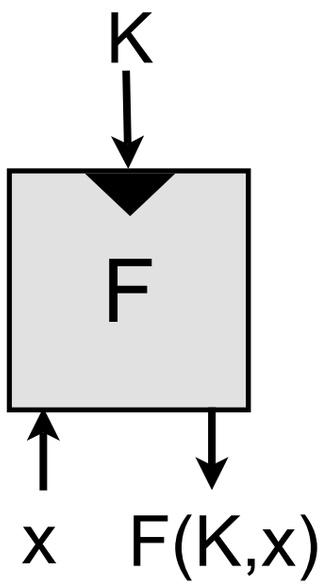
Security Against Related-Key Attacks: Constructions & Applications

David Cash

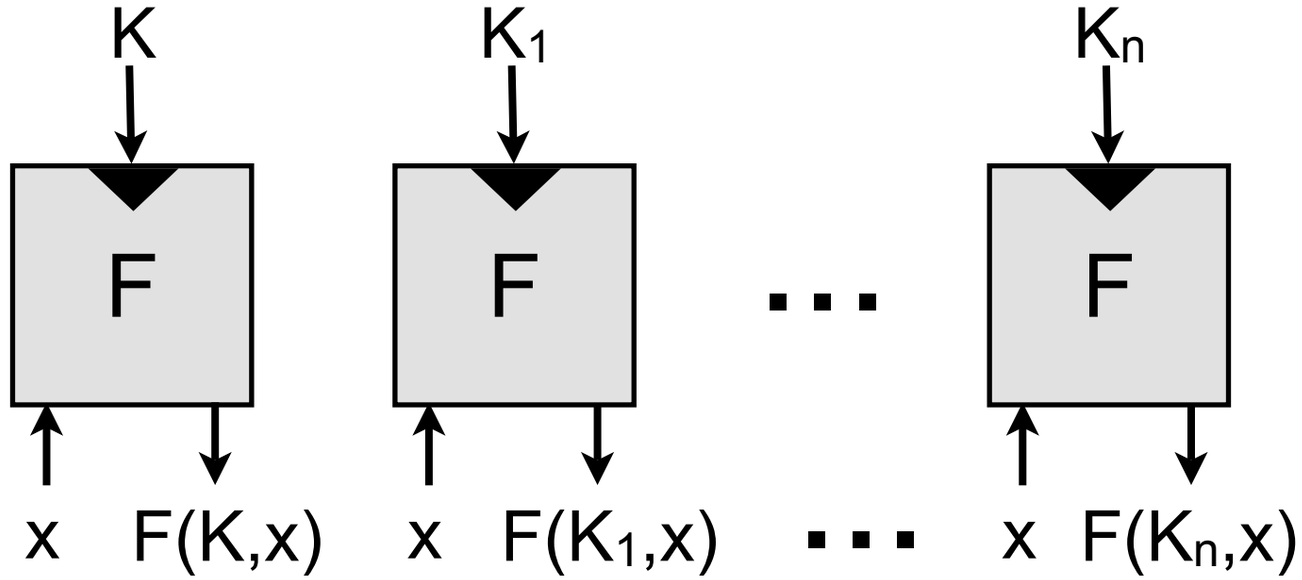
Ruhr-University Bochum

Joint work with Mihir Bellare (UCSD) and Rachel Miller (MIT)

Single-key attack on blockcipher F:



Related-key attack on blockcipher F:



K_1, ..., K_n derived from K in **adversary-specified** way.

Motivations for RKA Security

RKAs model **physical tampering**

- Tampering with a device can flip key bits

RKAs can model attacks enabled by **insecure legacy key exchange protocols**

- Adversary may interfere with sessions to force agreed keys to satisfy known relationship. Possible against 2PKDP and WEP.

Blockcipher designers focus on them, so there is a need for a theoretical foundation.

In Practice

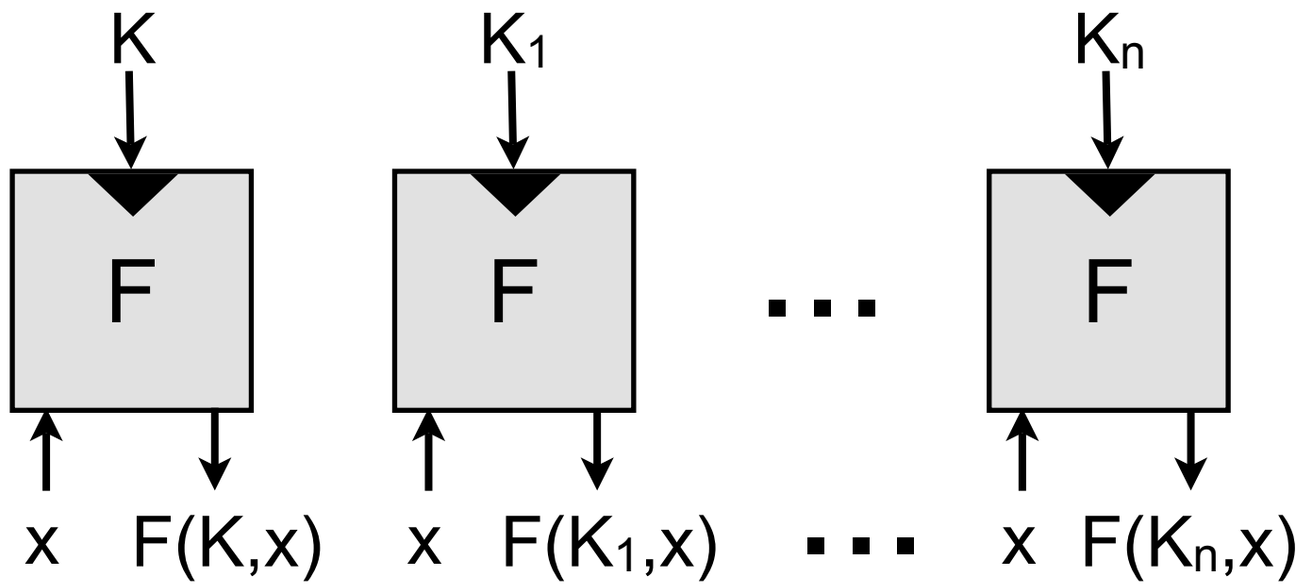
- RKAs introduced by Biham and Knudsen in early 1990s
- Since then, hundreds of papers mounting RKAs on various blockciphers
- Security goal for Rijndael/AES and all/other modern blockciphers.
- Recent unsettling RKAs on AES-192, AES-256

