On the Security of Keyed Hashing

Joan Daemen (based on joint work with Jonathan Fuchs and Yann Rotella)
Radboud University (The Netherlands)

ISC Winter School on Information Security and Cryptology, February 24, 2021

Outline

Deck functions and some modes

An alternative for block-cipher based crypto

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output
   - goal : F_K behaves like a random oracle $\mathcal{RO}$

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output
   - goal : F_K behaves like a random oracle $\mathcal{RO}$
   - PRF distinguishing advantage $\epsilon_p(M, N)$ assumed to be small

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output
   - goal : F_K behaves like a random oracle $\mathcal{RO}$
   - PRF distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts

An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output
   - goal : F_K behaves like a random oracle $\mathcal{RO}$
   - PRF distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build encryption or authentication mode of a random oracle
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it

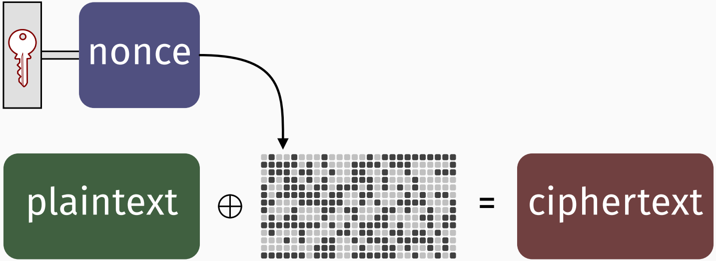
An alternative for block-cipher based crypto

1. Instead of a block cipher, construct a *deck function* F_K
   - F_K has arbitrary-length input and output
   - goal : F_K behaves like a random oracle $\mathcal{RO}$
   - PRF distinguishing advantage $\epsilon_p(M, N)$ assumed to be small
   - assurance: based on public scrutiny by cryptanalysts
2. Build encryption or authentication mode of a random oracle
   - prove upper bound $\epsilon_m(M, N)$ for probability of breaking it

Security of mode with concrete F_K

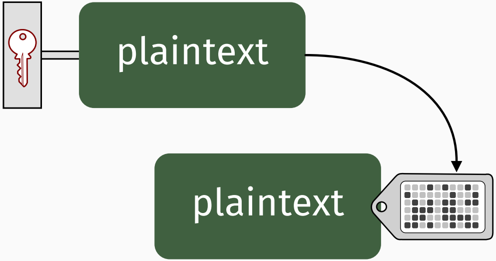
Breaking probability $\leq \epsilon_m(M, N) + \epsilon_p(M, N)$

Stream encryption: short input, long output

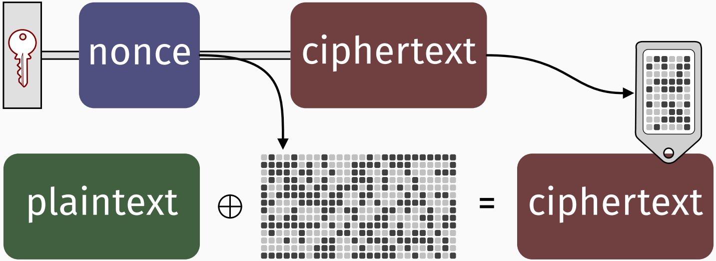


$$C \leftarrow P + F_K(N)$$

MAC computation: long input, short output

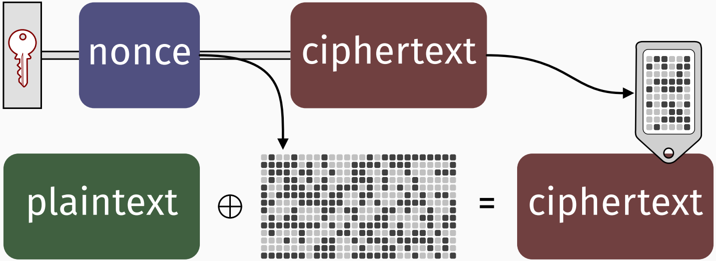


$$T \leftarrow 0^t + F_K(P)$$



$$C = P + F_K(N), \quad T = 0^t + F_K(C \circ N)$$

Authenticated encryption (AE)



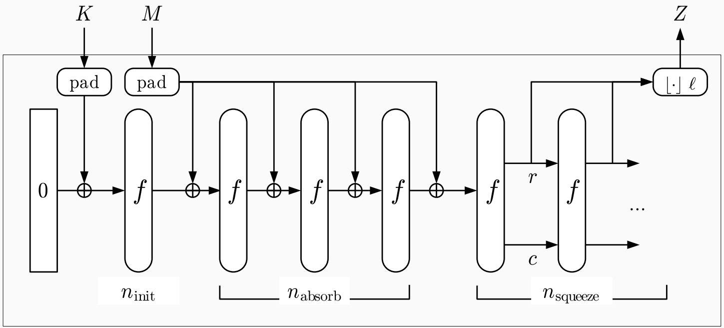
$$C = P + F_K(N), \quad T = 0^t + F_K(C \circ N)$$

…and much more, see Youtube video of All on Deck! [Keccak Team, RWC 2020]

`https://www.youtube.com/watch?v=CQDsLhf-d-A` at minute 30

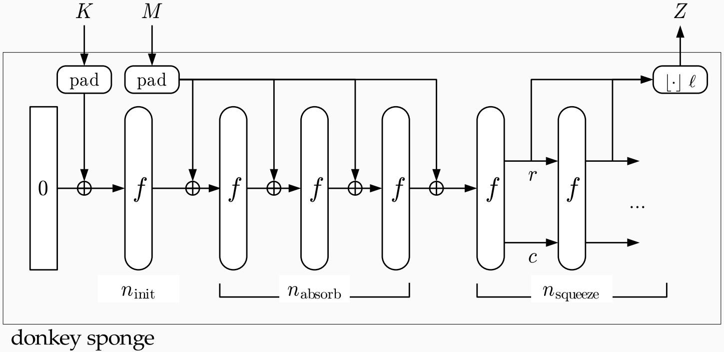
How to build a deck function?

A serial deck function construction: donkeySponge [KT, DIAC 2012]

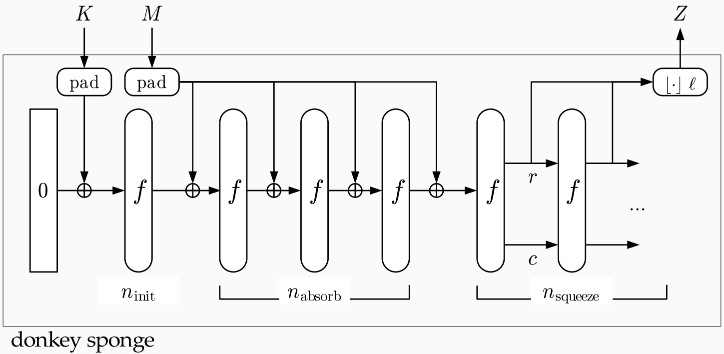


donkey sponge

A serial deck function construction: donkeySponge [KT, DIAC 2012]



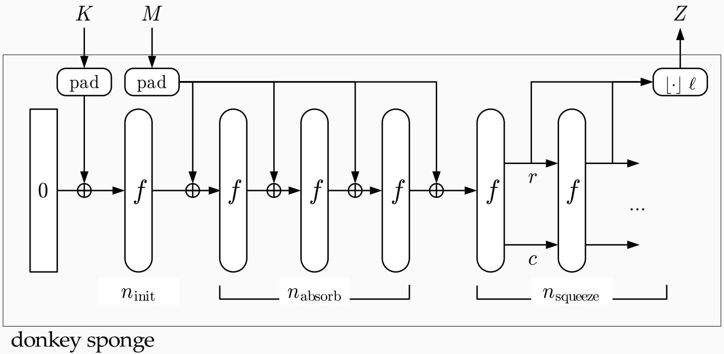
Sponge with secret key K as initial state



Sponge with secret key K as initial state

- Compression of input blocks into state: *full-state* sponge absorbing

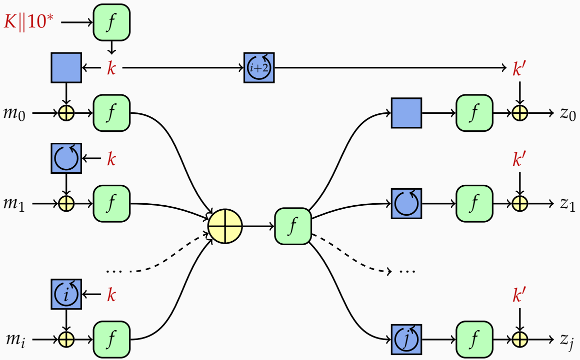
A serial deck function construction: donkeySponge [KT, DIAC 2012]



