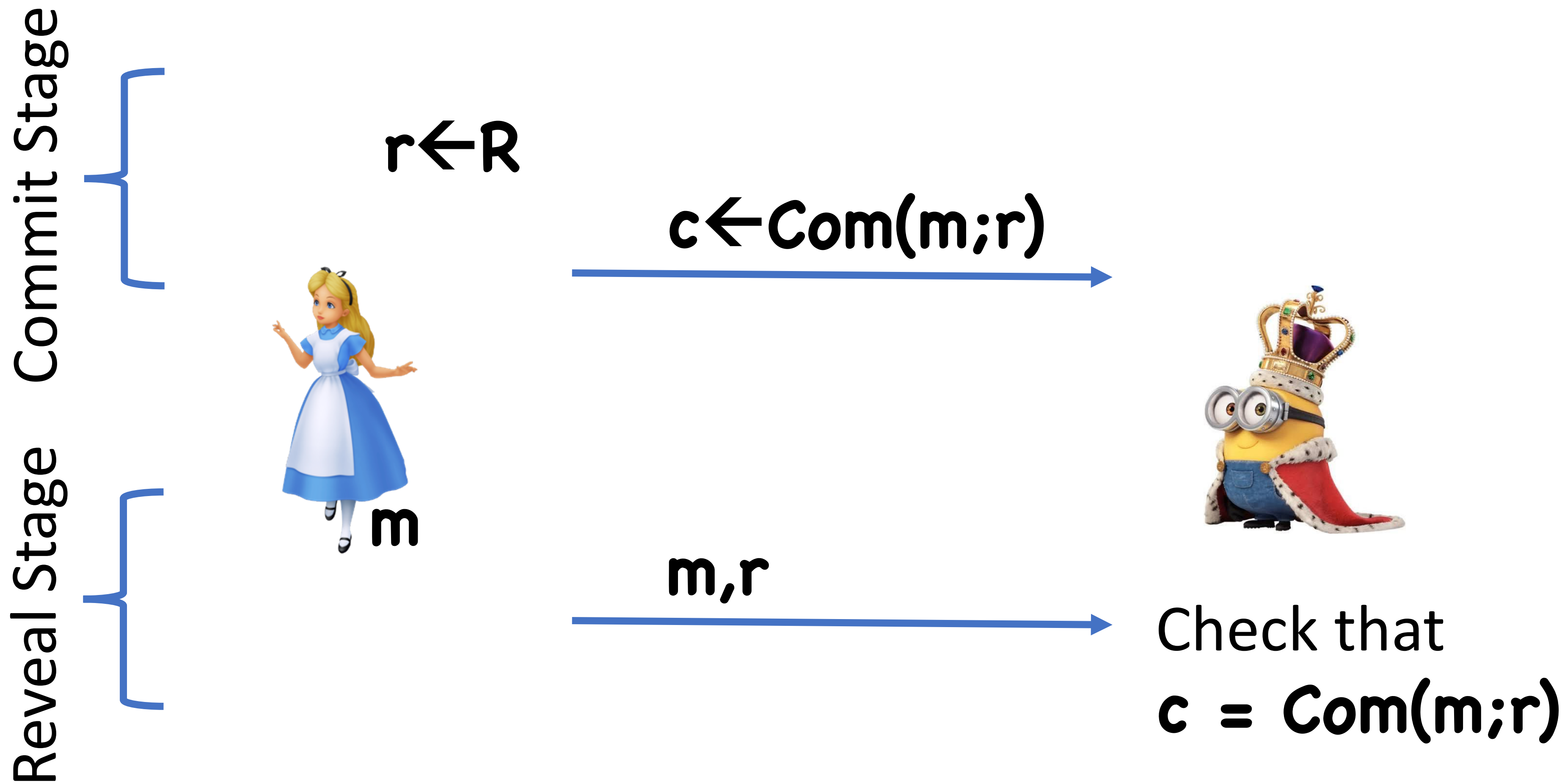
Com(m; r): outputs a commitment **c** to **m**