[Biryukov, Khovratovich'09]

[Biryukov, Dunkelman, Keller, Khovratovich,Shamir'10]

In Theory

Model of Bellare and Kohno (2003) defining what it means for F to be a Φ-RKA-PRF, where Φ is a set of related-key deriving functions



$K_i = \varphi(K)$ where $\varphi \in \Phi$

Today's Talk

Part 1: Constructions of Φ-RKA-PRFs

- Security proofs under standard assumptions (DDH)
- In standard model
- For "interesting", "non-trivial" Φ (group induced)
- Via new technical approach (key-malleability, key-fingerprints)

Warning: Not practical (proof of concept)!

Today's Talk

Part 1: Constructions of Φ-RKA-PRFs

Part 2: RKA security of other primitives

- RKA definitions for: wPRFs, Sigs, CCA-PKE, IBE, ...
- Practical methods for transferring RKA security between primitives
- Theoretical relations to explain strange behavior of which Φ are achievable

Part 1: Outline

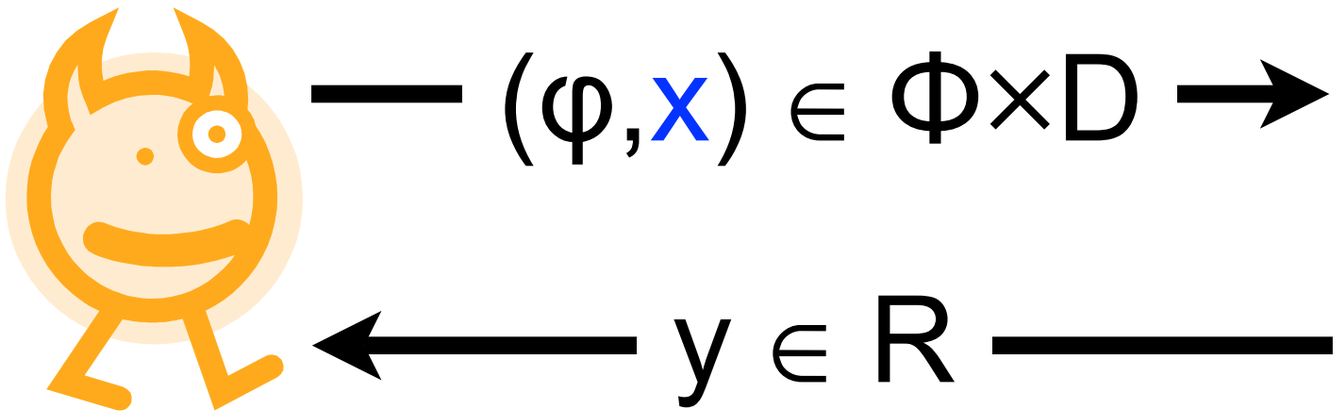
Part 1: Constructions of Φ-RKA-PRFs

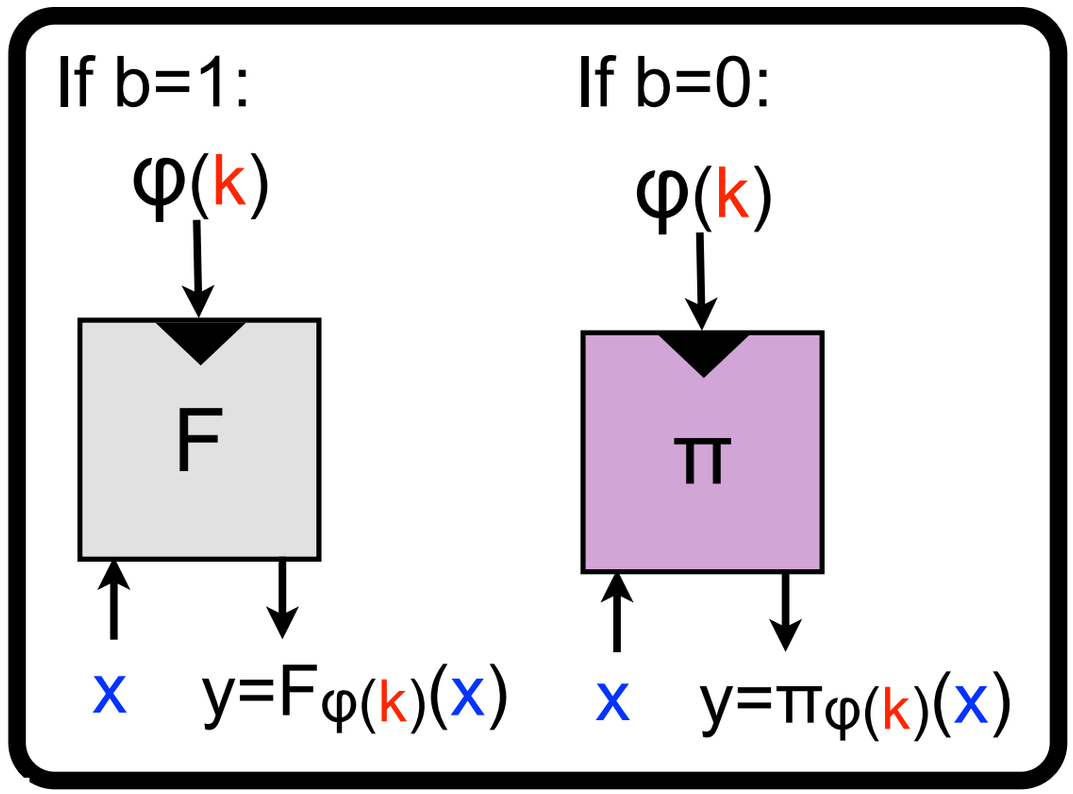
- Φ-RKA-PRF model and basic prior results
- Our construction and security theorem
- An intuitive but flawed approach: Key Malleability
- How to fix the approach: Key Fingerprints
- Extensions

RKA-PRF Model and Security Definition [BK'03]

Let F: K×D → R be a blockcipher and Φ be a set of "allowed functions," each mapping K → K.