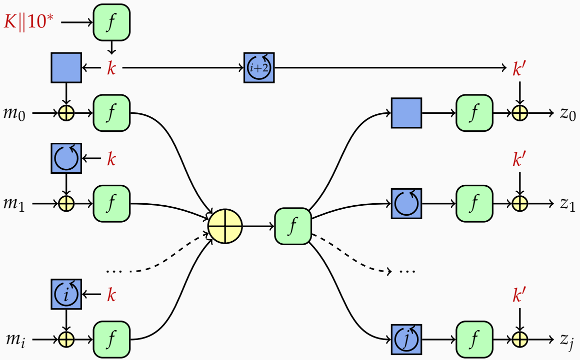
Sponge with secret key K as initial state

- Compression of input blocks into state: *full-state* sponge absorbing
- Expansion of state to output stream: standard sponge squeezing

A parallel deck function construction: Farfalle [KT, ToSC 2017]

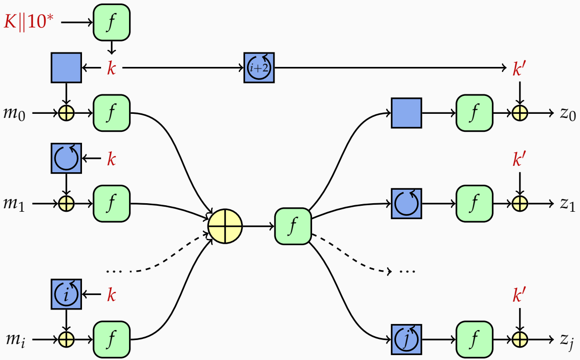


A parallel deck function construction: Farfalle [KT, ToSC 2017]



Expands secret key K to secret *rolling* mask k

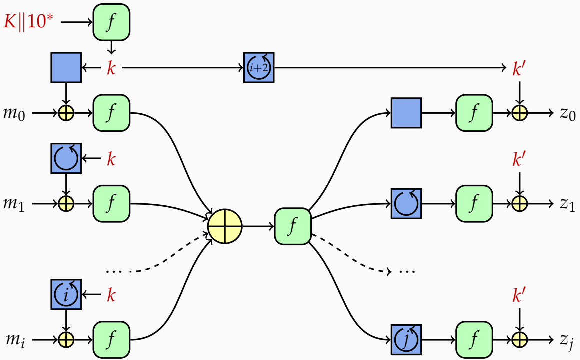
A parallel deck function construction: Farfalle [KT, ToSC 2017]



Expands secret key K to secret *rolling* mask k

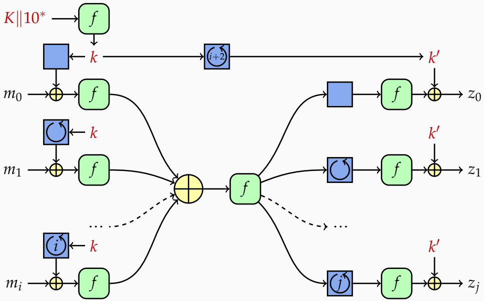
- Compression of *masked* input blocks into *accumulator*

A parallel deck function construction: Farfalle [KT, ToSC 2017]



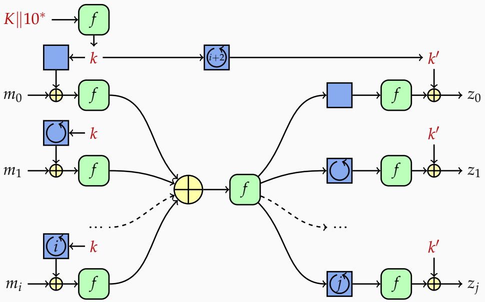
Expands secret key K to secret *rolling* mask k

- Compression of *masked* input blocks into *accumulator*
- Expansion: *rolling state* filtered by f and secret mask



To design you need to understand the attacks. Three types:

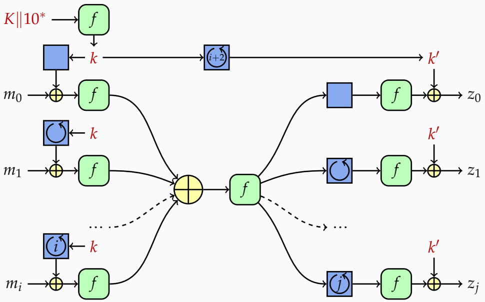
Attacks on a deck function



To design you need to understand the attacks. Three types:

- Using both input and output: as block cipher attacks

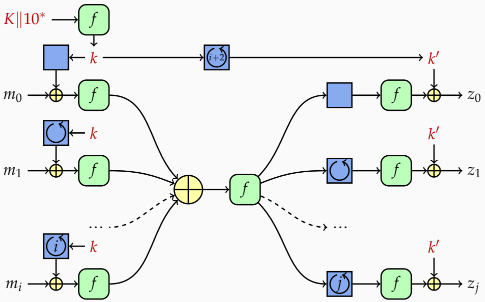
Attacks on a deck function



To design you need to understand the attacks. Three types:

- Using both input and output: as block cipher attacks
- Output-only: as classical stream cipher attacks

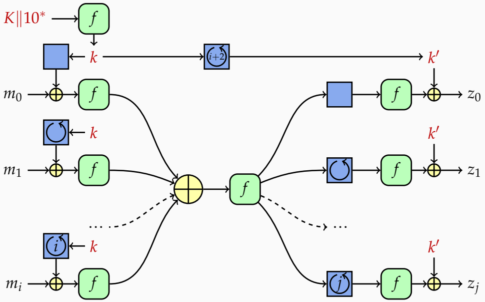
Attacks on a deck function



To design you need to understand the attacks. Three types:

- Using both input and output: as block cipher attacks
- Output-only: as classical stream cipher attacks
- Input-only: accumulator collisions

Attacks on a deck function



To design you need to understand the attacks. Three types:

- Using both input and output: as block cipher attacks
- Output-only: as classical stream cipher attacks
- Input-only: accumulator collisions: **this presentation**

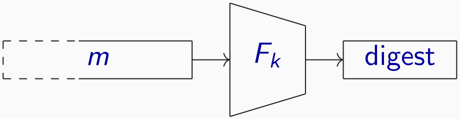
Keyed hashing

Keyed Hashing

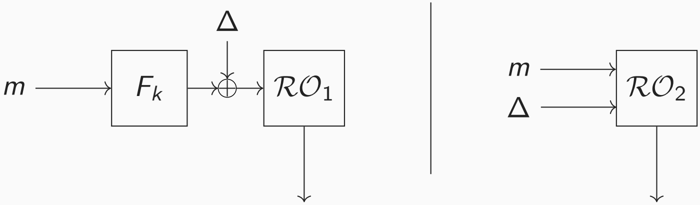
- $F : \mathcal{K} \times \mathcal{M} \to \mathcal{A}$
  - $\mathcal{K}$: key space
  - $\mathcal{M}$: message space
  - $\mathcal{A}$: digest space, forms an additive *group* and $\mathcal{A} \lll \mathcal{M}$

Keyed Hashing

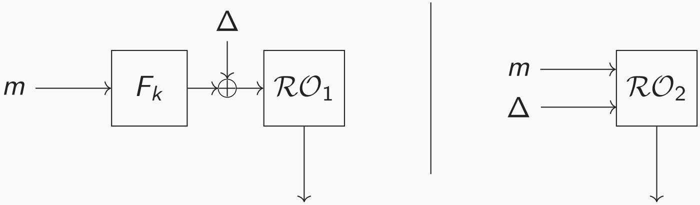
- $F : \mathcal{K} \times \mathcal{M} \to \mathcal{A}$
    - $\mathcal{K}$: key space
    - $\mathcal{M}$: message space
    - $\mathcal{A}$: digest space, forms an additive *group* and $\mathcal{A} \lll \mathcal{M}$
- Convention: F_k denotes F with a fixed key $k \in \mathcal{K}$



Security notion: blinded keyed hash security

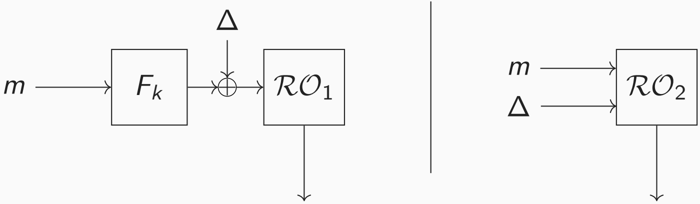


Security notion: blinded keyed hash security



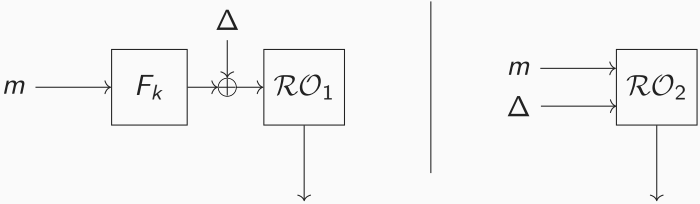
- Distinguishing setup with attacker that can send queries (m, Δ) to either:
  - real world: F_k followed by secret $\mathcal{RO}_1$
  - ideal world: secret $\mathcal{RO}_2$

