Salvaging Weak Security Bounds for Blockcipher-based Constructions

Thomas Shrimpton (University of Florida)
Seth Terashima (Qualcomm Technologies, inc.)

What weak bounds?

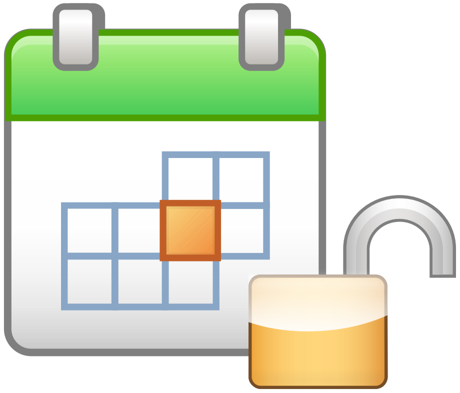
- ...from encrypting lots of data

  Intel Hardware RNG: Single-machine bound on Adversary exceeds 2^{-30} in **four months**, 2^{-40} in **four days**.

  With 1,000 machines (break-one-and-win), Adversary bound exceeds 2^{-20} in four days.

- ...from using small block, key sizes

  Sensor networks, “Internet of Things”



What weak bounds?

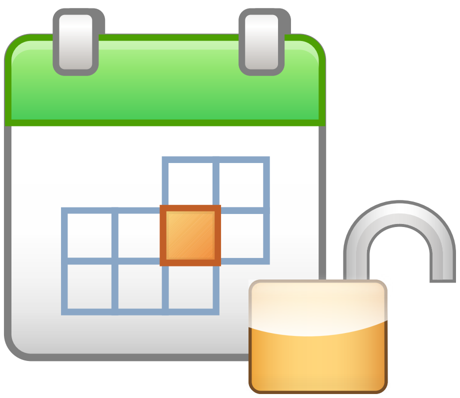
- ...from encrypting lots of data

  Intel Hardware RNG: Single-machine bound on Adversary exceeds 2^{-30} in **four months**, 2^{-40} in **four days**.

  With 1,000 machines (break-one-and-win), Adversary bound exceeds 2^{-20} in four days.

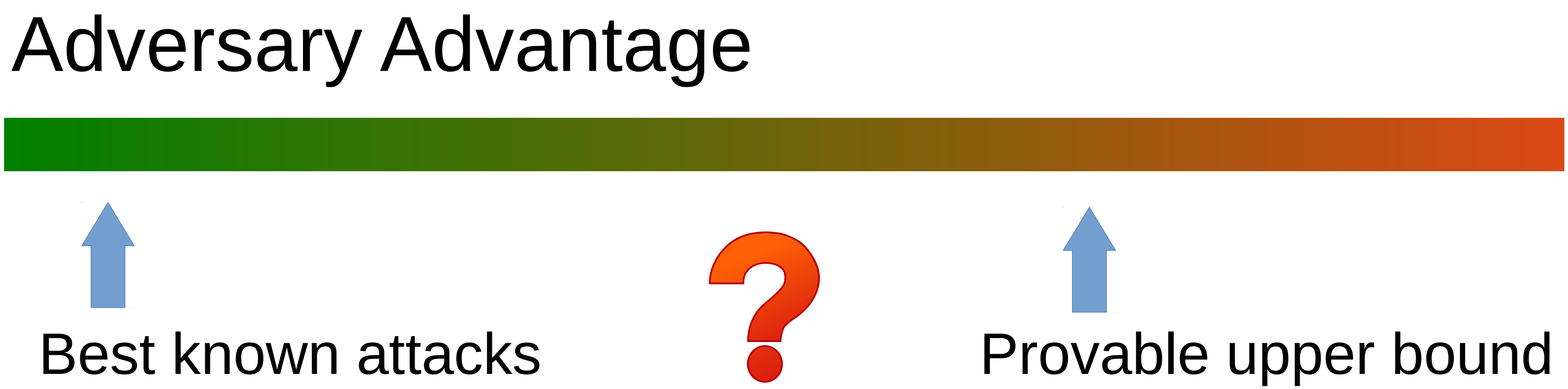
- ...from using small block, key sizes

  Sensor networks, “Internet of Things”



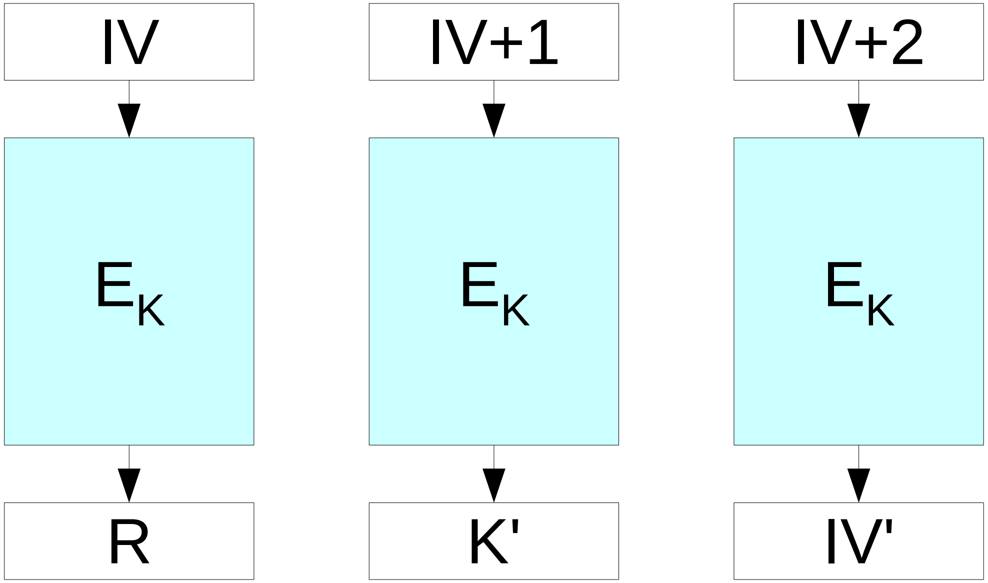
Rekeying can help, but “hybrid arguments” multiply Adversary advantage by number of keys used.

Don't panic.