Setup: Pick random k∈K, b∈{0,1}, random function π: K×D → R





Repeat until adversary outputs b′.

F is a **Φ-RKA-PRF** if Pr[b′ = b] -1/2 is small for all efficient adversaries.

Prior RKA-PRFs

Constructions of provable Φ-RKA-PRFs:

- For "uninteresting" Φ [Bellare,Kohno]
- In random oracle, ideal-cipher model [Bellare,Kohno], [Lucks]
- Under non-standard, interactive assumptions [Lucks], [Goldenberg,Liskov]

No proof under standard assumptions that RKA security is achievable for **ANY** interesting set Φ!

Problem evades naive solutions.

The “allowed functions” set Φ **determines the power of the RKA**.

Examples of possible Φ:

Φ-RKA-PRF is equivalent to PRF

- Φ = {id}, where id: K → K is the identity function on K

The “allowed functions” set Φ **determines the power of the RKA**.

Examples of possible Φ:

[Lucks'04]

Group-induced classes of functions:

Assume K is a group with operation $*$.

Define $\Phi^* = \{ \varphi^*_\Delta \mid \Delta \in K \}$ where $\varphi^*_\Delta(k) = k * \Delta$

Let $K = \{0,1\}^n$. We can define:

- $\Phi^{\oplus} = \{ \varphi^{\oplus}_\Delta \mid \Delta \in \{0,1\}^n \}$ where $\varphi^{\oplus}_\Delta(k) = k \oplus \Delta$
- $\Phi^{+} = \{ \varphi^{+}_\Delta \mid \Delta \in \{0,1\}^n \}$ where $\varphi^{+}_\Delta(k) = k+\Delta \bmod 2^n$

We will consider other groups for our constructions.

Group-induced classes are **technically interesting**.

- They allow for **arbitrary key modification**: for every pair of keys k, k′, some function in Φ^* that maps k to k′.
- Understanding RKAs with group-induced classes is a step in understanding the achievability of RKA security in general.

Prior Group-Induced RKA-PRFs

Ideal cipher resists group-induced RKAs. [Bellare,Kohno], [Lucks]

Standard model constructions under novel interactive assumptions. [Lucks], [Goldenberg,Liskov]

Example:

Definition 3. *Let P, P_4, g, g_2, and g_3 be defined as above. Let r be a random value in $\mathbb{Z}^*_{P_4}$. Define*

$$f(x) = g^{\left(g_2^{\left(g_3^{x+r}\right)}\right)} \bmod P.$$

Define $R = \{z \in \mathbb{Z}_P \mid \exists k \in \mathbb{Z}_{P_4} : z = F'_{\mathrm{DH}}(k)\}$.

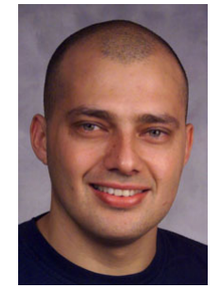
Diffie-Hellman Random Function Assumption (DHRFA): *It is infeasible, to distinguish f from a random function $\mathbb{Z}_{P_4} \to R$.*

Our results: New **standard-model constructions** of Φ^*-RKA-PRFs and Φ^*-RKA-PRPs under **standard assumptions** - DDH and DLIN

We will use the classic Naor-Reingold PRF, denoted NR:

$$\text{NR}: (\mathbf{Z}_p^*)^{n+1} \times \{0,1\}^n \rightarrow \mathbf{G} \ , \qquad \text{NR}_k(x) = g^{k_0 \cdot \prod_{i:\, x_i=1} k_i}$$

p is prime and **G** is a group of order p with generator g.



We will use the group-induced class

$$\Phi^* = \{ \varphi_d^* \mid d \in (\mathbf{Z}_p^*)^{n+1} \}, \quad \varphi_d^*(k) = k * d$$

where $*$ component-wise multiplication, i.e.

$$k * d = (k_0 \cdot d_0, \ldots , k_n \cdot d_n)$$

Our Construction

Let H be a hash function mapping into $\{0,1\}^n$.

Theorem. The function F defined by

$$F: (\mathbf{Z}_p^*)^{n+1} \times \{0,1\}^n \rightarrow \mathbf{G}\ ,\ F_k(x) = NR_k(H(x, g, g^{k_0}, g^{k_0 k_1}, \ldots, g^{k_0 k_n}))$$

is a Φ^*-RKA-PRF, assuming DDH holds in **G** and H is collision-resistant.

Φ^*-RKA-PRF security means adversary derives related keys by multiplying with chosen constants.

Our Construction

Let H be a hash function mapping into $\{0,1\}^n$.

Theorem. The function F defined by

$$F: (\mathbf{Z}_p^*)^{n+1} \times \{0,1\}^n \rightarrow \mathbf{G} \ , \ F_k(x) = NR_k(H(x, g, g^{k_0}, g^{k_0 k_1}, \ldots, g^{k_0 k_n}))$$

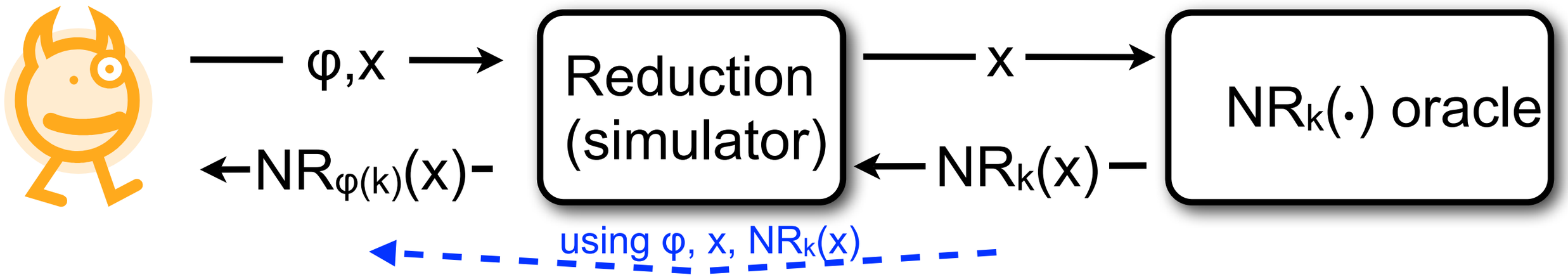
is a Φ^*-RKA-PRF, assuming DDH holds in **G** and H is collision-resistant.