Security notion: blinded keyed hash security



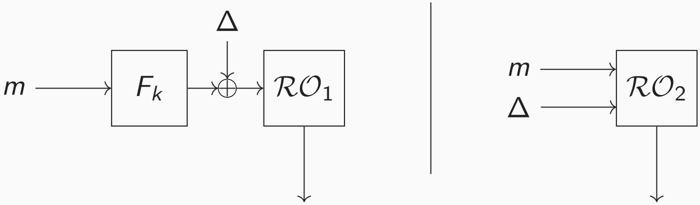
- Distinguishing setup with attacker that can send queries (m, Δ) to either:
  - real world: F_k followed by secret $\mathcal{RO}_1$
  - ideal world: secret $\mathcal{RO}_2$
- Only way to distinguish: collision at input of $\mathcal{RO}_1$

Security notion: blinded keyed hash security



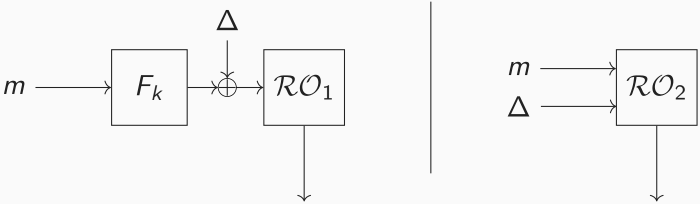
- Distinguishing setup with attacker that can send queries (m, Δ) to either:
  - real world: F_k followed by secret $\mathcal{RO}_1$
  - ideal world: secret $\mathcal{RO}_2$
- Only way to distinguish: collision at input of $\mathcal{RO}_1$
  - success probability independent of attacker's computational resources

Security notion: blinded keyed hash security

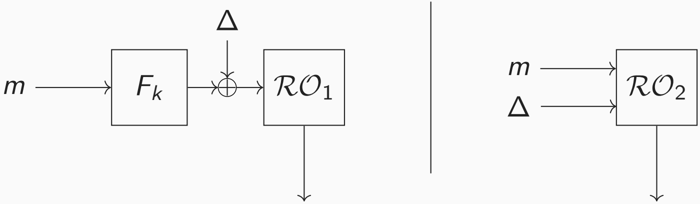


- Distinguishing setup with attacker that can send queries (m, Δ) to either:
  - real world: F_k followed by secret $\mathcal{RO}_1$
  - ideal world: secret $\mathcal{RO}_2$
- Only way to distinguish: collision at input of $\mathcal{RO}_1$
  - success probability independent of attacker's computational resources
  - adaptability does not help so attacker can fix queries in advance

Blinded keyed hash security and deck functions



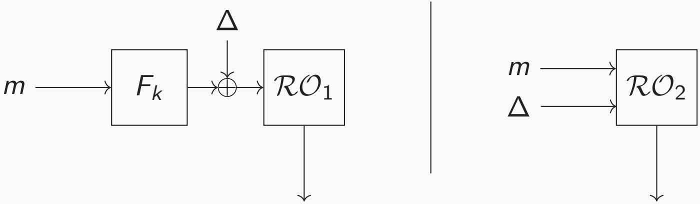
Applied to deck functions:



Applied to deck functions:

- This expresses the security against input-only attacks on compression phase

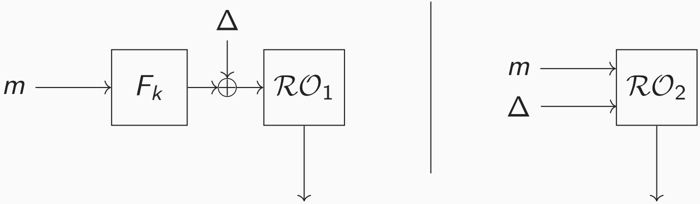
Blinded keyed hash security and deck functions



Applied to deck functions:

- This expresses the security against input-only attacks on compression phase
- We model the expansion phase as an independent $\mathcal{RO}_1$

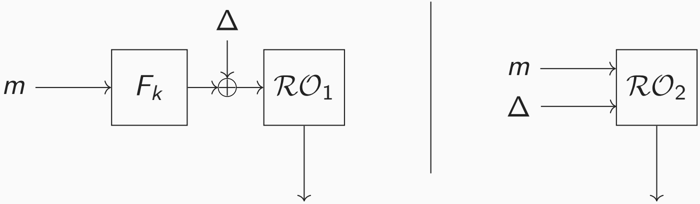
Blinded keyed hash security and deck functions



Applied to deck functions:

- This expresses the security against input-only attacks on compression phase
- We model the expansion phase as an independent $\mathcal{RO}_1$
- $\Delta = 0$ (but $\Delta \neq 0$ is meaningful in reductions and other scenario's)

Blinded keyed hash security and deck functions



Applied to deck functions:

- This expresses the security against input-only attacks on compression phase
- We model the expansion phase as an independent $\mathcal{RO}_1$
- $\Delta = 0$ (but $\Delta \neq 0$ is meaningful in reductions and other scenario's)
- We study the collision probability of sets of queries AKA *message sets* D

Collision Probability and security strength

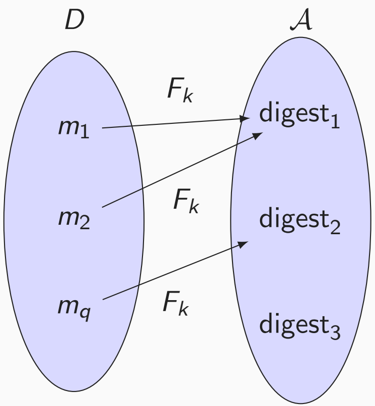
- F_k maps messages in D to digests,

Collision Probability and security strength

- F_k maps messages in D to digests, all different ones . . .

Collision Probability and security strength

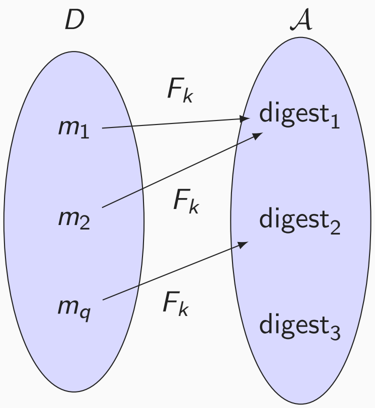
- F_k maps messages in D to digests, all different ones ...
  or not: this a *collision* in $F_k(D)$



Collision Probability and security strength

- F_k maps messages in D to digests, all different ones ... or not: this a *collision* in $F_k(D)$
- Solution set of D:

$$\mathcal{S}(D) = \{k \in \mathcal{K} \mid \text{collision in } F_k(D)\}$$



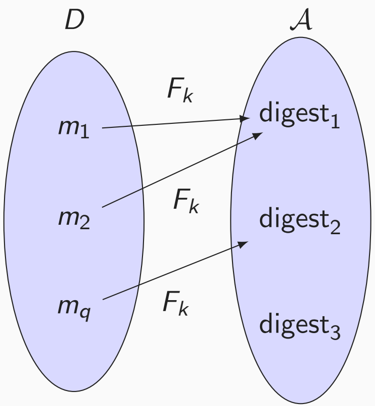
Collision Probability and security strength

- F_k maps messages in D to digests, all different ones ... or not: this a *collision* in $F_k(D)$
- Solution set of D:

$$\mathcal{S}(D) = \{k \in \mathcal{K} \mid \text{collision in } F_k(D)\}$$

- Collision probability of a message set:

$$\mathrm{CP}_F(D) = \frac{\#\mathcal{S}(D)}{\#\mathcal{K}}$$



Collision Probability and security strength

- F_k maps messages in D to digests, all different ones ...
  or not: this a *collision* in $F_k(D)$
- Solution set of D:

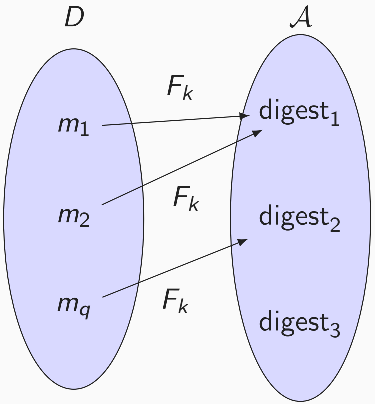
$$\mathcal{S}(D) = \{k \in \mathcal{K} \mid \text{collision in } F_k(D)\}$$