Commitments with Setup

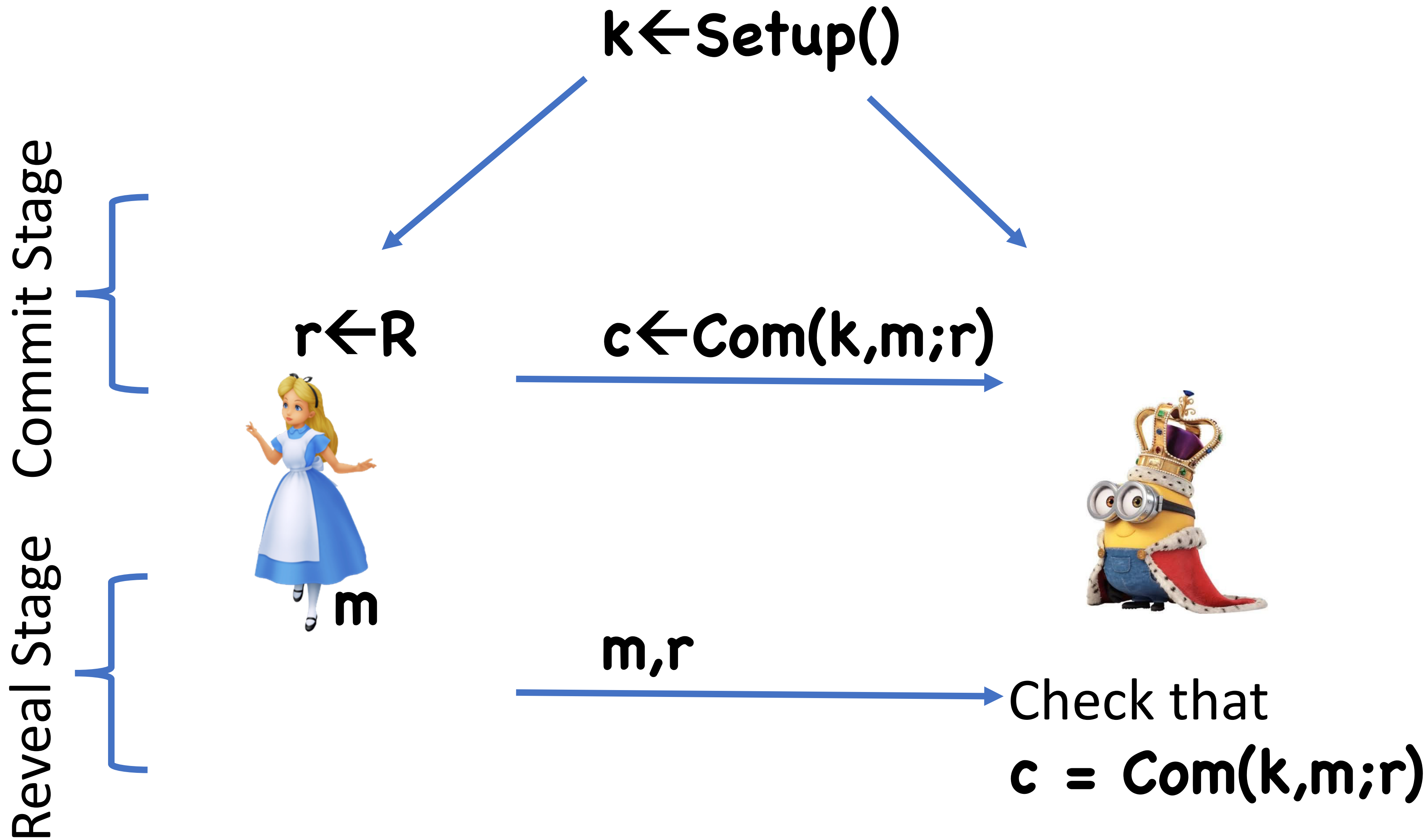
Message space **M**
Ciphertext Space **C**
(suppressing security parameter)