Why is this theorem true?

Let's start with a thought experiment, and try to prove that the original NR function is a Φ^*-RKA-PRF.

Let's try to reduce Φ^*-RKA-PRF security to PRF security.

This means we need to simulate queries in the RKA game:



Possible proof strategy: Using φ, x, and $NR_k(x)$, just compute $NR_{\varphi(k)}(x)$ ourselves.

If we can do this, then we can simulate RKA queries, and the proof should go through.

It turns out that we **can** do this!

Claim. Given **d**, x, and $NR_{\mathbf{k}}(x)$, we can efficiently compute $NR_{\mathbf{k}*\mathbf{d}}(x)$.

Proof: Output

$$NR_k(x)^{d_0 \cdot \prod_{i:\, x_i=1} d_i} = g^{(k_0 \cdot \prod_{i:\, x_i=1} k_i)\,(d_0 \cdot \prod_{i:\, x_i=1} d_i)}$$

$$= g^{k_0\, d_0 \cdot \prod_{i:\, x_i=1} k_i\, d_i}$$

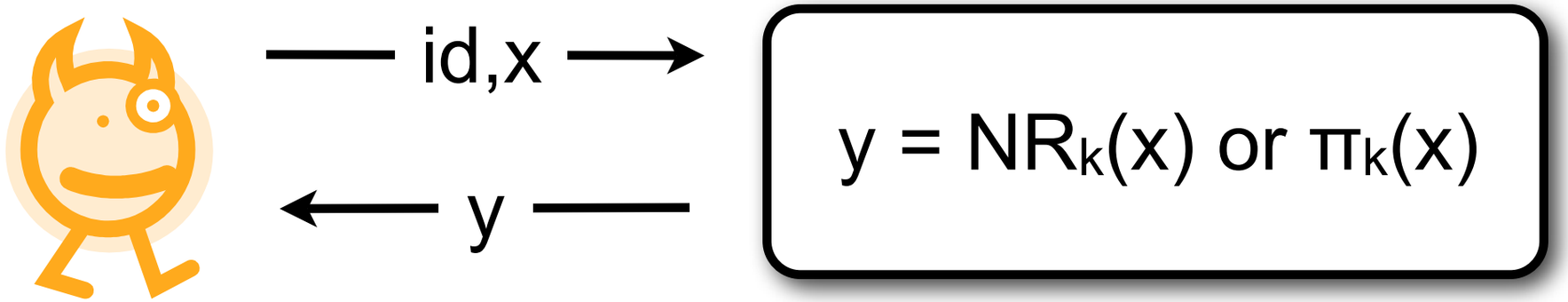
$$= NR_{k*d}(x)$$ ❑

This is all we needed for the simulation!

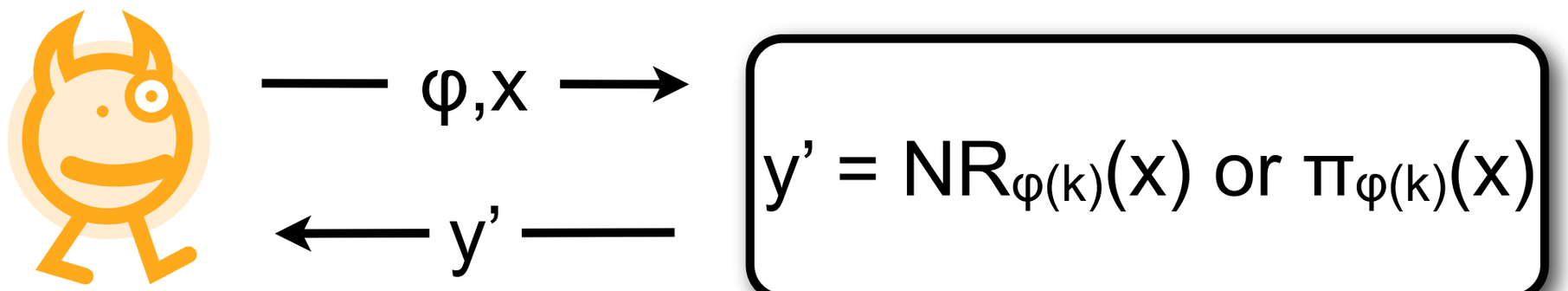
Something must be wrong... this means we can attack the Φ^*-RKA-PRF security of NR!

Two query attack on Φ^*-RKA-PRF-security of NR:

Pick some arbitrary x and query for $NR_k(x)$. (set φ = id)

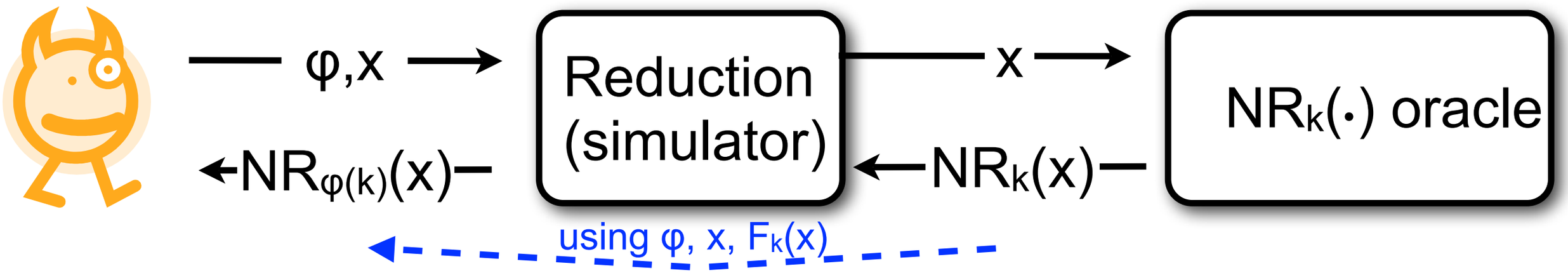


Now pick some φ and use φ,x,y to "predict" $NR_{\varphi(k)}(x)$. Then query (φ,x).