- Collision probability of a message set:

$$\mathrm{CP}_F(D) = \frac{\#\mathcal{S}(D)}{\#\mathcal{K}}$$

- Collision probability limit

$$\mathrm{CPL}(q) = \max_{D \text{ with } \#D=q} \mathrm{CP}_F(D)$$

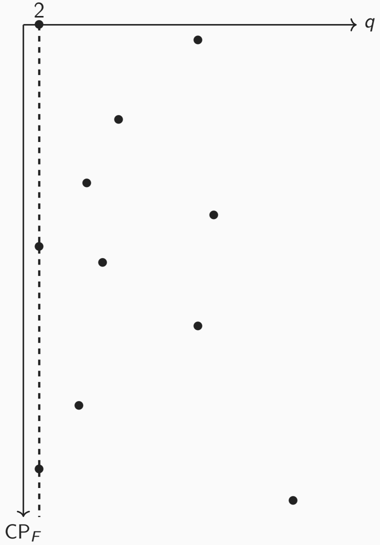


Graphical view

Graphical view

- We can position message sets D as points in a plane

$$(x, y) = (\#D, \mathrm{CP}_F(D))$$

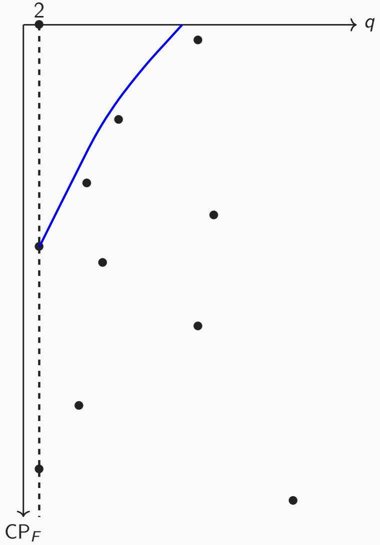


Graphical view

- We can position message sets D as points in a plane

$$(x, y) = (\#D, \mathrm{CP}_F(D))$$

- $\mathrm{CPL}(q)$ is the envelope of all these points



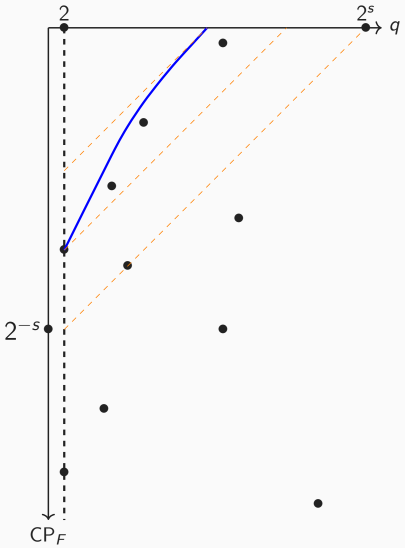
Graphical view

- We can position message sets D as points in a plane

$$(x, y) = (\#D, \mathrm{CP}_F(D))$$

- $\mathrm{CPL}(q)$ is the envelope of all these points
- Security strength s:

$$s = \min_D \left(\log_2(\#D) - \log_2(\mathrm{CP}_F(D))\right)$$



Graphical view

- We can position message sets D as points in a plane

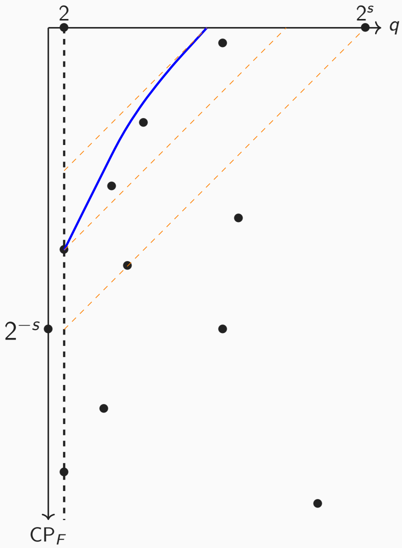
$$(x, y) = (\#D, \mathrm{CP}_F(D))$$

- $\mathrm{CPL}(q)$ is the envelope of all these points
- Security strength s:

$$s = \min_D \left(\log_2(\#D) - \log_2(\mathrm{CP}_F(D))\right)$$

- $\mathrm{CPL}(q)$ defines security strength

$$s = \min_q \left(\log_2(q) - \log_2(\mathrm{CPL}(q))\right)$$



Graphical view

- We can position message sets D as points in a plane

$$(x, y) = (\#D, \mathrm{CP}_F(D))$$

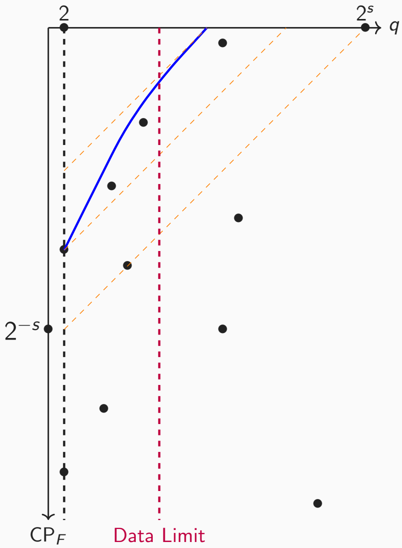
- $\mathrm{CPL}(q)$ is the envelope of all these points
- Security strength s:

$$s = \min_D \left(\log_2(\#D) - \log_2(\mathrm{CP}_F(D))\right)$$

- $\mathrm{CPL}(q)$ defines security strength

$$s = \min_q \left(\log_2(q) - \log_2(\mathrm{CPL}(q))\right)$$

- In real-world settings q may be limited



The Birthday Bound

The Birthday Bound

- Let F be a random function

- Let F be a random function
- Let D be 2 random messages in $\mathcal{M}$

$$\mathrm{CP}_F(D) = \frac{1}{\#\mathcal{A}}$$

The Birthday Bound

- Let F be a random function
- Let D be 2 random messages in $\mathcal{M}$

$$\mathrm{CP}_F(D) = \frac{1}{\#\mathcal{A}}$$

- Let D be q random messages in $\mathcal{M}$

$$\mathrm{CP}_F(D) \approx \binom{q}{2} \frac{1}{\#\mathcal{A}}$$

- Let F be a random function
- Let D be 2 random messages in $\mathcal{M}$

$$\mathrm{CP}_F(D) = \frac{1}{\#\mathcal{A}}$$

- Let D be q random messages in $\mathcal{M}$

$$\mathrm{CP}_F(D) \approx \binom{q}{2} \frac{1}{\#\mathcal{A}}$$

- For an actual function it is worse, so:

$$\mathrm{CPL}(q) \geq \binom{q}{2} \frac{1}{\#\mathcal{A}}$$

The Birthday Bound

- Let F be a random function
- Let D be 2 random messages in $\mathcal{M}$

$$\mathsf{CP}_F(D) = \frac{1}{\#\mathcal{A}}$$

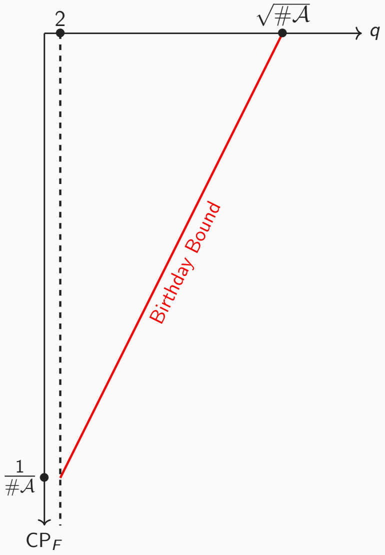
- Let D be q random messages in $\mathcal{M}$

$$\mathsf{CP}_F(D) \approx \binom{q}{2} \frac{1}{\#\mathcal{A}}$$

- For an actual function it is worse, so:

$$\mathsf{CPL}(q) \geq \binom{q}{2} \frac{1}{\#\mathcal{A}}$$

- This is the *Birthday Bound*



The Quadratic Limit

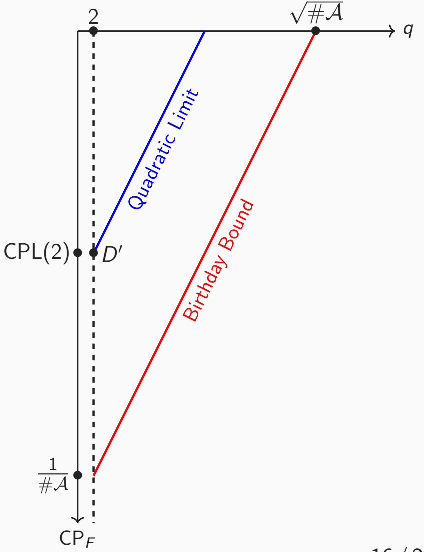
Often we know $\mathrm{CPL}(v)$ for small values of v