Adversary Advantage
Best known attacks
Provable upper bound

Case Study: NIST CTR-DRBG

(Counter-mode based deterministic random bit generator)

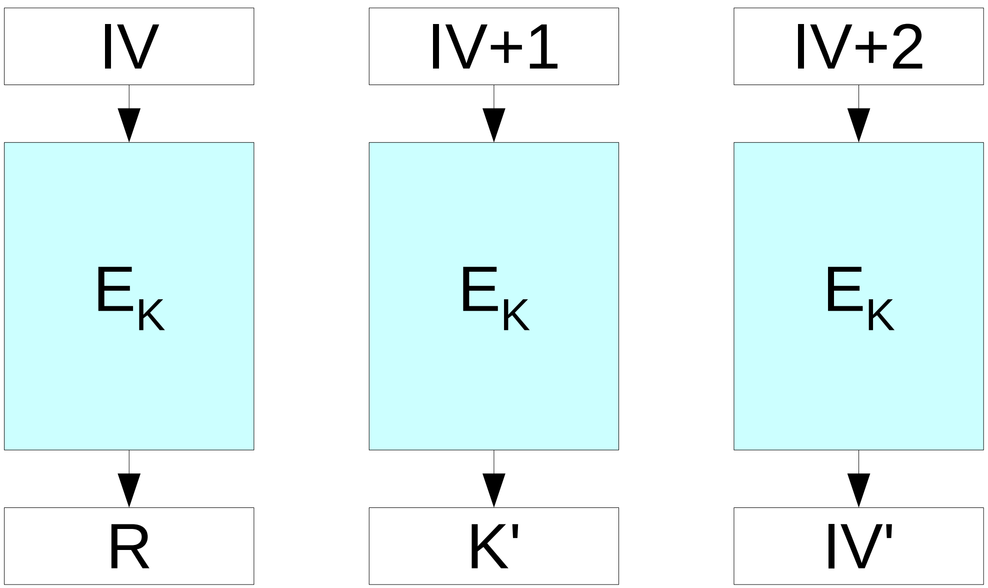


Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

Case Study: NIST CTR-DRBG

(Counter-mode based deterministic random bit generator)



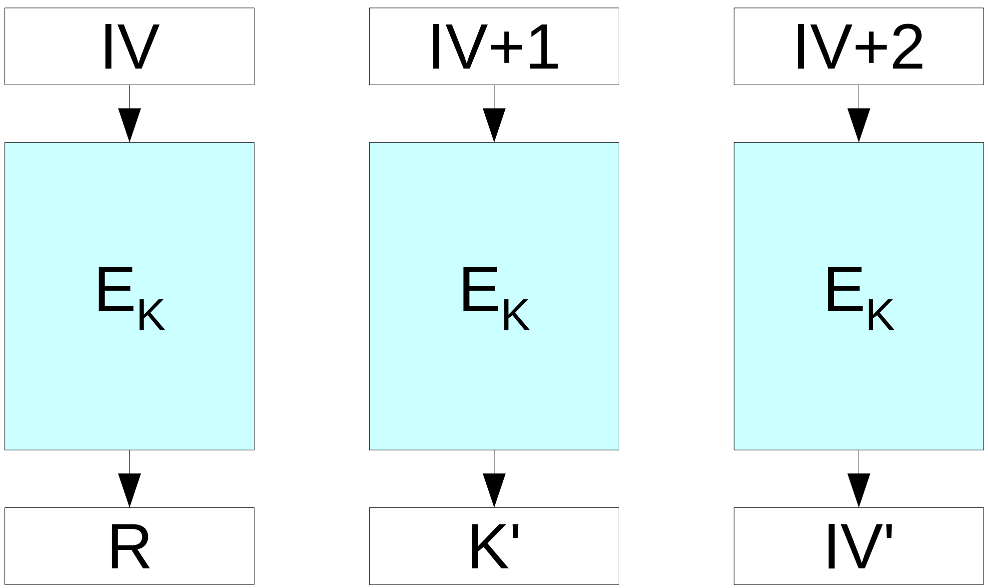
Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

$$\mathsf{Adv}^{\mathsf{DRBG}}_{\mathsf{NIST\text{-}CTR\text{-}DRBG}[E]}(q,t) \le \frac{3}{2^n} + q\mathsf{Adv}^{\mathsf{PRP}}_{E}(3,t)$$

Case Study: NIST CTR-DRBG

(Counter-mode based deterministic random bit generator)



Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

$$\mathsf{Adv}^{\mathsf{DRBG}}_{\mathsf{NIST\text{-}CTR\text{-}DRBG}[E]}(q,t) \le \frac{3}{2^n} + \boxed{q\mathsf{Adv}^{\mathsf{PRP}}_{E}(3,t)}$$

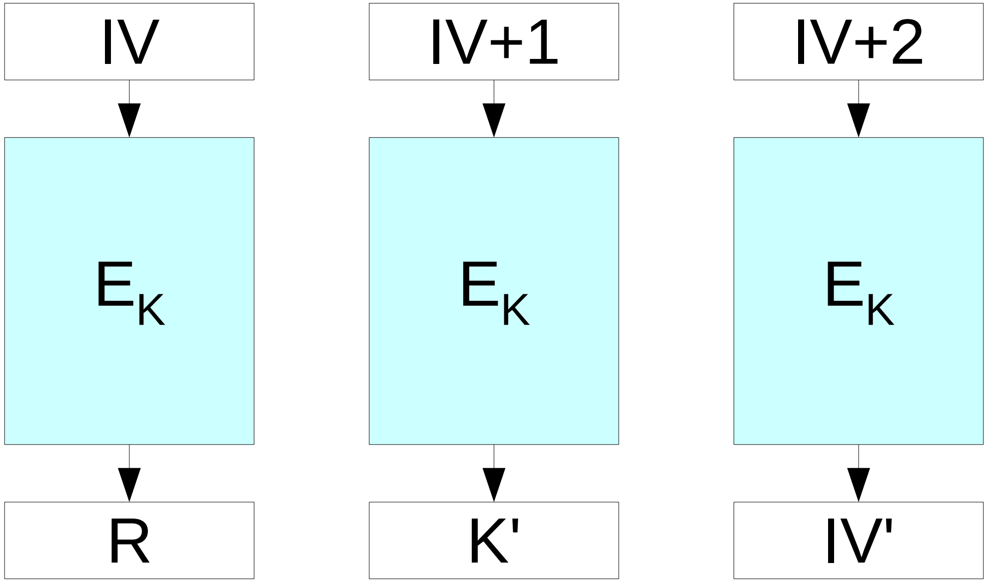
$$\square \approx \frac{tq}{2^k} \approx \frac{q^2}{2^k}$$

Case Study: NIST CTR-DRBG

How tight is this bound?