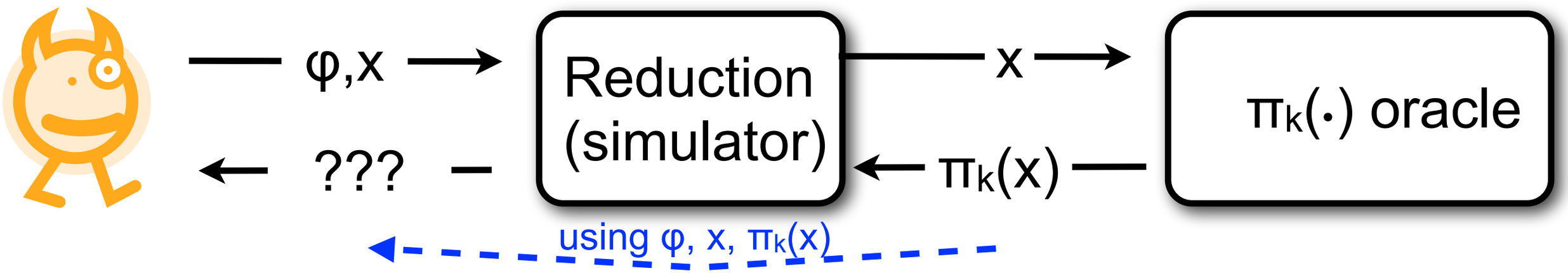


If y' matches predicted value, then guess real, otherwise rand.

What went wrong in our proof attempt?

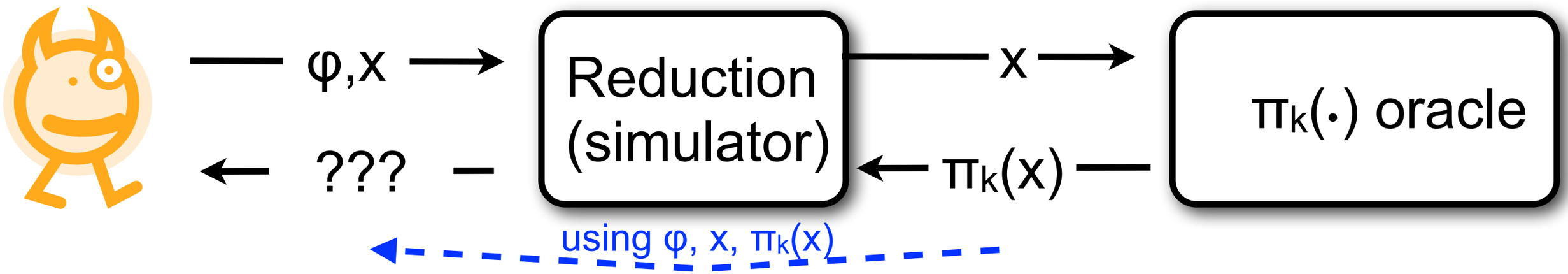


The problem is that the random case is not simulated properly:



Key Observation:

This works as long as the adversary never repeats an x.



Reduction will compute

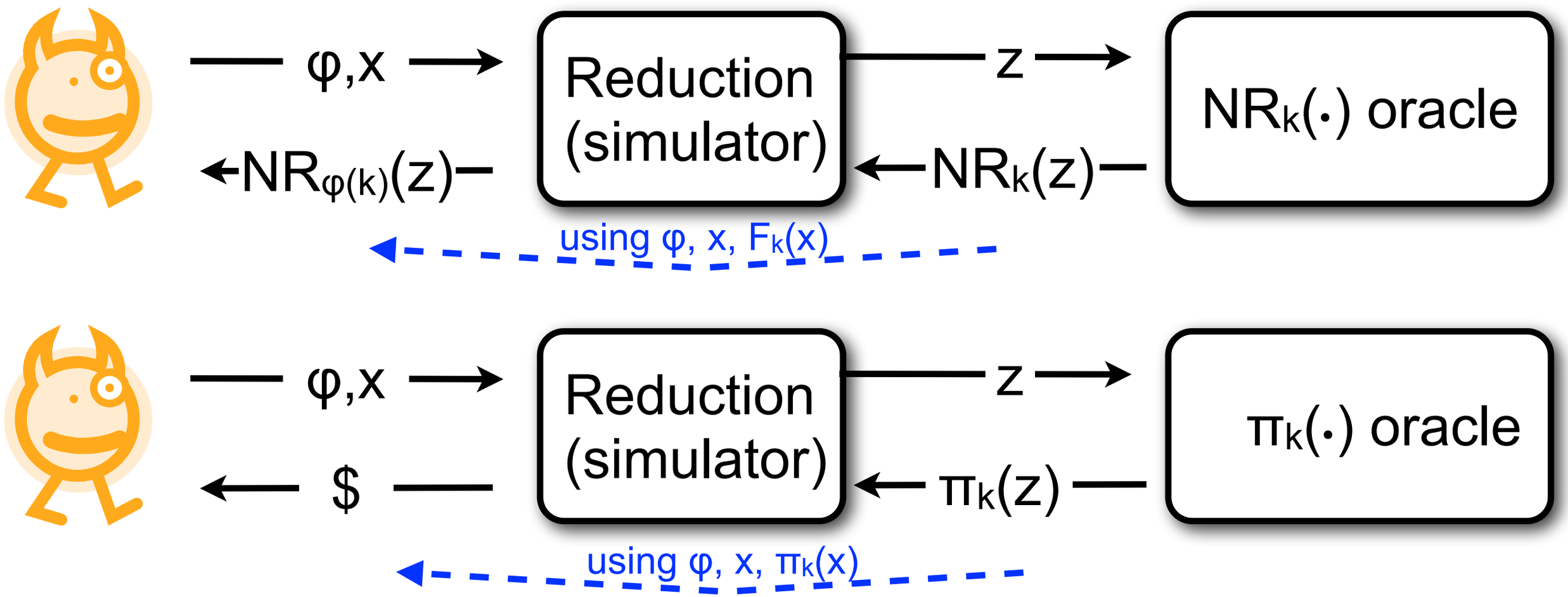
$$\pi_k(x) \quad d_0 \cdot \prod_{i:\, x_i=1} d_i$$

If each query is on a fresh x, then each value from the simulator will be uniform and independent - i.e. the simulation will be good.