The Quadratic Limit

Often we know CPL(v) for small values of v

- For $v = 2$ it is easy to show that

$$\mathrm{CPL}(q) \le \binom{q}{2}\mathrm{CPL}(2)$$

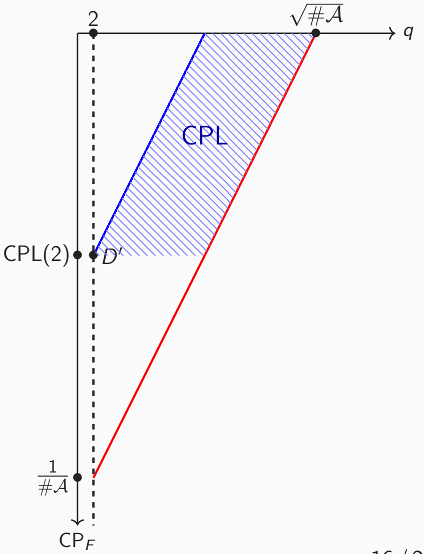


The Quadratic Limit

Often we know $\mathrm{CPL}(v)$ for small values of v

- For $v = 2$ it is easy to show that

$$\mathrm{CPL}(q) \leq \binom{q}{2} \mathrm{CPL}(2)$$



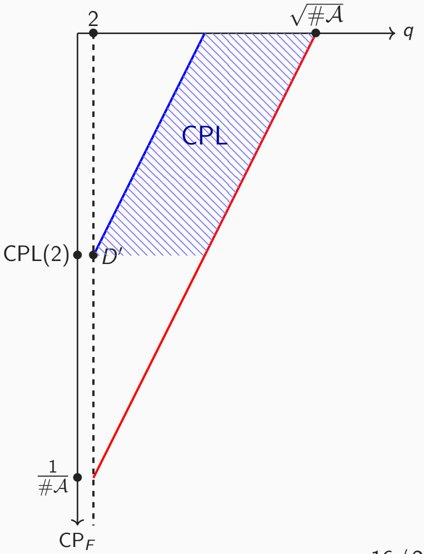
The Quadratic Limit

Often we know $\mathrm{CPL}(v)$ for small values of v

- For $v = 2$ it is easy to show that

$$\mathrm{CPL}(q) \leq \binom{q}{2} \mathrm{CPL}(2)$$

- Equality on two conditions:

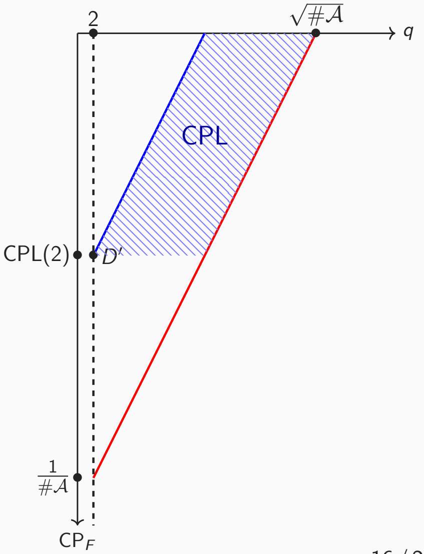


Often we know $\mathrm{CPL}(v)$ for small values of v

- For $v = 2$ it is easy to show that

$$\mathrm{CPL}(q) \leq \binom{q}{2} \mathrm{CPL}(2)$$

- Equality on two conditions:
  - $\exists D$ with all $\binom{q}{2}$ pairs D' having $\mathrm{CPL}(2)$: it is *collision-dense*



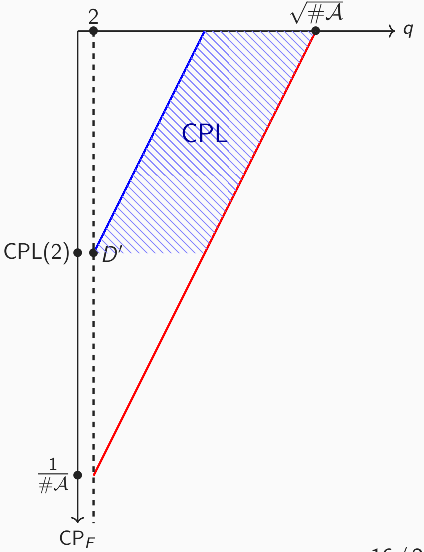
The Quadratic Limit

Often we know $\mathsf{CPL}(v)$ for small values of v

- For $v = 2$ it is easy to show that

$$\mathsf{CPL}(q) \leq \binom{q}{2} \mathsf{CPL}(2)$$

- Equality on two conditions:
  - $\exists D$ with all $\binom{q}{2}$ pairs D' having $\mathsf{CPL}(2)$: it is *collision-dense*
  - the $\mathcal{S}(D')$ are disjunct (*union bound*)



The Quadratic Limit

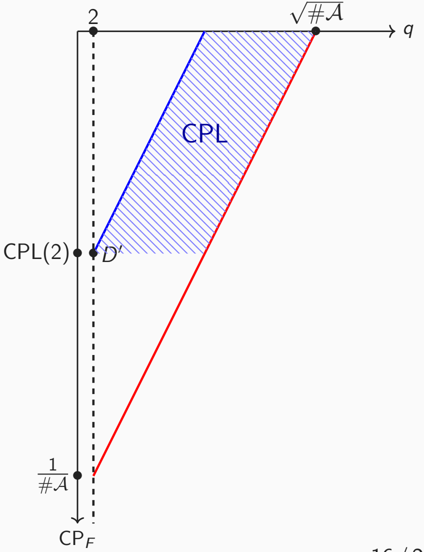
Often we know $\mathsf{CPL}(v)$ for small values of v

- For $v = 2$ it is easy to show that

$$\mathsf{CPL}(q) \leq \binom{q}{2} \mathsf{CPL}(2)$$

- Equality on two conditions:
  - $\exists D$ with all $\binom{q}{2}$ pairs D' having $\mathsf{CPL}(2)$: it is *collision-dense*
  - the $\mathcal{S}(D')$ are disjunct (*union bound*)
- We **prove** that in general, for any $q > v > 1$

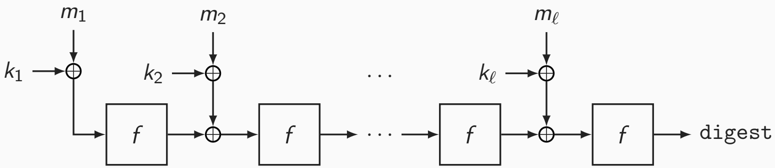
$$\mathsf{CPL}(q) \leq \frac{q(q-1)}{v(v-1)} \mathsf{CPL}(v)$$



Two concrete constructions

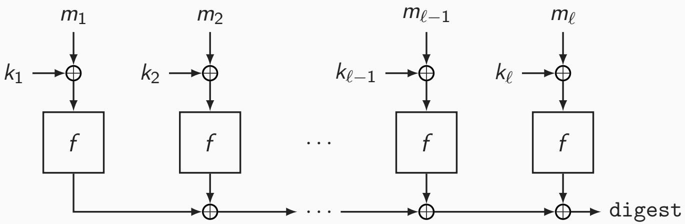
Serial construction

- From a *block function* $f : G \to G$ ...
- we build a keyed compression function $F_{\text{serial}} : \mathcal{K} \times \mathcal{M} \to \mathcal{A}$ with
  - $\mathcal{K} = G^\kappa$
  - $\mathcal{A} = G$
  - $\mathcal{M} = \bigcup_{\ell=1}^{\kappa} G^\ell$
- f is typically a permutation, but not necessarily



Parallel construction

- From a *block function* $f : G \to G'$ ...
- we build a keyed compression function $F_{\mathsf{serial}} : \mathcal{K} \times \mathcal{M} \to \mathcal{A}$ with
  - $\mathcal{K} = G^\kappa$
  - $\mathcal{A} = G'$
  - $\mathcal{M} = \bigcup_{\ell=1}^{\kappa} G^\ell$
- f is typically a permutation, but not necessarily



Two-message attacks: CPL(2)

Two-message attacks: CPL(2)

