Optimal PRFs from Blockcipher Designs

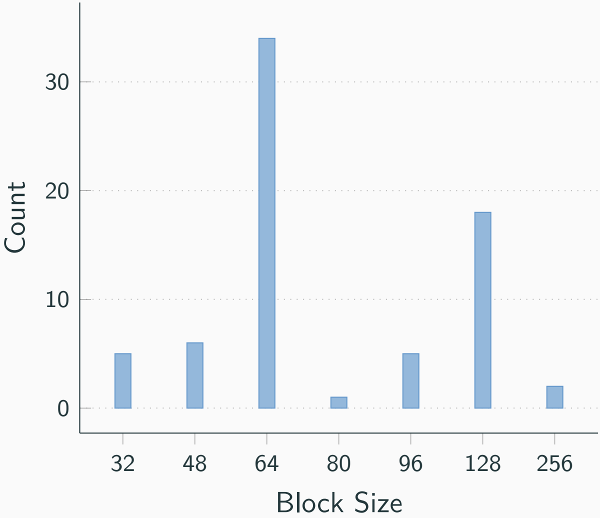
Bart Mennink [1]   Samuel Neves [2]

FSE 2018

[1]Radboud University

[2]University of Coimbra

Lightweight Cipher Block Sizes



Birthday Attacks

- "On the Practical (In-)Security of 64-bit Block Ciphers – Collision Attacks on HTTP over TLS and OpenVPN"
- "Impossible plaintext cryptanalysis and probable-plaintext collision attacks of 64-bit block cipher modes"
- "The Missing Difference Problem, and its Applications to Counter Mode Encryption"
- "Optimal Forgeries Against Polynomial-Based MACs and GCM"

Invertibility as a Liability

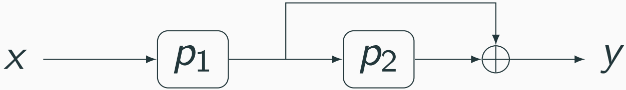
- AES-GCM, AES-CCM, ...
    - Needs a PRF, not a PRP
    - PRP in fact the greatest contributor to security degradation
- Why don't we design PRFs instead?
    - We actually do, but they're usually {truncated, xored, ...} from idealized permutations
    - Permutations are what we know how to build
    - Losing information, but not too much, is tricky
    - Non-invertible round functions lose too much

Invertibility as a Liability

- AES-GCM, AES-CCM, ...
  - Needs a PRF, not a PRP
  - PRP in fact the greatest contributor to security degradation
- Why don't we design PRFs instead?
  - We actually do, but they're usually {truncated, xored, ...} from idealized permutations
  - Permutations are what we know how to build
  - Losing information, but not too much, is tricky
  - Non-invertible round functions lose too much
- *Can we design PRFs without performance or security hit?*

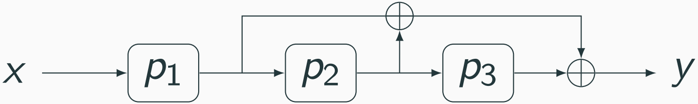
GEDMD

Generalized EDMD



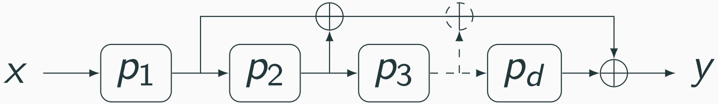
- $\mathbf{Adv}^{\mathrm{PRF}}_{EDMD^{p_1,p_2}}(\mathcal{D}) \leq q/2^n$ (CRYPTO 2017)
- Simple reduction to xor of permutations, extensively studied

Generalized EDMD



- $\mathbf{Adv}^{\mathrm{PRF}}_{EDMD^{p_1,p_2}}(\mathcal{D}) \leq q/2^n$ (CRYPTO 2017)
- Simple reduction to xor of permutations, extensively studied
- No reason to limit ourselves to 2 permutations

Generalized EDMD



- $\mathbf{Adv}^{\mathrm{PRF}}_{EDMD^{p_1,p_2}}(\mathcal{D}) \leq q/2^n$ (CRYPTO 2017)
- Simple reduction to xor of permutations, extensively studied
- No reason to limit ourselves to 2 permutations
- Generalization also reduces to EDMD *or* xor of d permutations

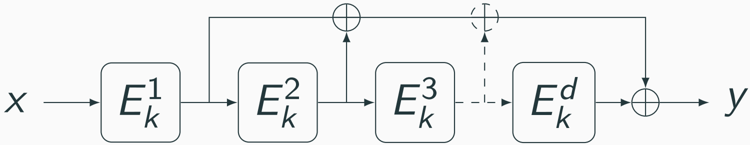
FastPRF

Design Principle

$$x \longrightarrow \boxed{E_k^1} \longrightarrow \boxed{E_k^2} \longrightarrow \boxed{E_k^3} \dashrightarrow \boxed{E_k^d} \longrightarrow y$$