Setup(): Outputs a key **k**
Com(k, m; r): outputs a commitment **c** to **m**

Using Commitments



Using Commitments (with setup)



Security Properties

Hiding: **c** should hide **m**

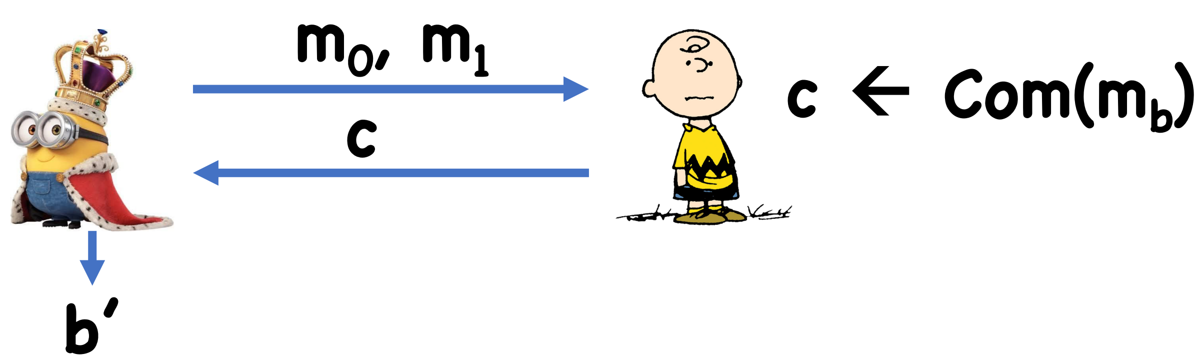
- Perfect hiding: for any $\mathbf{m_0}$, $\mathbf{m_1}$,

$$\mathrm{Com}(m_0) \stackrel{d}{=} \mathrm{Com}(m_1)$$

- Statistical hiding: for any $\mathbf{m_0}$, $\mathbf{m_1}$,

$$\Delta(\ \mathrm{Com}(m_0), \mathrm{Com}(m_1)\ ) < \mathrm{negl}$$

- Computational hiding:



Security Properties (with Setup)

Hiding: **c** should hide **m**

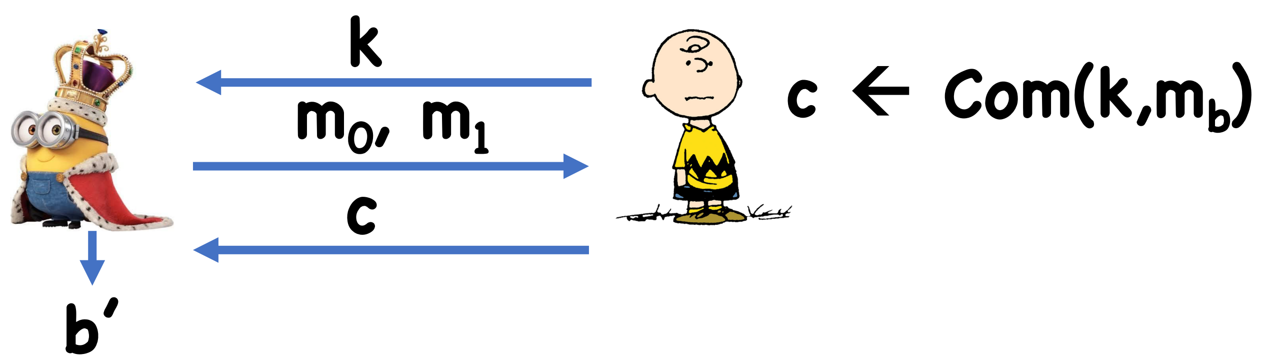
- Perfect hiding: for any m_0, m_1,

$$k,\mathrm{Com}(k,m_0) \stackrel{d}{=} k,\mathrm{Com}(k,m_1)$$

- Statistical hiding: for any m_0, m_1,

$$\Delta(\ [k,\mathrm{Com}(k,m_0)],\ [k,\mathrm{Com}(k,m_1)]\ ) < \mathrm{negl}$$