- We **prove** that for f a permutation, the best attacks have equal-length messages

Two-message attacks: CPL(2)

- We **prove** that for f a permutation, the best attacks have equal-length messages
- Definitions:
  - Fixed-length CPL: $\mathrm{CPL}_n(2)$ denotes $\mathrm{CPL}(2)$ for messages in $G^n \subset \mathcal{M}$

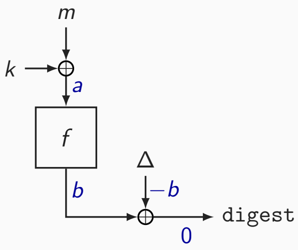
Two-message attacks: CPL(2)

- We **prove** that for f a permutation, the best attacks have equal-length messages
- Definitions:
  - Fixed-length CPL: $\mathrm{CPL}_n(2)$ denotes $\mathrm{CPL}(2)$ for messages in $G^n \subset \mathcal{M}$
  - Differential probability: $\mathrm{DP}_f(a, b) = \Pr(f(m + k + a) - f(m + k) = b)$

Two-message attacks: CPL(2)

- We **prove** that for f a permutation, the best attacks have equal-length messages
- Definitions:
  - Fixed-length CPL: $\mathsf{CPL}_n(2)$ denotes $\mathsf{CPL}(2)$ for messages in $G^n \subset \mathcal{M}$
  - Differential probability: $\mathsf{DP}_f(a, b) = \Pr(f(m+k+a) - f(m+k) = b)$
- We **prove** for both constructions $\mathsf{CPL}_n(2) \leq \mathsf{CPL}_{n-1}(2)$ for $n > 1$

- We **prove** that for f a permutation, the best attacks have equal-length messages
- Definitions:
  - Fixed-length CPL: $\mathsf{CPL}_n(2)$ denotes $\mathsf{CPL}(2)$ for messages in $G^n \subset \mathcal{M}$
  - Differential probability: $\mathsf{DP}_f(a,b) = \Pr(f(m+k+a) - f(m+k) = b)$
- We **prove** for both constructions $\mathsf{CPL}_n(2) \leq \mathsf{CPL}_{n-1}(2)$ for $n > 1$
- We **prove** for both constructions: $\mathsf{CPL}(2) = \max_{a,b} \mathsf{DP}_f(a,b)$



Two-message attacks: $\mathsf{CPL}(2)$ with limitation $\Delta = 0$

Two-message attacks: CPL(2) with limitation $\Delta = 0$

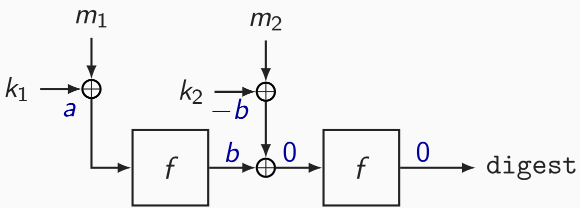
- In deck functions we have $\Delta = 0$

Two-message attacks: $\mathrm{CPL}(2)$ with limitation $\Delta = 0$

- In deck functions we have $\Delta = 0$
- We **prove** if $\Delta = 0$, then $\mathrm{CPL}(2) = \mathrm{CPL}_2(2)$ with

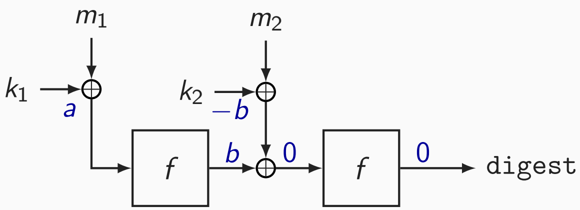
Two-message attacks: $\mathrm{CPL}(2)$ with limitation $\Delta = 0$

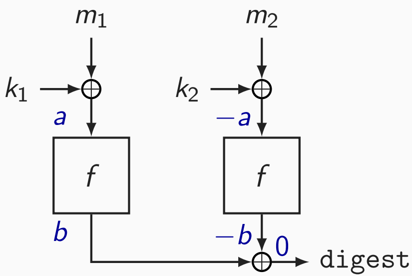
- In deck functions we have $\Delta = 0$
- We **prove** if $\Delta = 0$, then $\mathrm{CPL}(2) = \mathrm{CPL}_2(2)$ with
  - serial construction: $\mathrm{CPL}_2(2) = \max_{a,b} \mathrm{DP}_f(a, b)$



Two-message attacks: CPL(2) with limitation $\Delta = 0$

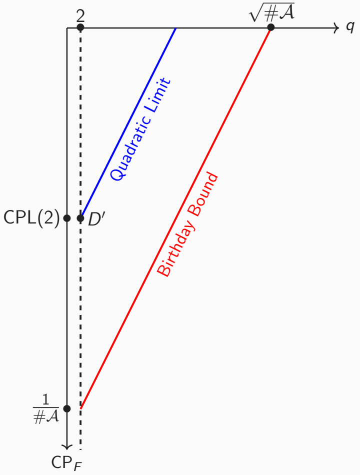
- In deck functions we have $\Delta = 0$
- We **prove** if $\Delta = 0$, then $\mathrm{CPL}(2) = \mathrm{CPL}_2(2)$ with
  - serial construction: $\mathrm{CPL}_2(2) = \max_{a,b} \mathrm{DP}_f(a, b)$
  - parallel construction: $\mathrm{CPL}_2(2) = \max_a \sum_b (\mathrm{DP}_f(a, b))^2$



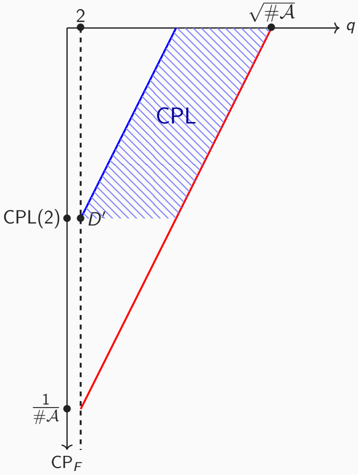


What can we conclude from CPL(2)?

What can we conclude from CPL(2)?



What can we conclude from CPL(2)?

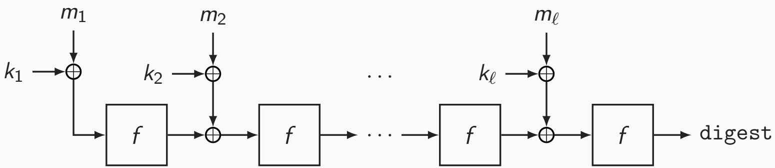


Intermezzo: offset invariance

Definition: a message set D offset by μ: $D + \mu = \{m + \mu \mid m \in D\}$

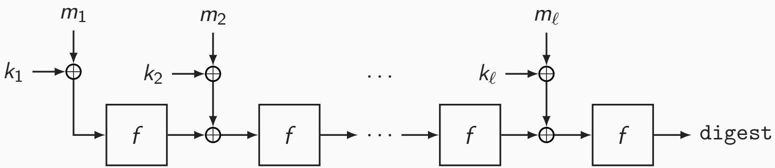
Intermezzo: offset invariance

Definition: a message set D offset by μ: $D + \mu = \{m + \mu \mid m \in D\}$



Intermezzo: offset invariance

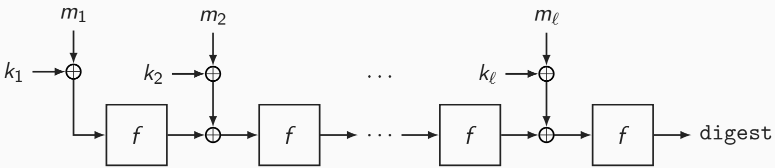
Definition: a message set D offset by μ: $D + \mu = \{m + \mu \mid m \in D\}$



We **prove**, for both the serial and the parallel construction:

Intermezzo: offset invariance

Definition: a message set D offset by μ: $D + \mu = \{m + \mu \mid m \in D\}$



We **prove**, for both the serial and the parallel construction:

Lemma (offset-invariance)