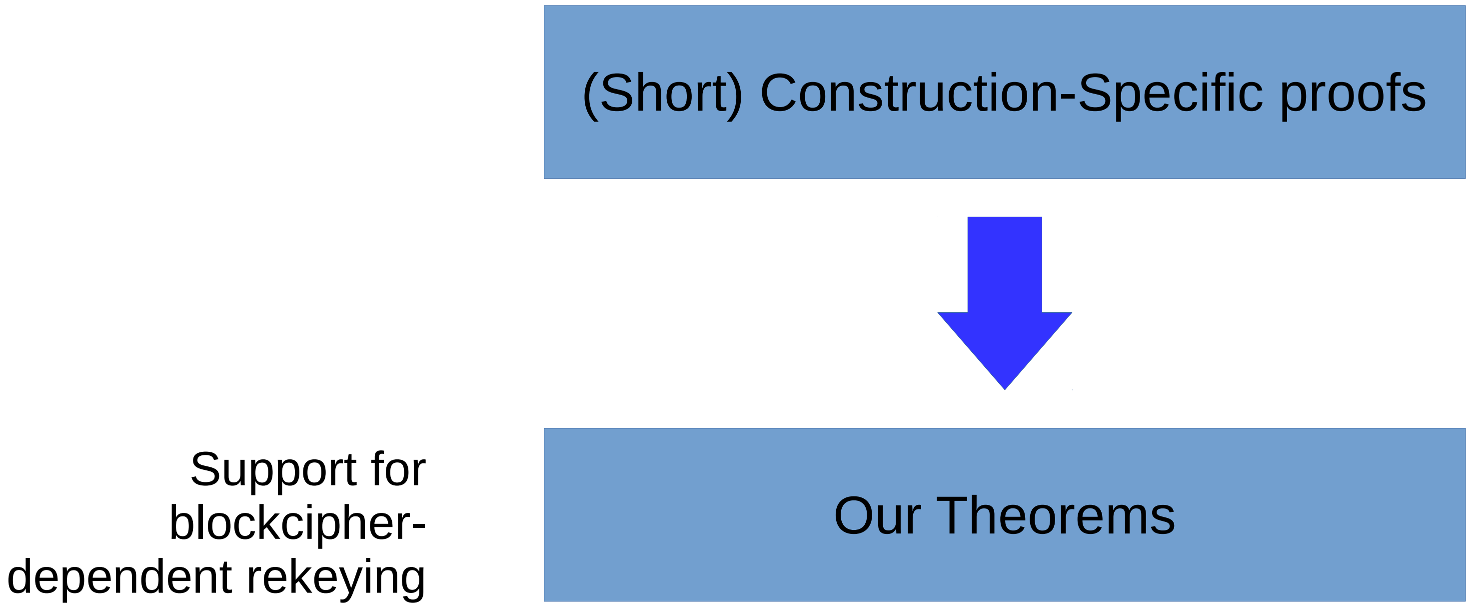
Generic PRP attack on q keys with q time:

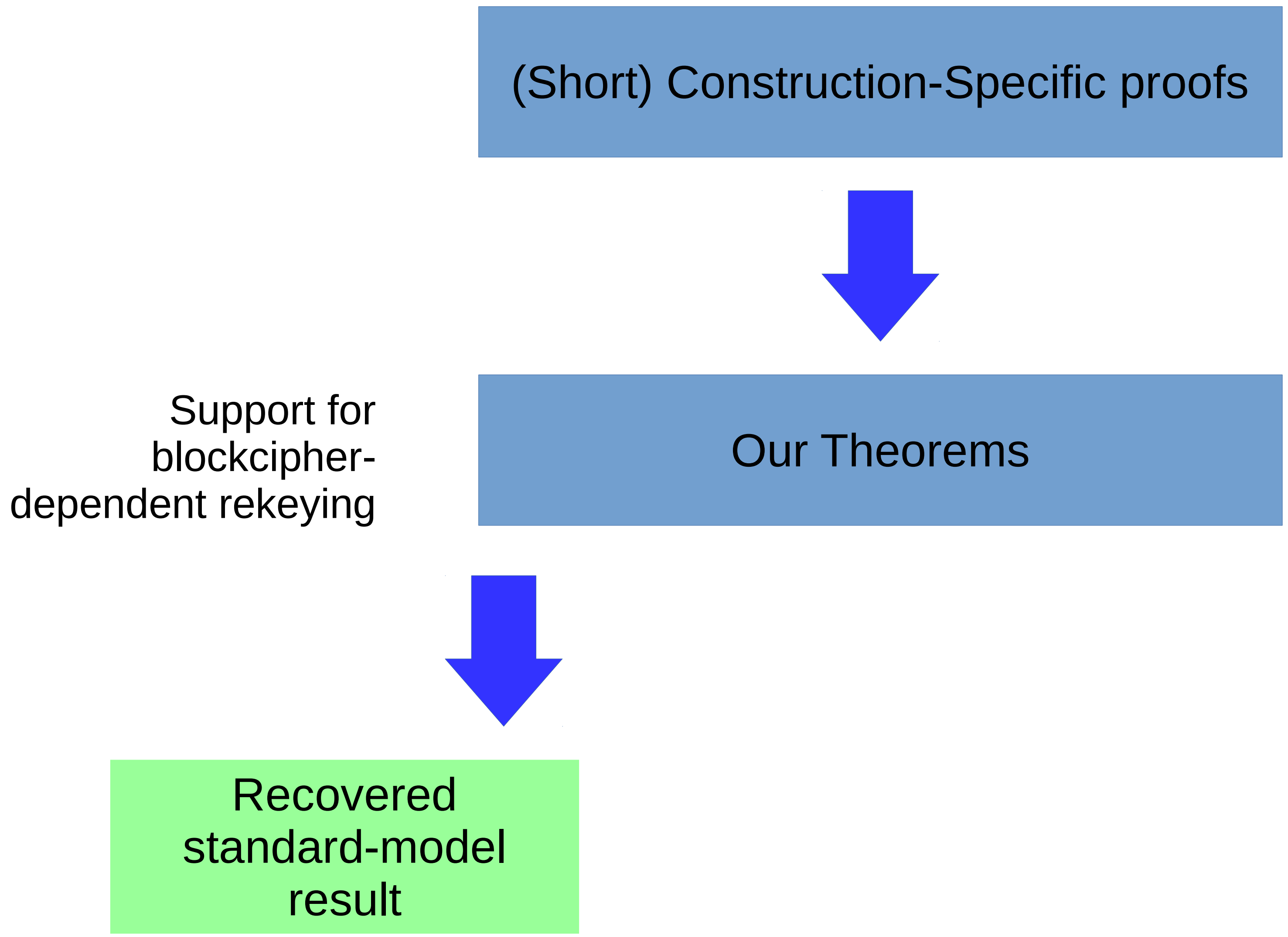
- Encrypt 0^n under each of the q keys
- Choose q distinct keys at random, encrypt 0^n under each
- Look for matches (use a hash table)
- Advantage: ~ $q^2/2^k$