- Computational hiding:



Security Properties

Binding: Impossible to change committed value

- Perfect binding: For any **c**, ∃ at most a single **m** such that **c = Com(m;r)** for some **r**

- Computational binding: no PPT adversary can find $(m_0,r_0),(m_1,r_1)$ such that:

  $$Com(m_0;r_0)=Com(m_1;r_1)$$
  $$m_0 \neq m_1$$

Security Properties (with Setup)

Binding: Impossible to change committed value

- Perfect binding: For any **k,c**, $\exists$ at most a single **m** such that **c = Com(k,m;r)** for some **r**

- Statistical binding: except with negligible prob over **k**, for any **c**, $\exists$ at most a single **m** such that **c = Com(k,m;r)** for some **r**

- Computational binding: no PPT adversary, given **k←Setup()**, can find $(m_0,r_0),(m_1,r_1)$ such that

  $Com(k,m_0;r_0)=Com(k,m_1;r_1)$

  $m_0 \neq m_1$

Who Runs **Setup()**

Trusted third party (TTP)?

Alice?

- Must ensure that Alice cannot devise **k** for which she can break binding
- If binding holds, can actually devise scheme **Com'** without setup

Bob?

- Must ensure Bob cannot devise **k** for which he can break hiding

Anagrams as Commitment Schemes

Com(m) = sort characters of message

Problems?

- Not hiding: “Jupiter has four moons” vs “Jupiter has five moons”

- Not binding: Kepler decodes Galileo’s anagram to conclude Mars has two moons

Anagrams as Commitment Schemes

Com(m) = add random superfluous text, then sort characters of message

Might still not be hiding

- Need to guarantee, for example that expected number of each letter in output is independent of input string

Still not binding…

Other Bad Commitments

Com(m) = m

- Has binding, but no hiding

Com(m;r) = m⊕r

- Has hiding, but no binding

Can a commitment scheme be both statistically hiding and statistically binding?

A Simple Commitment Scheme

Let **H** be a hash function

Com(m;r) = H(m || r)

Binding?

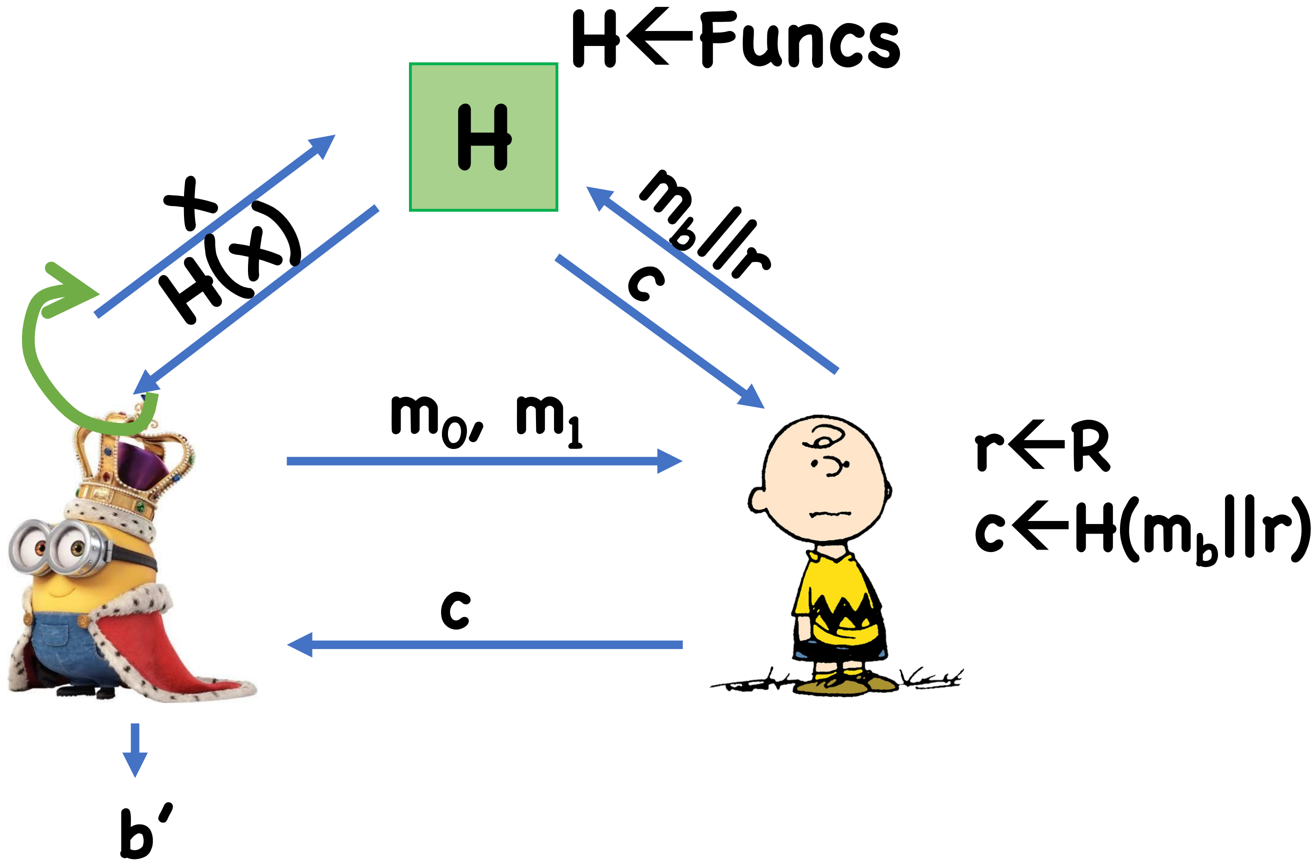
Hiding?