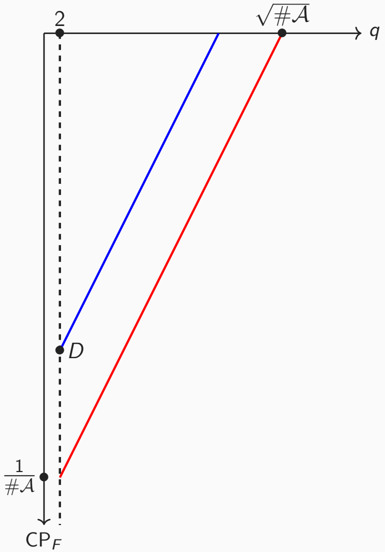
The collision probability is invariant under an offsets of the message set

$$\forall \mu \in G^\kappa : \mathrm{CP}_F(D + \mu) = \mathrm{CP}_F(D)$$

Linear extension of a message set

For $D' = D \cup (D + \mu)$ with μ random

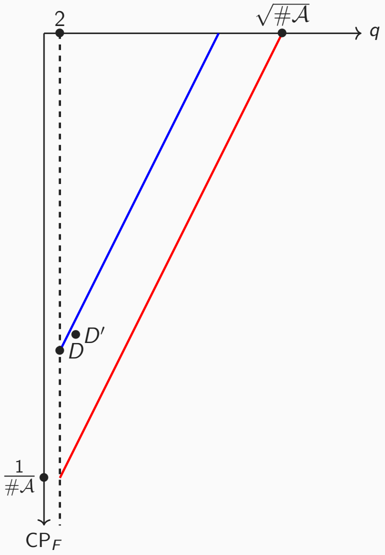
$\mathrm{E}(\mathrm{CP}(D')) \geq 1 - (1 - \mathrm{CP}(D))^2 \approx 2\mathrm{CP}(D)$



Linear extension of a message set

For $D' = D \cup (D + \mu)$ with μ random

$\mathrm{E}(\mathsf{CP}(D')) \geq 1 - (1 - \mathsf{CP}(D))^2 \approx 2\mathsf{CP}(D)$



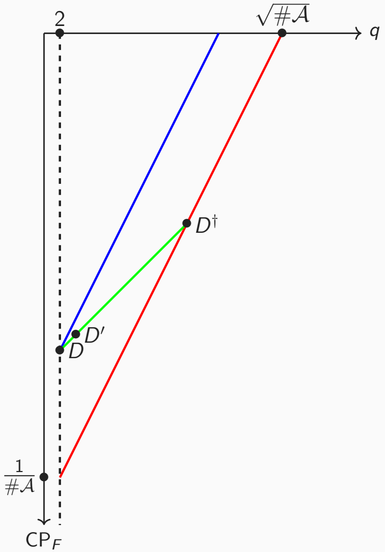
For $D' = D \cup (D + \mu)$ with μ random

$$\mathrm{E}(\mathrm{CP}(D')) \geq 1 - (1 - \mathrm{CP}(D))^2 \approx 2\mathrm{CP}(D)$$

In general for $D' = \bigcup_{1 \leq i \leq n} D + \mu_i$

$$\mathrm{E}(\mathrm{CP}(D')) \approx n\mathrm{CP}(D)$$

... up to the birthday bound ($D^\dagger$)



Linear extension of a message set

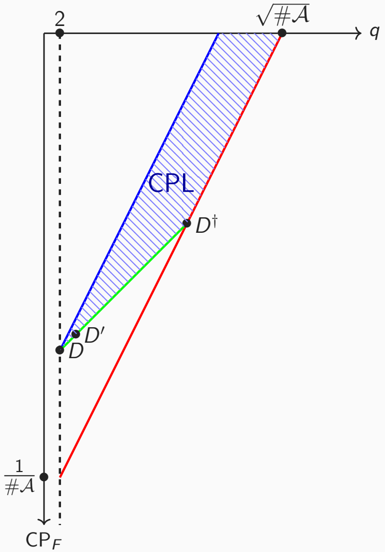
For $D' = D \cup (D + \mu)$ with μ random

$$\mathrm{E}(\mathsf{CP}(D')) \geq 1 - (1 - \mathsf{CP}(D))^2 \approx 2\mathsf{CP}(D)$$

In general for $D' = \bigcup_{1 \leq i \leq n} D + \mu_i$

$$\mathrm{E}(\mathsf{CP}(D')) \approx n\mathsf{CP}(D)$$

... up to the birthday bound ($D^\dagger$)



Choosing the block function

Take 1: pseudorandom permutation

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key
  - distinguishing advantage by adversary limited to forward queries: PRP

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key
  - distinguishing advantage by adversary limited to forward queries: PRP
  - claimed advantage $\epsilon_p(N, M)$ gives:

$$\mathrm{CPL}(q) \leq \binom{q}{2} 2^{-n} + \epsilon_p(N, q)$$

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key
  - distinguishing advantage by adversary limited to forward queries: PRP
  - claimed advantage $\epsilon_p(N, M)$ gives:

$$\mathrm{CPL}(q) \leq \binom{q}{2} 2^{-n} + \epsilon_p(N, q)$$

- If $\epsilon_p(N, q)$ is small, we find ourselves on the birthday bound

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key
  - distinguishing advantage by adversary limited to forward queries: PRP
  - claimed advantage $\epsilon_p(N, M)$ gives:

$$\mathrm{CPL}(q) \leq \binom{q}{2} 2^{-n} + \epsilon_p(N, q)$$

- If $\epsilon_p(N, q)$ is small, we find ourselves on the birthday bound
- This is common practice in MAC functions
  - spot the serial construction in CBC-MAC [ANSI X9.9 1986]
  - spot the parallel construction in PMAC [Black, Rogaway 2001]

Take 1: pseudorandom permutation

- Take a n-bit block cipher with a secret key
  - distinguishing advantage by adversary limited to forward queries: PRP
  - claimed advantage $\epsilon_p(N, M)$ gives:

$$\mathrm{CPL}(q) \leq \binom{q}{2} 2^{-n} + \epsilon_p(N, q)$$

- If $\epsilon_p(N, q)$ is small, we find ourselves on the birthday bound
- This is common practice in MAC functions
  - spot the serial construction in CBC-MAC [ANSI X9.9 1986]
  - spot the parallel construction in PMAC [Black, Rogaway 2001]
- Did we really come all this way to fall back on block ciphers?

Take 2: strong block function

Take 2: strong block function

Take n-bit f that satisfies $\max_{a,b} \mathrm{DP}(a, b) = 2^{x-n}$ for some small x

Take 2: strong block function

Take n-bit f that satisfies $\max_{a,b} \mathrm{DP}(a, b) = 2^{x-n}$ for some small x

Example

- 4-round unkeyed AES
- as used in Pelican-MAC [Daemen, Rijmen 2005]

Take 2: strong block function

Take n-bit f that satisfies $\max_{a,b} \mathrm{DP}(a, b) = 2^{x-n}$ for some small x

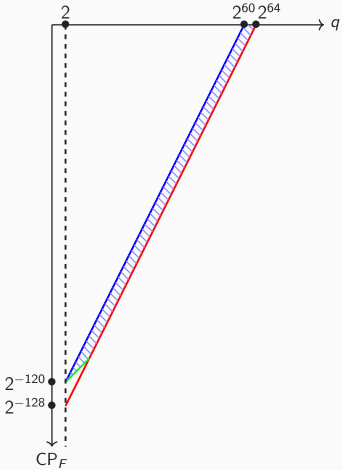
Example

- 4-round unkeyed AES
- as used in Pelican-MAC [Daemen, Rijmen 2005]
- in serial construction

Take n-bit f that satisfies $\max_{a,b} \mathrm{DP}(a, b) = 2^{x-n}$ for some small x

Example

- 4-round unkeyed AES
- as used in Pelican-MAC [Daemen, Rijmen 2005]
- in serial construction
  - plausibly $\max_{a,b} \mathrm{DP}(a, b) < 2^{-120}$
  - security almost as good as full AES

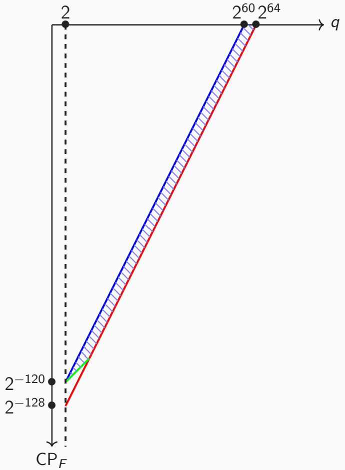


Take 2: strong block function

Take n-bit f that satisfies $\max_{a,b} \mathrm{DP}(a, b) = 2^{x-n}$ for some small x

Example

- 4-round unkeyed AES
- as used in Pelican-MAC [Daemen, Rijmen 2005]
- in serial construction
    - plausibly $\max_{a,b} \mathrm{DP}(a, b) < 2^{-120}$
    - security almost as good as full AES
    - 2.5 times faster than AES in CBC MAC



Take 3: wide permutation

Take 3: wide permutation

- Three steps:

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathrm{CPL}(q)/q \leq 2^{-s}$ for all q

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathrm{CPL}(q)/q \leq 2^{-s}$ for all q
- Conservative approach: take number of rounds such that $\mathrm{CPL}(2) \geq 2^{-2s}$