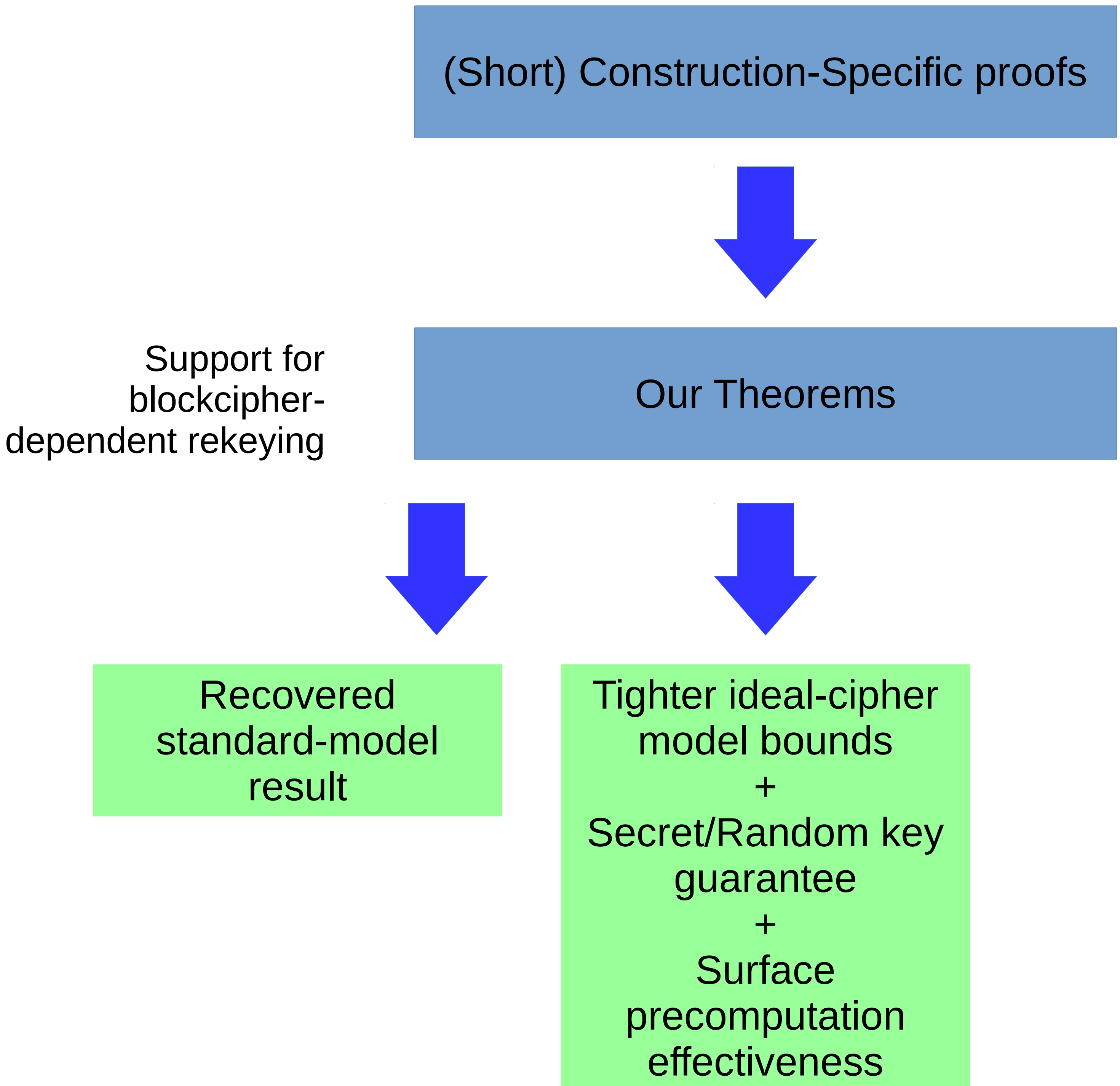


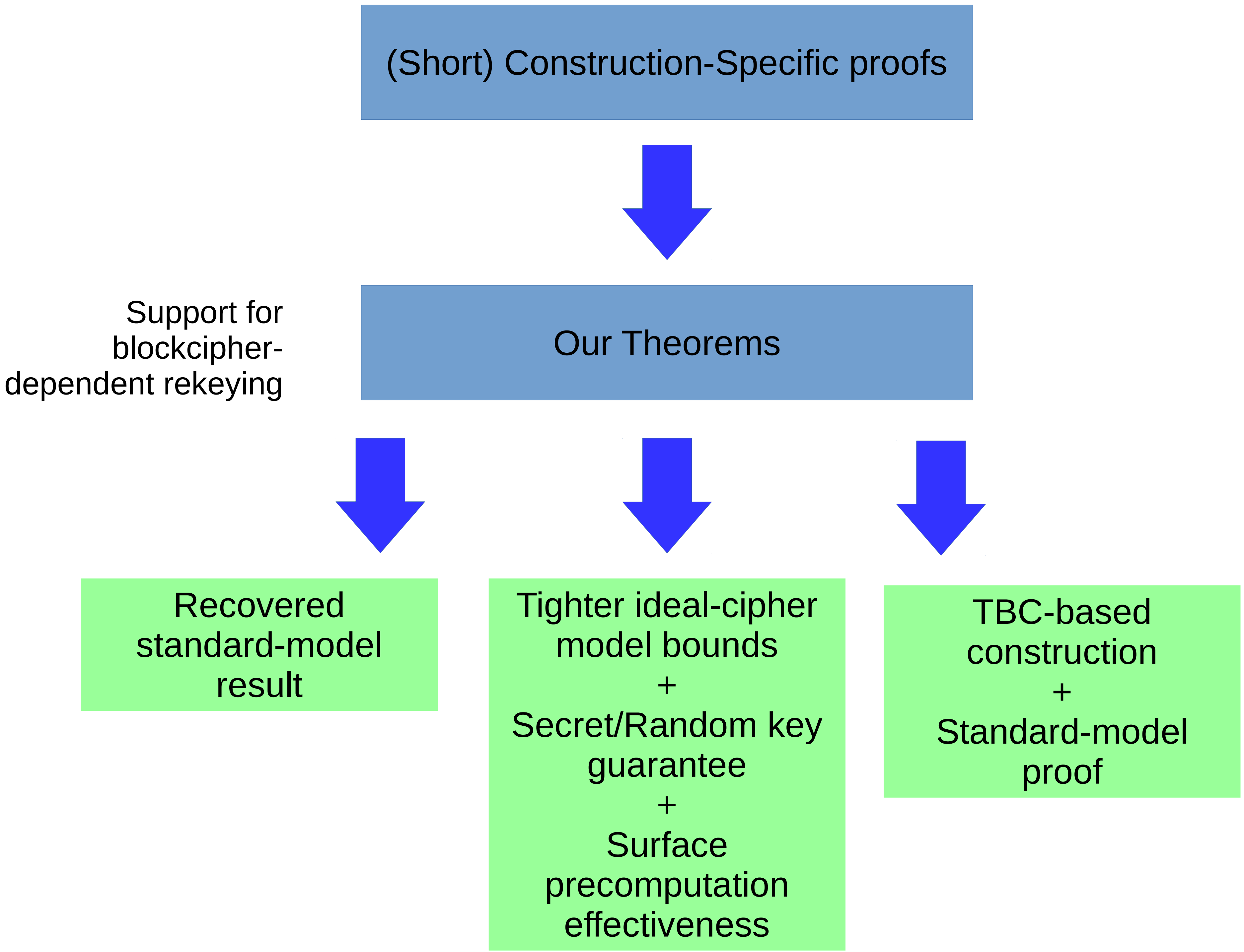
Attack doesn't work here because the **mode of operation prevents it**.