Theorem: $\mathbf{Com(m;r) = H(m||r)}$ has:

- Perfect binding assuming $\mathbf{H}$ is injective
- Computational binding assuming $\mathbf{H}$ is collision resistance (implied by RO)

- Computational hiding in the Random Oracle Model

Hiding



Proof of Hiding

Suppose  never queries **H** on $\mathbf{m_b||r}$

Then all query answers and commitment **c** seen by  are independent uniform strings

-  has no chance of determining **b**

Probability  queries on $\mathbf{m_b||r}$?

- At most $\mathbf{q/|R|}$ = negligible

“Standard Model” Commitments?

Random oracle model proof is heuristic argument for security

Can we prove it under assumptions such as collision resistance, etc?

Single Bit to Many Bit

Let **(Setup,Com)** be a commitment scheme for single bit messages

Let $\mathbf{Com'(k,m;\ r)=(Com(k,m_1;r_1),\dots,Com(k,m_t;r_t))}$

- $\mathbf{m = (m_1,\dots,m_t),\ m_i \in \{0,1\}}$
- $\mathbf{r = (r_1,\dots,r_t),\ r_i}$ are randomness for **Com**

Theorem: If **(Setup,Com)** is **(t,ε)-**binding, then **(Setup,Com')** is **(t-t',ε)-**binding

Theorem: If **(Setup,Com)** is **(t,ε)-**hiding, then **(Setup,Com')** is **(t,qε)-hding**

Binding

Suppose  breaks binding of **Com'**

Given **k**, produces $(m_1^0, r_1^0), \ldots, (m_t^0, r_t^0)$, $(m_1^1, r_1^1), \ldots, (m_t^1, r_t^1)$ such that

- $(m_1^0, \ldots, m_t^0) \neq (m_1^1, \ldots, m_t^1)$
- $\mathrm{Com}(k, m_i^0; r_i^0) = \mathrm{Com}(k, m_i^1; r_i^1)$ for all **i**