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathrm{CPL}(q)/q \leq 2^{-s}$ for all q
- Conservative approach: take number of rounds such that $\mathrm{CPL}(2) \geq 2^{-2s}$
- More refined approach

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathrm{CPL}(q)/q \leq 2^{-s}$ for all q
- Conservative approach: take number of rounds such that $\mathrm{CPL}(2) \geq 2^{-2s}$
- More refined approach
  - considers ability to build collision-dense message sets

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathsf{CPL}(q)/q \leq 2^{-s}$ for all q
- Conservative approach: take number of rounds such that $\mathsf{CPL}(2) \geq 2^{-2s}$
- More refined approach
  - considers ability to build collision-dense message sets
  - considers overlap between solution sets within such message sets

Take 3: wide permutation

- Three steps:
  1. Define a target security strength s
  2. Choose for f a permutation with width $\gg 2s$ (typical widths: $384, 512, 1600$)
  3. Take enough rounds in f so that $\mathrm{CPL}(q)/q \leq 2^{-s}$ for all q
- Conservative approach: take number of rounds such that $\mathrm{CPL}(2) \geq 2^{-2s}$
- More refined approach
  - considers ability to build collision-dense message sets
  - considers overlap between solution sets within such message sets
  - requires deep understanding of difference propagation properties of f

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round XOODOO

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round Xoodoo
- Serial construction
  - $\mathrm{CPL}(2) = \max_{a,b} \mathrm{DP}(a, b) = 2^{-36}$
  - collision-dense input sets up to size $q = 2^{18}$
  - security strength $s = 18$

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round Xoodoo
- Serial construction
  - $\mathrm{CPL}(2) = \max_{a,b} \mathrm{DP}(a, b) = 2^{-36}$
  - collision-dense input sets up to size $q = 2^{18}$
  - security strength $s = 18$
- Parallel construction
  - $\mathrm{CPL}(2) = \max_{a} \sum_{b} (\mathrm{DP}(a, b))^2 = 2^{-44}$
  - collision-dense input sets up to size $q = 2^{8}$
  - security strength $s = 36$

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round Xoodoo
- Serial construction
  - $\mathrm{CPL}(2) = \max_{a,b} \mathrm{DP}(a,b) = 2^{-36}$
  - collision-dense input sets up to size $q = 2^{18}$
  - security strength $s = 18$
- Parallel construction
  - $\mathrm{CPL}(2) = \max_a \sum_b (\mathrm{DP}(a,b))^2 = 2^{-44}$
  - collision-dense input sets up to size $q = 2^{8}$
  - security strength $s = 36$
- Observations:

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round XOODOO
- Serial construction
  - $\mathrm{CPL}(2) = \max_{a,b} \mathrm{DP}(a,b) = 2^{-36}$
  - collision-dense input sets up to size $q = 2^{18}$
  - security strength $s = 18$
- Parallel construction
  - $\mathrm{CPL}(2) = \max_{a} \sum_{b} (\mathrm{DP}(a,b))^2 = 2^{-44}$
  - collision-dense input sets up to size $q = 2^{8}$
  - security strength $s = 36$
- Observations:
  - parallel construction twice as secure as serial construction

Example of wide permutation: Xoodoo with width 384 [KT, ToSC 2018]

- We experimented with toy example 3-round Xoodoo
- Serial construction
  - $\mathrm{CPL}(2) = \max_{a,b} \mathrm{DP}(a,b) = 2^{-36}$
  - collision-dense input sets up to size $q = 2^{18}$
  - security strength $s = 18$
- Parallel construction
  - $\mathrm{CPL}(2) = \max_{a} \sum_b (\mathrm{DP}(a,b))^2 = 2^{-44}$
  - collision-dense input sets up to size $q = 2^{8}$
  - security strength $s = 36$
- Observations:
  - parallel construction twice as secure as serial construction
  - $\mathrm{CPL}(q)$ has quadratic segment followed by linear segment

Example of wide permutation: Xoodoo

:

Example of wide permutation: Xoodoo

For 6-round XOODOO:

Example of wide permutation: Xoodoo

For 6-round XOODOO:

- current trail bounds imply:

Example of wide permutation: Xoodoo

For 6-round Xoodoo:

- current trail bounds imply:
  - serial: $\mathrm{CPL}(2) \leq 2^{-104}$ and $s \geq 84$

Example of wide permutation: Xoodoo

For 6-round Xoodoo:

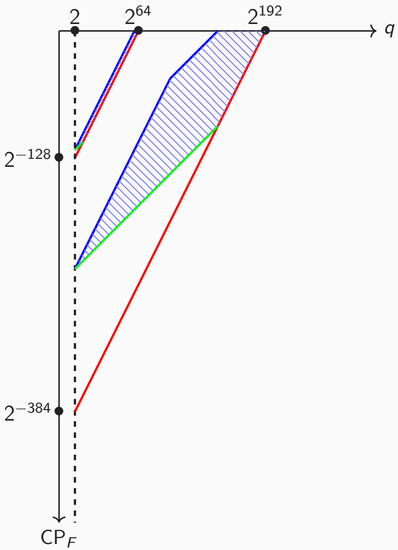
- current trail bounds imply:
    - serial: $\mathsf{CPL}(2) \leq 2^{-104}$ and $s \geq 84$
    - parallel: $\mathsf{CPL}(2) \leq 2^{-198}$ and $s \geq 168$

Example of wide permutation: Xoodoo

For 6-round XOODOO:

- current trail bounds imply:
  - serial: $\mathrm{CPL}(2) \leq 2^{-104}$ and $s \geq 84$
  - parallel: $\mathrm{CPL}(2) \leq 2^{-198}$ and $s \geq 168$

  assuming ...
  - independent keys
  - no massive trail clustering in differentials



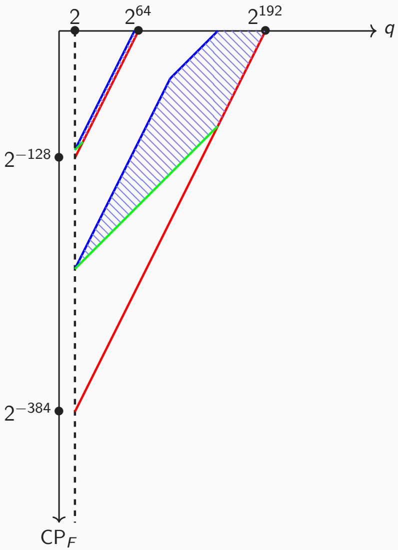
Example of wide permutation: Xoodoo

For 6-round Xoodoo:

- current trail bounds imply:
  - serial: $\mathrm{CPL}(2) \leq 2^{-104}$ and $s \geq 84$
  - parallel: $\mathrm{CPL}(2) \leq 2^{-198}$ and $s \geq 168$

  assuming . . .
  - independent keys
  - no massive trail clustering in differentials

is it fair to compare 6R Xoodoo with 4R AES?



Example of wide permutation: Xoodoo

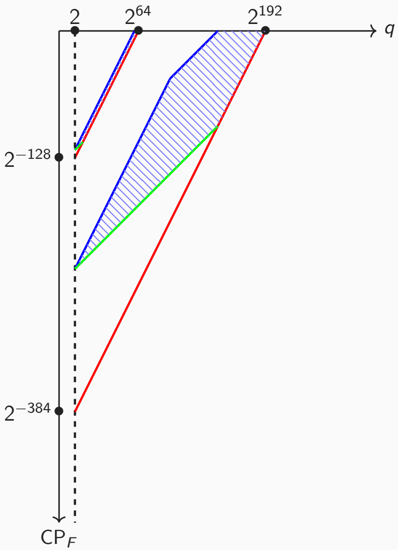
For 6-round XOODOO:

- current trail bounds imply:
  - serial: $\mathsf{CPL}(2) \leq 2^{-104}$ and $s \geq 84$
  - parallel: $\mathsf{CPL}(2) \leq 2^{-198}$ and $s \geq 168$

  assuming ...
  - independent keys
  - no massive trail clustering in differentials

is it fair to compare 6R XOODOO with 4R AES?

- 4R AES takes about 3 times more operations per bit than 6R XOODOO



Conclusion

Keyed hashing with wide permutations

- can be very competitive
- parallel construction outperforms serial construction

Conclusion

Keyed hashing with wide permutations

- can be very competitive
- parallel construction outperforms serial construction

Future work

- further investigate trails in wide permutations
- add key expansion

Conclusion

Keyed hashing with wide permutations

- can be very competitive
- parallel construction outperforms serial construction

Future work

- further investigate trails in wide permutations
- add key expansion

Thank you for your attention!