We can't reuse a plaintext, attack q “target” keys simultaneously with a single “test” key.

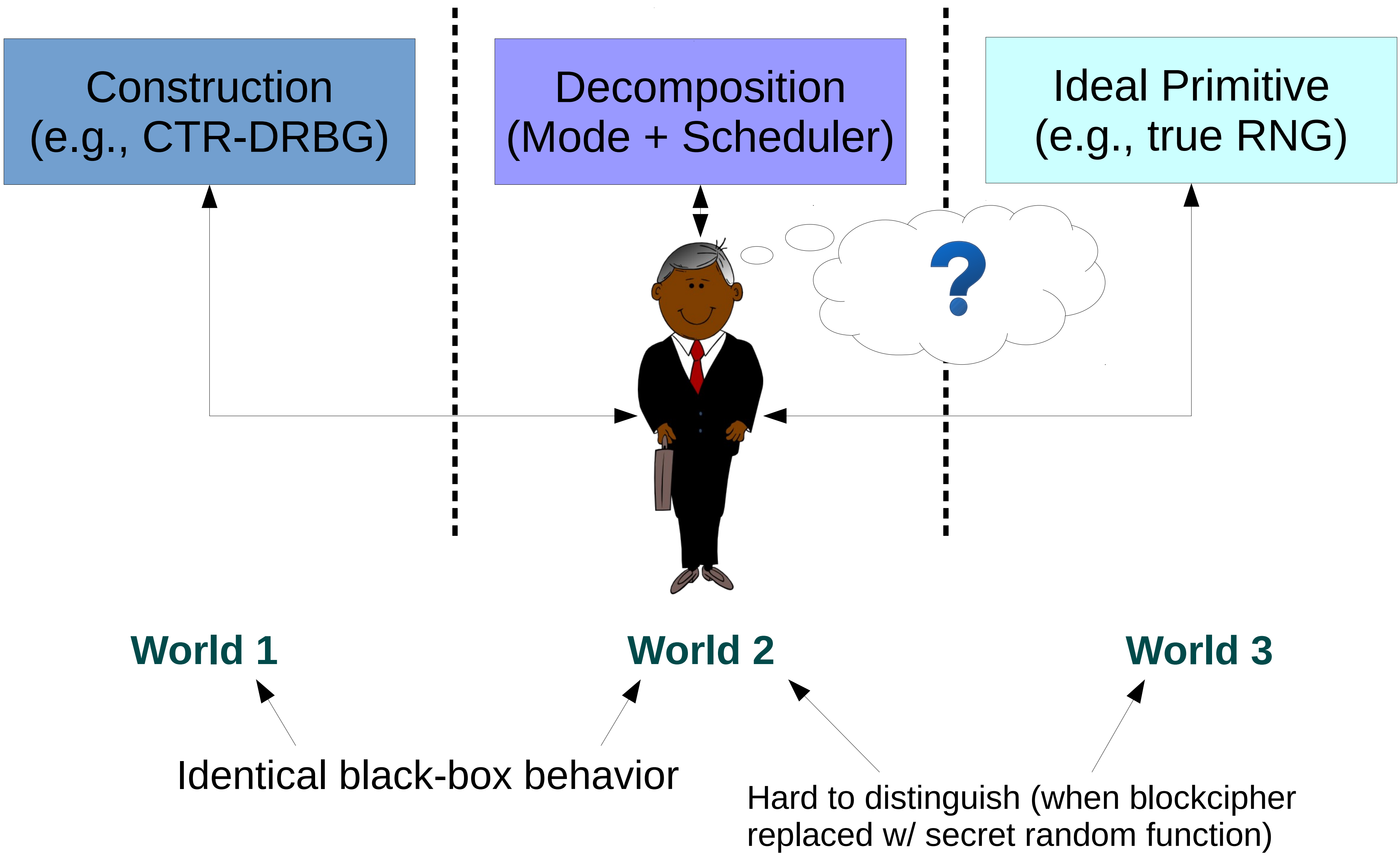

(Short) Construction-Specific proofs
Our Theorems
Support for blockcipher-dependent rekeying


(Short) Construction-Specific proofs
Our Theorems
Support for blockcipher-dependent rekeying
Recovered standard-model result