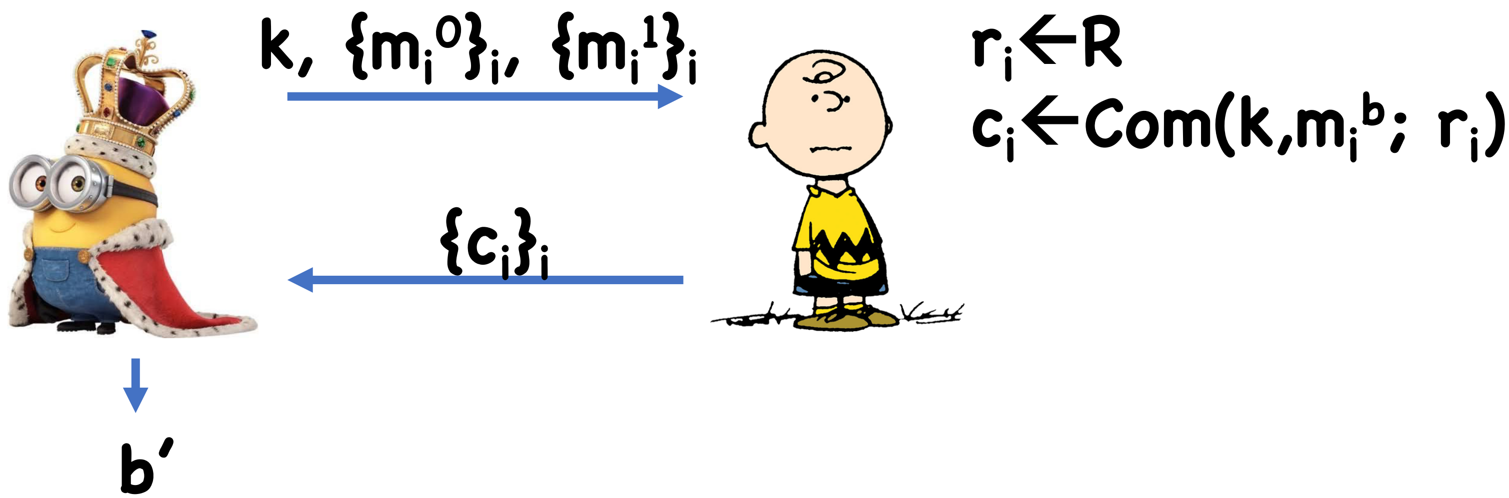
Therefore, $\exists$ **i** such that $m_i^0 \neq m_i^1$ but

$$\mathrm{Com}(k, m_i^0; r_i^0) = \mathrm{Com}(k, m_i^1; r_i^1)$$

$\Rightarrow$ Break binding of **Com**

Hiding

Suppose  breaks (say, computational malicious) hiding



Hiding

Proof by Hybrids

Hybrid **j**:
- For each **$i \leq j$**, **$c_i = \mathsf{Com}(k,m_i^1,r_i)$**
- For each **$i > j$**, **$c_i = \mathsf{Com}(k,m_i^0,r_i)$**

Hybrid **0**: commit to **$\{m_i^0\}_i$**
Hybrid **t**: commit to **$\{m_i^1\}_i$**

$\exists$ **j** such that  distinguishes Hyb **j-1** from Hyb **j**
$\Rightarrow$ break hiding of **Com**

Single Bit to Many Bit

Let **(Setup,Com)** be a commitment scheme for single bit messages

Let $\mathbf{Com'(k,m;\ r)=(Com(k,m_1;r_1),\ldots,Com(k,m_t;r_t))}$
- $\mathbf{m = (m_1,\ldots,m_t),\ m_i \in \{0,1\}}$
- $\mathbf{r = (r_1,\ldots,r_t),\ r_i}$ are randomness for **Com**

Therefore, suffices to focus on commitments for single bit messages

Statistically Hiding Commitments?

Let **H** be a collision resistant hash function with domain $\mathbf{X=\{0,1\}\times R}$ and range **Z**

Setup(): k←K, output **k**
Com(k, m; r) = H(k, (m,r))

Binding?

Hiding?