Now let's modify NR to exploit the key observation.

Define F by $F_k(x) = NR_k(z)$, where $z = f(k,x)$ and f is injective.

The idea is that each z queried by reduction will be "fresh" - if φ or x change, then so will z.



But how to define f(k,x)? The reduction needs to compute z.

But how to define an injective f(k,x)?

k || x (concatenation) and H(k,x) are both “injective” ...

.... but the reduction won’t be able to compute them.

Instead, we introduce a new tool: **key fingerprints**

Definition. A **key-fingerprint for E** is a tuple of inputs $(w_1, \ldots, w_m)$ such that for every pair of distinct keys k, k’,

$$E_k(w_i) \neq E_{k'}(w_i) \text{ for some } i$$

How to construct a key-fingerprint for NR:

Use $(w_0, ..., w_n)$, where

$w_0 = 0000...0$
$w_1 = 1000...0$
$w_2 = 0100...0$
$w_3 = 0010...0$

...

$w_n = 0000...1$

Claim: This is a key fingerprint for NR.

Proof: Fix any $k \neq k'$. Need to find i s.t. $NR_k(w_i) \neq NR_{k'}(w_i)$.

If $k_0 \neq k'_0$, then done. $NR_k(w_0) = g^{k_0} \neq NR_{k'}(w_0) = g^{k'_0}$

Otherwise, $k_0 = k'_0$, but
$k_i \neq k'_i$ for some $i > 0$, so: $NR_k(w_i) = g^{k_0 k_i} \neq NR_{k'}(w_i) = g^{k'_0 k'_i}$

❑

How to Use Key Fingerprints

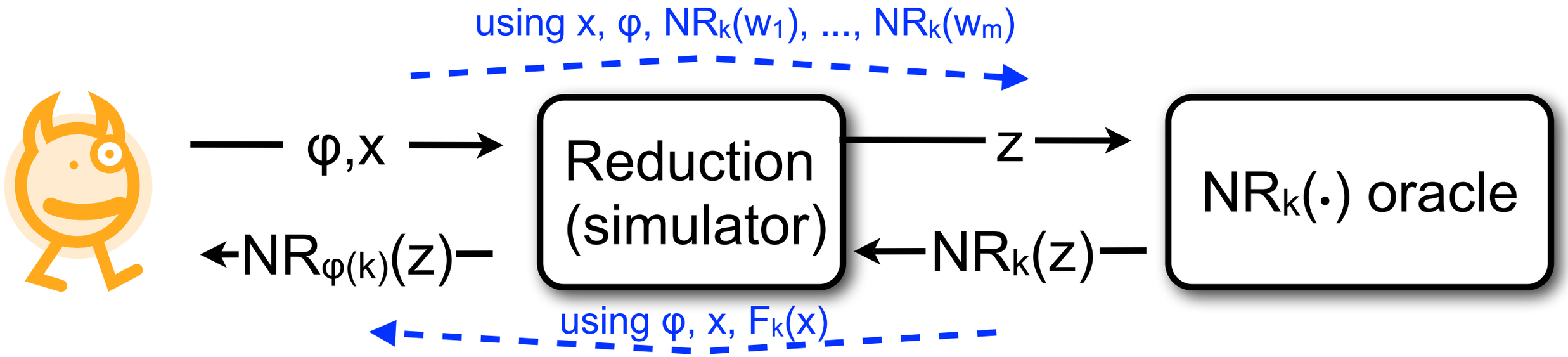
Define F by $F_k(x) = NR_k(z)$, where $z = f(k,x)$ and f is injective.

Set

$$z = f(k,x) = H(x, NR_k(w_1), \ldots NR_k(w_m))$$

to get a (computationally) unique z.

In proof:



Putting everything together:

Theorem. The function F by

$$F: (\mathbf{Z}_p^*)^{n+1} \times \{0,1\}^n \rightarrow \mathbf{G}\ ,\ F_k(x) = NR_k(H(x, g, g^{k_0}, g^{k_0 k_1}, \ldots, g^{k_0 k_n}))$$

is a Φ^*-RKA-PRF, assuming DDH holds in **G** and H is collision-resistant.

Same as $F_k(x) = NR_k(H(x, NR_k(w_0), \ldots, NR_k(w_n)))$.

Proof reduces Φ^*-RKA-PRF security to PRF security of NR.

(Naor and Reingold proved that NR is a PRF under DDH.)

Many proof details omitted here. **Full version on eprint.**

See paper for **generalization**: we can use any "key-malleable PRF" that admits a key fingerprint.

Extension #1:

We build a Φ^+-RKA-PRF, where Φ^+ allows component-wise addition.

Requires exponential (in n) hardness of DDH.

Extension #2:

We build a Φ-RKA-PRF under the DLIN assumption, which is **"weaker" than DDH**.

This works by adapting our approach to use the DLIN-based PRF of Lewko and Waters.


[Lewko, Waters'09]

Constructing RKA-PRPs

So far we've just done RKA-PRFs, not RKA-PRPs.

Given a RKA-PRF and regular PRP, we can construct an RKA-PRP.

Theorem. Let F be a Φ-RKA-PRF and E be a PRP. Let

$$G_k(x) = E_{k'}(x), \text{ where } k' = F_k(0^n).$$

Then G is a Φ-RKA-PRP.

Proof omitted. See full version on eprint.