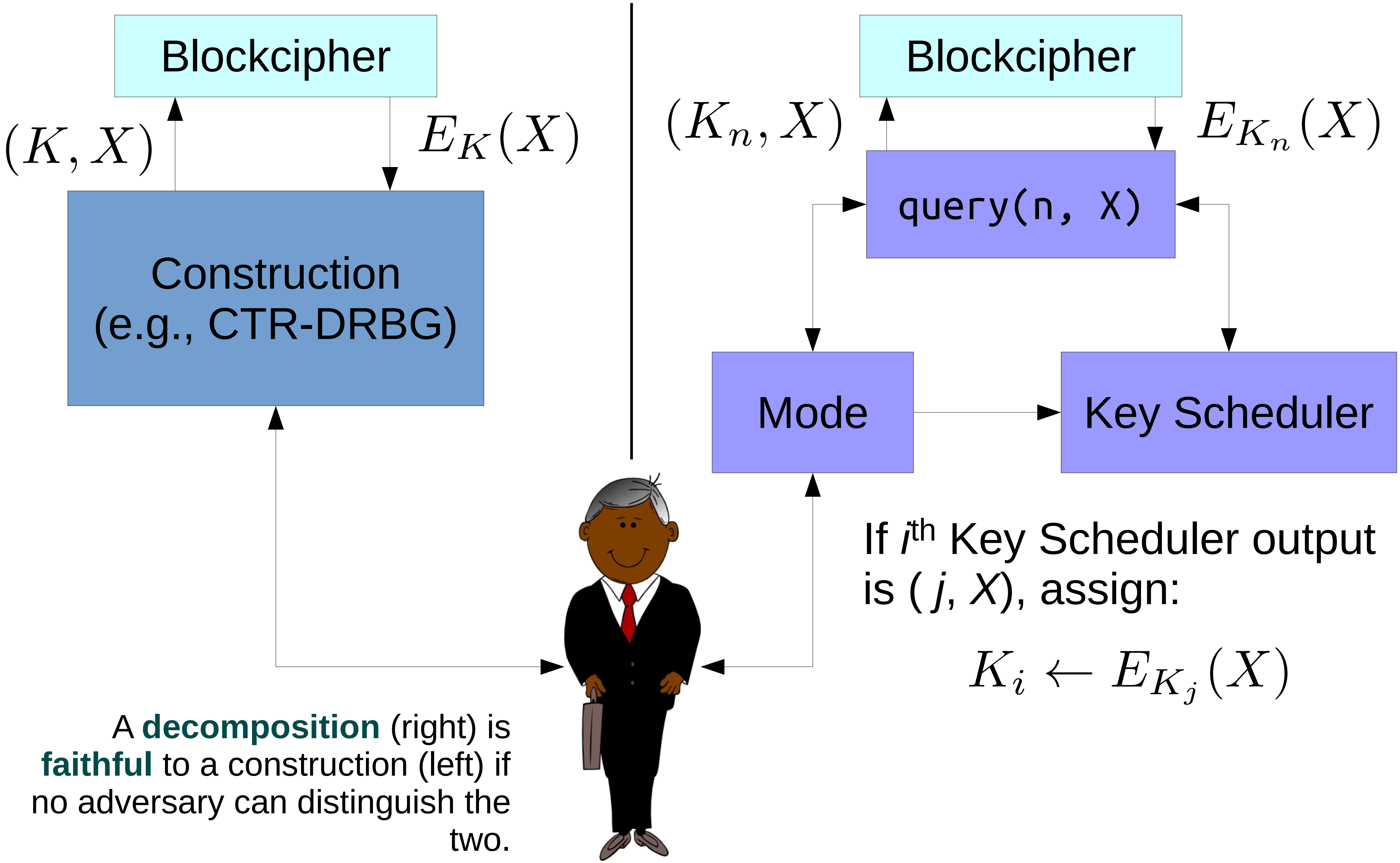

(Short) Construction-Specific proofs
Support for blockcipher-dependent rekeying
Our Theorems
Recovered standard-model result
Tighter ideal-cipher model bounds
+
Secret/Random key guarantee
+
Surface precomputation effectiveness


(Short) Construction-Specific proofs
Our Theorems
Support for blockcipher-dependent rekeying
Recovered standard-model result
Tighter ideal-cipher model bounds
+
Secret/Random key guarantee
+
Surface precomputation effectiveness
TBC-based construction
+
Standard-model proof

ICM with Key-Oblivious Access



Key-Oblivious Access



If i^{th} Key Scheduler output is (j, X), assign:

$$K_i \leftarrow E_{K_j}(X)$$

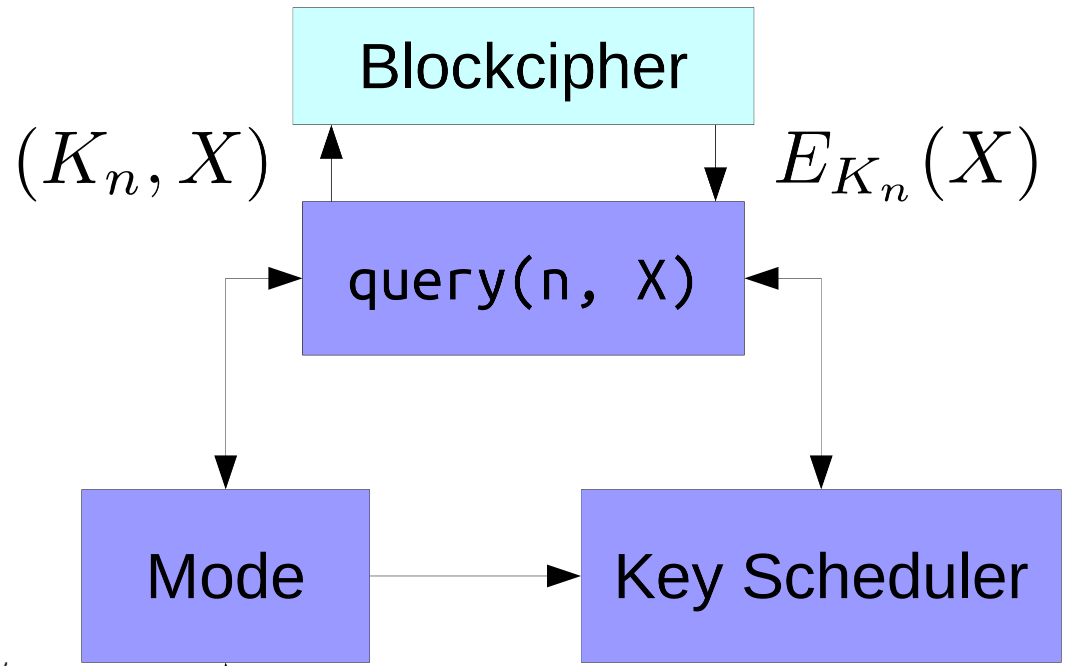
A **decomposition** (right) is **faithful** to a construction (left) if no adversary can distinguish the two.