Statistically Hiding Commitments

Let **F** be a pairwise independent function family with domain **X={0,1}×R** and range **Y**

Let **H** be a collision resistant hash function with domain **Y** and range **Z**

Setup(): f←F, k←K, output **(f,k)**
Com((f,k), m; r) = H(k, f(m,r))

Theorem: If **|Y|/|X|** is "sufficiently large" and **H** is collision resistant, then **(Setup,Com)** has computational binding

Theorem: If **|X|** is "sufficiently large", then **(Setup,Com)** has statistical hiding

Theorem: If **H** is **(t,ε)**-collision resistant, then **(Setup,Com)** is $(t-t', \varepsilon+|Y|/|X|^2)$-computationally binding

Proof:

- Suppose $|Y| = |X|^2 \times \gamma$
- For any $x_0 \neq x_1$, $\Pr[f(x_0)=f(x_1)] < 1/(|X|^2 \times \gamma)$
- Union bound:

$$\Pr[\exists\ x_0 \neq x_1 \text{ s.t. } f(x_0)=f(x_1)] < \gamma$$

- Therefore, **f** is injective ⟹ any collision for Com must be a collision for **H**

Theorem: If **|X|** is "sufficiently large", then **(Setup,Com)** has statistical hiding

Goal: show **(f, k, H(k, f(0,r)))** is statistically close to **(f, k, H(k, f(1,r)))**

Min-entropy

Definition: Given a distribution **D** over a set **X**, the min-entropy of **D**, denoted $\mathbf{H_\infty(D)}$, is

$$- \min_x \log_2(\Pr[x \leftarrow D])$$

Examples:

- $H_\infty(\{0,1\}^n) = n$
- $H_\infty($ random n bit string with parity $0) = ?$
- $H_\infty($ random $i>0$ where $\Pr[i] = 2^{-i}$ $) = ?$

Leftover Hash Lemma

Lemma: Let **D** be a distribution on **X**, and **F** a family of pairwise independent functions from **X** to **Y**. Then

$$\Delta(\ (f,\ f(D))\ ,\ (f,\ R)\ ) \leq \varepsilon \text{ where}$$

- $f \leftarrow F$
- $R \leftarrow Y$
- $\log |Y| \leq H_\infty(D) + 2 \log \varepsilon$

"Crooked" Leftover Hash Lemma

Lemma: Let **D** be a distribution on **X**, and **F** a family of pairwise independent functions from **X** to **Y**, and **h** be any function from **Y** to **Z**. Then

$$\Delta(\ (f,\ h(f(D)))\ ,\ (f,\ h(R))\ )\ \leq\ \varepsilon \text{ where}$$

- $f \leftarrow F$
- $R \leftarrow Y$
- $\log |Z| \leq H_\infty(D) + 2 \log \varepsilon - 1$

Theorem: If we set $|R|=|Z|^3$, then **(Setup,Com)** is **(4/|Z|)-**statistically hiding

Goal: show $(f,\ k,\ H(k,\ f(0,r))\ )$ is statistically close to $(f,\ k,\ H(k,\ f(1,r))\ )$

Let $D=(0,r)$, min-entropy $\log\ |R|$

Set $R\ =|Z|^3$, $\varepsilon\ =\ 2/|Z|$

Then $\log\ |Z|\ \leq\ H_\infty(D)\ +\ 2\ \log\ \varepsilon\ -\ 1$

Theorem: If we set $|R|=|Z|^3$, then **(Setup,Com)** is **$(4/|Z|)$**-statistically hiding

For any **k**,

$$\Delta(\ (f,\ H(k,\ f(0,r)))\ ,\ (f,\ H(k,\ U))\ )\ \leq\ \varepsilon$$

Thus

$$\Delta(\ (f,\ H(k,\ f(0,r)))\ ,\ (f,\ H(k,\ f(1,r)))\ )\ \leq\ 2\varepsilon$$

Therefore

$$\Delta(\ (f,\ k,\ H(k,\ f(0,r))\ )\ ,\ (f,\ k,\ H(k,\ f(1,r))\ )\ \leq\ 2\varepsilon$$