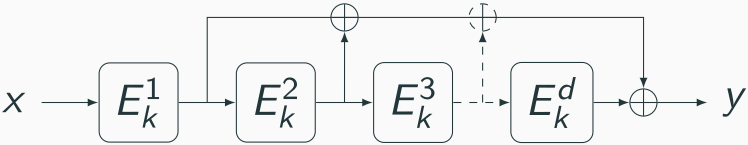
- Treat block cipher E_k as composition of permutations

Design Principle



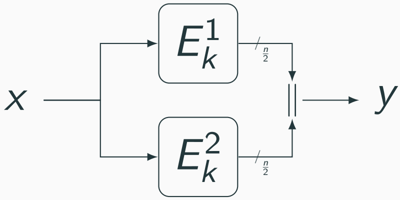
- Treat block cipher E_k as composition of permutations
- Apply GEDMD using imperfect permutations $E_k^1, E_k^2, \ldots$
- "Prove-then-prune"

Design Principle



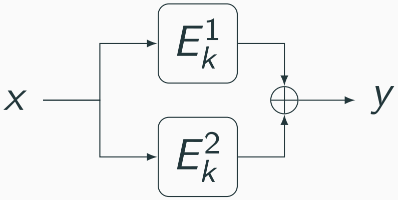
- Treat block cipher E_k as composition of permutations
- Apply GEDMD using imperfect permutations $E_k^1, E_k^2, \ldots$
- "Prove-then-prune"
- *Why GEDMD?*

Truncated Permutations



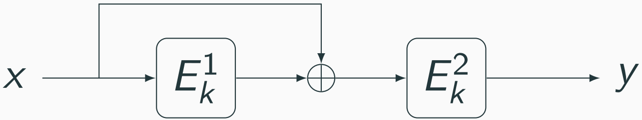
- At best $2^{3n/4}$ security
- Attacker gets direct access to weaker E_k^1 and E_k^2
- Risky

Sum of Permutations



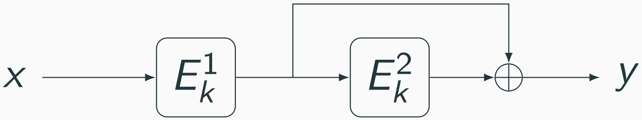
- Interesting properties may get through $E_k^1 \oplus E_k^2$
- E.g., linear/differential/integral characteristics
- Still risky

EDM (Cogliati-Seurin)



- Attacker has some control over input of E_k^2
- Differential collisions if E_k^1 has high-probability differential
- Does not generalize easily to more permutations

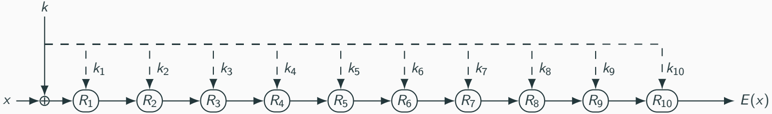
(G)EDMD (Mennink-Neves)



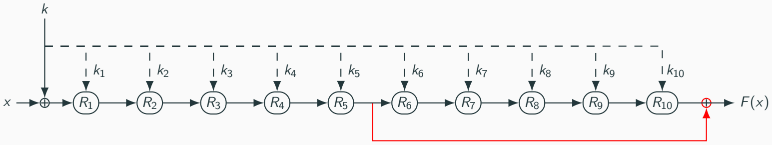
- No direct control over intermediate states
- Output always masked by full application of E_k
- Appears to be the least risky option!