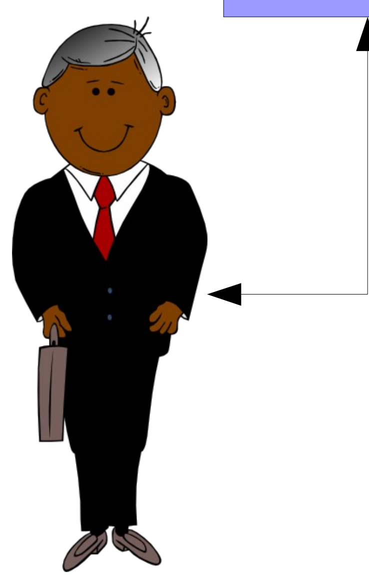
Key-Oblivious Access

A mode is **compatible** with a scheduler if they cannot be forced to evaluate `query` at the same point (n, X).

Only constructions that use random, secret keys have compatible decompositions.

- Allows reduction to standard model
- Guarantees no related keys, weak keys





If i^{th} Key Scheduler output is (j, X), assign:

$$K_i \leftarrow E_{K_j}(X)$$

Using the model

(what you need to do)

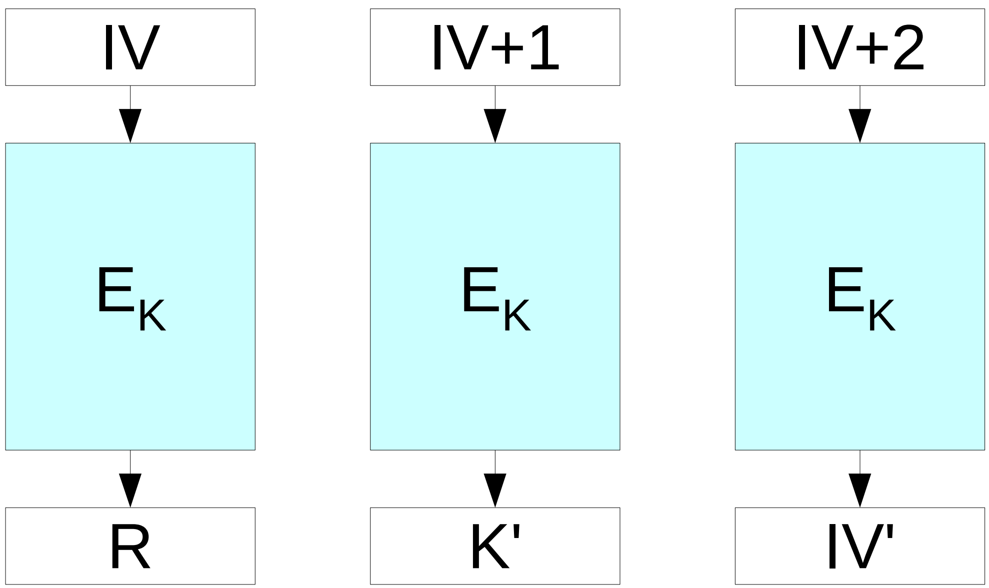
Correctness – Find a compatible decomposition

Efficiency – Bound the number of blockcipher queries made per adversary query, bound number of key handles used

Sparsity – No input block is encrypted under more than μ key handles (except with probability ε)

ICM-KOA Security – Show Adversary has advantage δ when distinguishing decomposition from ideal primitive **when the blockcipher is replaced by a random function that the adversary cannot compute "offline".**

Case Study: NIST CTR-DRBG

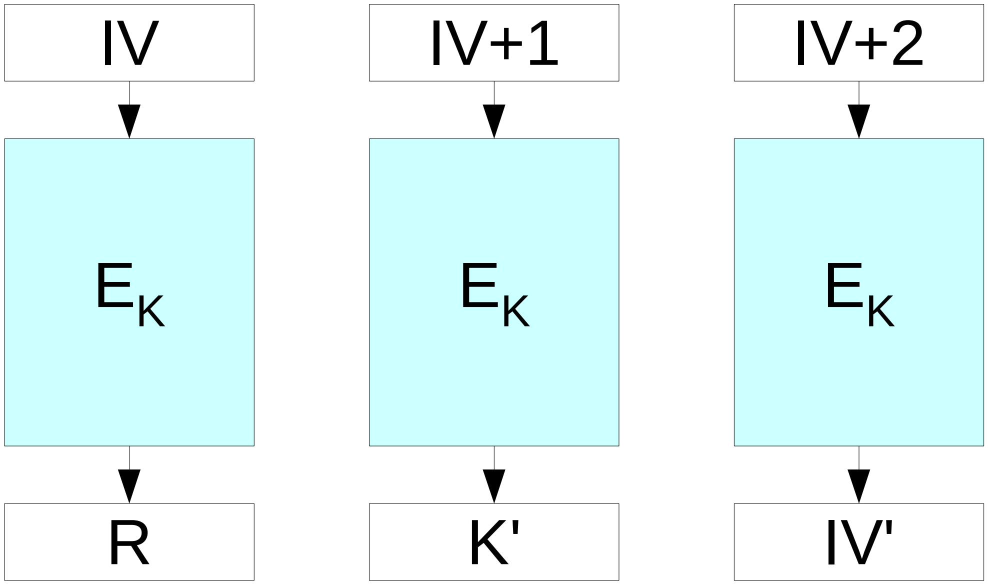


Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

Decomposition: The mode and scheduler both get the initial IV as a key, and track it as part of their respective states.