Statistically Binding Commitments

Let **G** be a PRG with domain $\{0,1\}^\lambda$, range $\{0,1\}^{3\lambda}$

Setup(): choose and output a random **3λ**-bit string **k**

Com(b; r): If **b=0**, output **G(r)**, if **b=1**, output $G(r) \oplus k$

Theorem: (Setup,Com) is **($2^{-\lambda}$)-**statistically binding

Theorem: If **G** is a **(t,ε)-**secure PRG, then **(Setup,Com)** is **(t-t',2ε)-**computationally hiding

Theorem: If $\mathbf{G}$ is a $\mathbf{(t,\varepsilon)}$-secure PRG, then $\mathbf{(Setup,Com)}$ is $\mathbf{(t-t',2\varepsilon)}$-computationally hiding

Hybrids:

- Hyb 0: $\mathbf{c = Com(0;r) = G(r)}$ where $\mathbf{r \leftarrow \{0,1\}^{\lambda}}$
- Hyb 1: $\mathbf{c \leftarrow \{0,1\}^{3\lambda}}$
- Hyb 2: $\mathbf{c = S' \oplus k}$, where $\mathbf{S' \leftarrow \{0,1\}^{3\lambda}}$
- Hyb 3: $\mathbf{c = Com(1;r) = G(r) \oplus k}$ where $\mathbf{r \leftarrow \{0,1\}^{\lambda}}$

Theorem: (Setup,Com) is **$(2^{-\lambda})$**-statistically binding

Proof:

For any **r,r′**, $\Pr[G(r) = G(r')\oplus k] = 2^{-3\lambda}$

By union bound:

$\Pr[\exists\, r,r'$ such that $Com(k,0)=Com(k,1)]$

$= \Pr[\exists\, r,r'$ such that $G(r) = G(r')\oplus k] < 2^{-\lambda}$

More Problems with Anagrams

Huygens Discovers Saturn's moon Titan

- Sends the following to Wallis

ADMOVERE OCULIS DISTANTIA SIDERA NOSTRIS, UUUUUUUCCCRR-HNBQX

(First part meaning "to direct our eyes to distant stars")

Plaintext: **saturno luna sua circunducitur diebus sexdecim horis quatuor**

("Saturn's moon is led around it in sixteen days and four hours")

More Problems with Anagrams

<u>Huygens Discovers Saturn's moon Titan</u>

- Wallis replies with

AAAAAAAAA B CCCCC DDDD EEEEEEEEE F H
IIIIIIIIIIII LLL MMMMMM NNNNNN OOOOOOO PPPPP
Q RRRRRRRRRRR SSSSSSSSSSSSS TTTTTTTT
UUUUUUUUUUUUUUUUU X

(Contains all of the letters in Huygens' message, plus some)

More Problems with Anagrams

Huygens Discovers Saturn’s moon Titan

- When Huygens finally reveals his discovery, Wallis responds by giving solution to his anagram:

saturni comes quasi lunando vehitur. diebus sexdecim circuitu rotatur. novas nuper saturni formas telescopo vidimus primitus. plura speramus

(“A companion of Saturn is carried in a curve. It is turned by a revolution in sixteen days. We have recently observed new shapes of Saturn with a telescope. We expect more.”)

- Tricked Huygens into thinking British astronomers had already discovered Titan

More Problems with Anagrams

Sometimes, hiding and binding are not enough

For some situations (e.g. claiming priority on discoveries) also want commitments to be "non-malleable"

- Shouldn't be able to cause predictable changes to committed value

Beyond scope of this course

Next Time

Basing crypto on number-theoretic assumptions

- Factoring
- Discrete Log

Project 1 Out

I have given you a hash function BAH
(Bad Algorithm for Hashing)

Your job:

- Determine what kind of hash function it is
- Break it using differential cryptanalysis
- Propose a fix that the teaching staff will try to break

Reminders

Homework 4 Due April 3

Project 2 Due April 17