Part 2: Outline

Part 2: RKA security of other primitives

- The Fragility of Φ
- RKA Security for wPRFs
- RKA Security for Digital Signatures, IBE, CCA-PKE, ...

A Step Back: Which Φ?

We just saw that Φ^* is "achievable" under DDH.

Φ-RKA-PRF security is impo

- Φ = All functions mappi
- Φ = All permutations ma
- Φ contains a constant function

φ is a **constant function** if there is a c s.t. φ(x) = c for all x.

Theorem: No F is Φ-RKA-PRF secure if Φ contains a constant function.

[Bellare,Kohno]

Proof idea: We give an attack. Query (φ,x) where φ is a constant function to get output y.

y will equal $F_{\varphi(K)}(x)$ or random... but we can compute $F_{\varphi(K)}(x)$ ourselves and check.

A Step Back: Which Φ?

Let keyspace be $\{0,1\}^n$.

Let $\Phi^{\oplus}$ and Φ^{+} be group-induced classes for bit-wise XOR and addition mod 2^n.

Theorem: No F is $\Phi^{\oplus} \cup \Phi^{+}$-RKA-PRF secure. [Bellare,Kohno]

Proof idea: We give an attack. Query same input x under keys

$$K \oplus (0^{n-i}\, 1\, 0^{i-1}) \quad \text{and} \quad K + 2^{i-1}$$

These keys are equal iff K[i] = 1, so responses match if K[i]=1.

Repeat to learn K bit-by-bit. ❑

A Step Back: Which Φ?

The set of possible Φ represent the “inherent power” of RKAs.

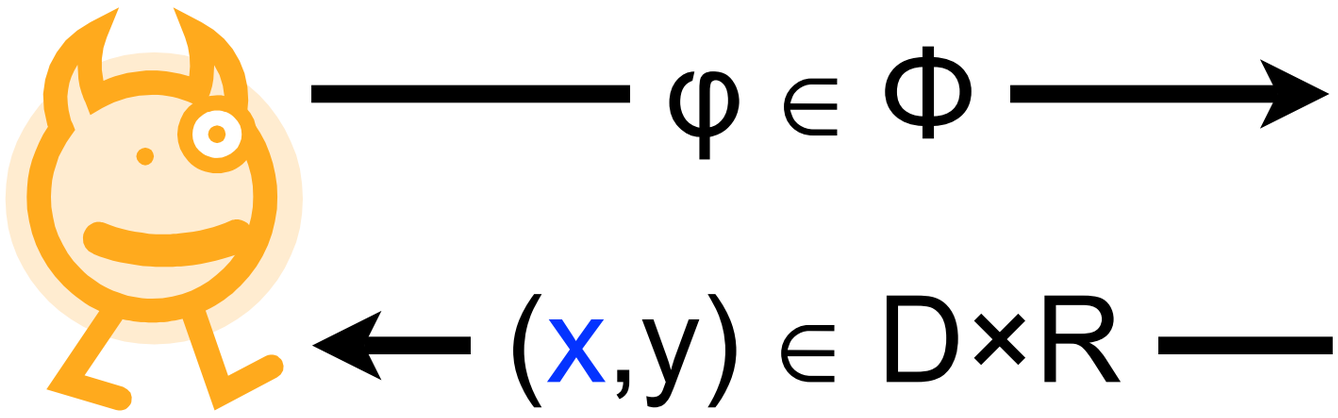
But which Φ are possible depends on what you’re attacking.

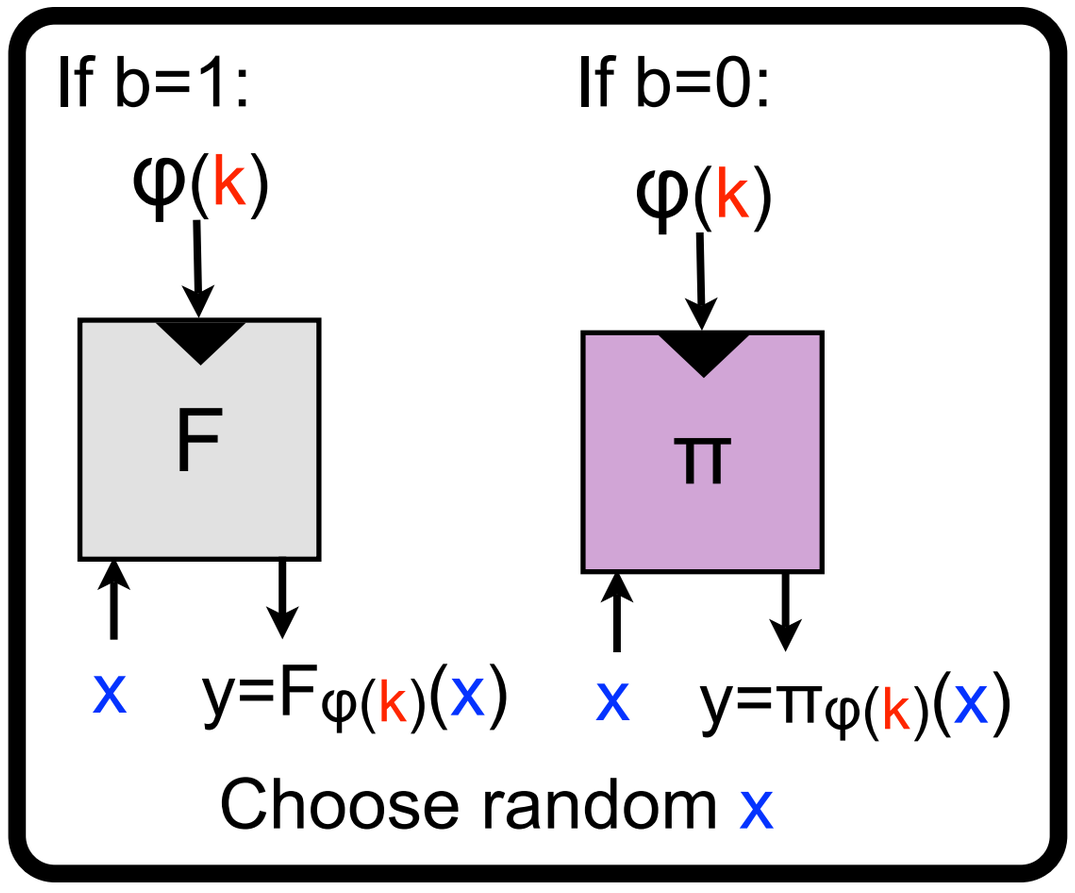
A good example to look at: weak PRFs (wPRFs).