Case Study: NIST CTR-DRBG

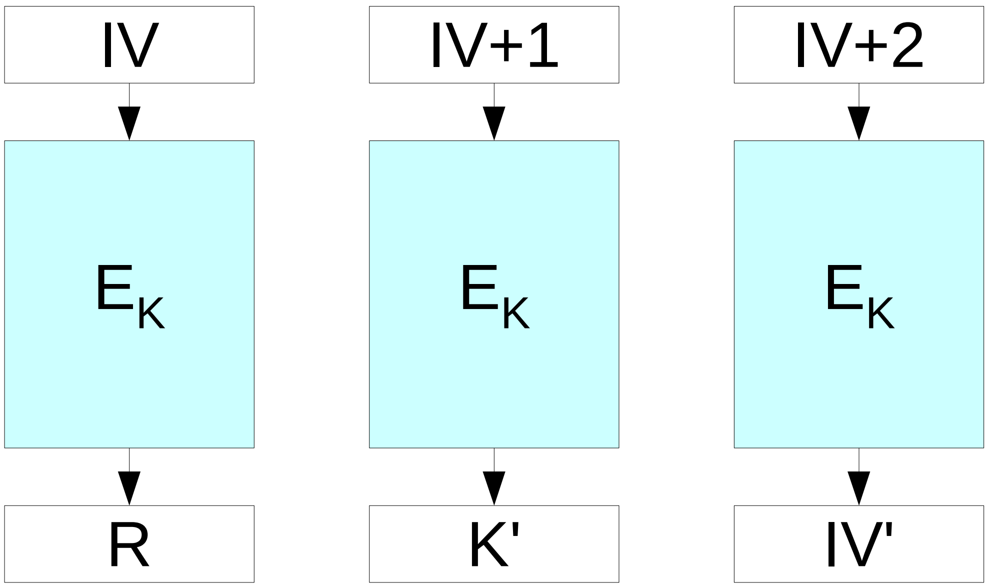


Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

Efficiency: Each key handle is used on three input blocks, and the number of key handles equals the number of adversary queries.

Case Study: NIST CTR-DRBG

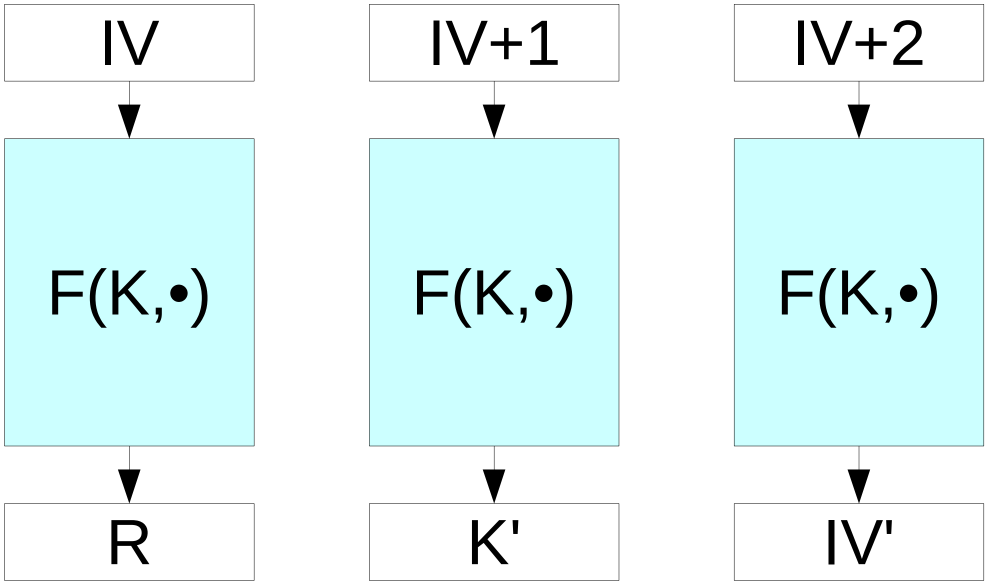


Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

Sparsity: No input block is encrypted under more than *c* key handles, except with probability ~ $(3q)^{c+1}/(2^{cn}(c+1)!)$. (Generalized birthday bound).

Case Study: NIST CTR-DRBG

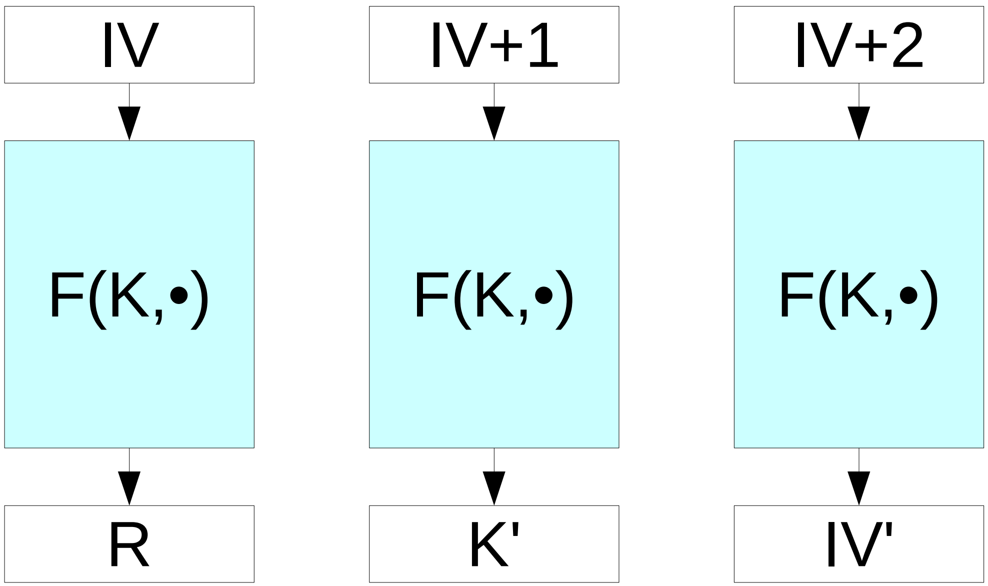


Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

ICM-KOA security: If F is a random function unknown the adversary, then the RNG behaves ideally unless a (K, X) pair is reused. This happens with probability at most $5q^2/2^{2n}$.

Case Study: NIST CTR-DRBG



Initialize with random (K, IV)

On each query:
Update (K, IV) ← (K', IV')
Return R as random value

$$\mathsf{Adv}^{\text{icm-ind-Rand}}_{\text{CTR-DRBG}}(A) \leq \frac{20q^2 + 24q_E + 3q(q_E + q_P) + 19q^3}{2^{2n}} + \frac{20q + 6q_E + 2q_P}{2^n} = \mathcal{O}\left(\frac{q^3}{2^{2n}}\right)$$

q Online queries $\quad q_P$ Precomputation queries $\quad q_E$ Offline queries

Case Study: NIST CTR-DRBG

In this case, the ICM-KOA:

- Recovers the $O(q^2/2^{128})$ standard model bound (**four days** to pass 2^{-40})
- *Also* gives an ICM result of **748,229 years** (2^{80} offline queries)

More generally, the ICM-KOA:

- Models blockcipher-dependent rekeying
- Gives a standard-model proof
- Offers tighter ICM bounds while forcing random + secret keys
- Quantifies effectiveness of precomputation, offline queries
- Implies standard-model security of a TBC-based construction

…for a small, single effort.

Questions?

Also in the paper: analysis of rekeyed-counter mode variants, and some general results about multi-instance distinguishability games.