Weak PRFs: RKA Security Definition

Let F: K×D → R be a blockcipher and Φ be a set of "allowed functions," each mapping K → K.

Setup: Pick random k∈K, b∈{0,1}, random function π: K×D → R





Repeat until adversary outputs b′.

F is a **Φ-RKA-wPRF** if Pr[b′ = b] -1/2 is small for all efficient adversaries.

RKA-wPRFs: Which Φ?

Φ-RKA-PRFs are always Φ-RKA-**w**PRFs.

Let: $\mathbf{RKA}$[PRF] = { Φ : ∃ F that is Φ-RKA-PRF secure}

$\mathbf{RKA}$[wPRF] = { Φ : ∃ F that is Φ-RKA-wPRF secure}

Claim: $\mathbf{RKA}$[PRF] ⊂ $\mathbf{RKA}$[wPRF].

(Proof is easy.)

Theorem: $\mathbf{RKA}$[PRF] ⊅ $\mathbf{RKA}$[wPRF].

i.e. wPRFs **inherently** resist more RKAs than PRFs.

Theorem: **RKA**[PRF] $\not\supset$ **RKA**[wPRF].

Proof: We need to find Φ s.t.

- ∃ F : F is Φ-RKA-wPRF
- ∄ F : F is Φ-RKA-PRF

Take $\Phi = \{id, \varphi_1, \ldots, \varphi_n, \varphi_{flip}\}$, where

- $\varphi_i(K)$ flips the first bit of K iff K[i] = 1
- $\varphi_{flip}(K)$ always flips the first bit of K

Claim: No F is Φ-RKA-PRF secure.

Learn K[i] by querying same x under $\varphi_i(K)$ and id.

- If K[i] = 0, responses match.
- If K[i] = 1, responses differ w.h.p. (because $\varphi_{flip} \in \Phi$)

Theorem: **RKA**[PRF] $\not\supset$ **RKA**[wPRF].

Proof: We need to find Φ s.t.

- ∃ F : F is Φ-RKA-wPRF
- ∄ F : F is Φ-RKA-PRF

Take Φ = {id, φ_1, …, φ_n, φ_{flip}}, where

- φ_i(K) flips the first bit of K iff K[i] = 1
- φ_{flip}(K) always flips the first bit of K

Claim: Any wPRF that ignores its first key bit is Φ-RKA-wPRF secure.

Queries in Φ-RKA-wPRF game always answered with same key - if no x repeats, there's no way to tell this from a random function.

RKAs for Other Primitives

We define Φ-RKA security for:

IBE-CPA: Key extraction queries under φ(msk).

PKE-CCA: Decryption queries under φ(sk).

EUF-CMA: Signing queries under φ(sk).

SE-CPA: Encryption queries under φ(K).

SE-CCA: Encryption/decryption queries under φ(K).

Relations Between **RKA**[Prim]

	PRF	wPRF	IBE	Sig	SE-CCA	SE-CPA	PKE-CCA
PRF	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$
wPRF	$\not\subseteq$	$\subseteq$	$\not\subseteq$			$\subseteq$	$\not\subseteq$
IBE	$\not\subseteq$	$\not\subseteq$	$\subseteq$	$\subseteq$	$\not\subseteq$	$\not\subseteq$	$\subseteq$
Sig	$\not\subseteq$	$\not\subseteq$		$\subseteq$	$\not\subseteq$	$\not\subseteq$	$\not\subseteq$
SE-CCA	$\not\subseteq$				$\subseteq$	$\subseteq$	
SE-CPA	$\not\subseteq$		$\not\subseteq$		$\not\subseteq$	$\subseteq$	$\not\subseteq$
PKE-CCA	$\not\subseteq$	$\not\subseteq$			$\not\subseteq$	$\not\subseteq$	$\subseteq$

- $\not\subseteq$ statements proven via counterexamples like before.
- $\subseteq$ statements proven by “RKA-security preserving” transforms.
- Transforms from PRFs are efficient and practical.

Relations Between **RKA**[Prim]

	PRF	wPRF	IBE	Sig	SE-CCA	SE-CPA	PKE-CCA
PRF	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$
wPRF	$\not\subseteq$	$\subseteq$	$\not\subseteq$	?	?	$\subseteq$	$\not\subseteq$
IBE	$\not\subseteq$	$\not\subseteq$	$\subseteq$	$\subseteq$	$\not\subseteq$	$\not\subseteq$	$\subseteq$
Sig	$\not\subseteq$	$\not\subseteq$	?	$\subseteq$	$\not\subseteq$	$\not\subseteq$	$\not\subseteq$
SE-CCA	$\not\subseteq$	?	?	?	$\subseteq$	$\subseteq$	?
SE-CPA	$\not\subseteq$	?	$\not\subseteq$	?	$\not\subseteq$	$\subseteq$	$\not\subseteq$
PKE-CCA	$\not\subseteq$	$\not\subseteq$	?	?	$\not\subseteq$	$\not\subseteq$	$\subseteq$

Open Questions!

Research Directions

Construct Φ-RKA-PRFs for other classes:

- $\Phi^{\oplus} = \{ \varphi^{\oplus} \mid \Delta \in \{0,1\}^n \}$ where $\varphi^{\oplus}(k) = k \oplus \Delta$
- $\Phi^{+} = \{ \varphi^{+} \mid \Delta \in \{0,1\}^n \}$ where $\varphi^{+}(k) = k + \Delta \bmod 2^n$

Maybe from LPN via a similar technique?

Assume we have an RKA-PRF. **What can we do with it?**

- More efficient protocols (key derivation, etc)
- New applications?

Security Against Related-Key Attacks: Constructions & Applications

Thanks!