AES-PRF

AES

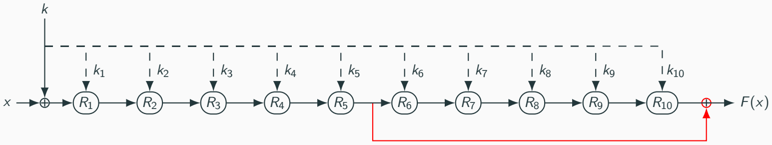


AES-PRF



AES-PRF-128 is AES-128, with a feed-forward after the 5th round

AES-PRF

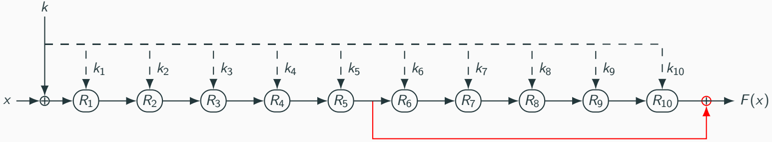


AES-PRF-128 is AES-128, with a feed-forward after the 5th round

AES-PRF-192 is AES-192, with a feed-forward after the 6th round

AES-PRF-256 is AES-256, with a feed-forward after the 7th round

AES-PRF



AES-PRF-128 is AES-128, with a feed-forward after the 5th round

AES-PRF-192 is AES-192, with a feed-forward after the 6th round

AES-PRF-256 is AES-256, with a feed-forward after the 7th round

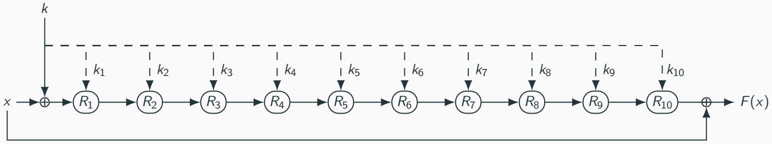
Not the only reasonable choices!

AES-PRF Security

- $\{5, 6, 7\}$-round AES, i.e., $E_k^1(\cdot)$, is weakest component
- But is masked by full AES
- Existing $\{4, 5\}$-round distinguishers do not work in this setting
- Differential and linear distinguishers are ineffective

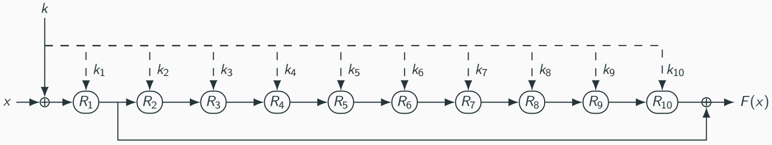
- Try to break unbalanced AES-PRF variants instead
- E.g., $\mathrm{AES}_{10}(x) \oplus x$, $\mathrm{AES}_{10}(x) \oplus \mathrm{AES}_1(x)$, ...

$\mathrm{AES}_{10}(x) \oplus x$



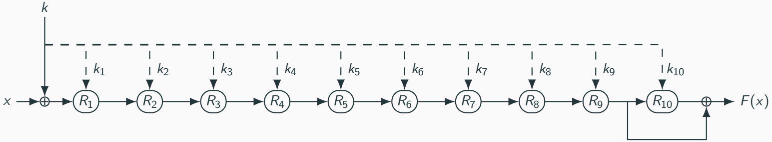
- This is simply Davies-Meyer
- $\mathrm{AES}_{10} = F(x) \oplus x$
- Distinguish in $\approx 2^{64}$ by standard method

$\mathrm{AES}_{10}(x) \oplus \mathrm{AES}_1(x)$



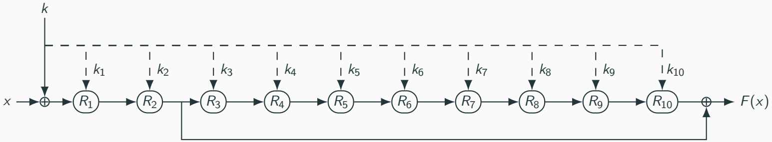
- Cancel out contribution of $\mathrm{AES}_1(x)$, 32 bits at a time
- Candidate keys with no collisions happen are likely correct
- Key recovery in $\approx 2^{67}$ queries and memory, 2^{101} time

$\mathrm{AES}_{10}(x) \oplus \mathrm{AES}_9(x)$



- No final `MixColumns`
- Output is of the form $S(x) \oplus x$
- Highly biased

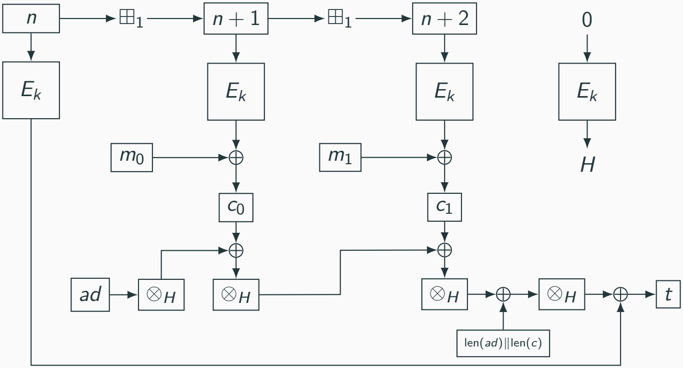
$\mathrm{AES}_{10}(x) \oplus \mathrm{AES}_2(x)$



- Canceling out $\mathrm{AES}_2(x)$ too expensive
- New strategy required
- Seems likely to be breakable as well

Applications of AES-PRF

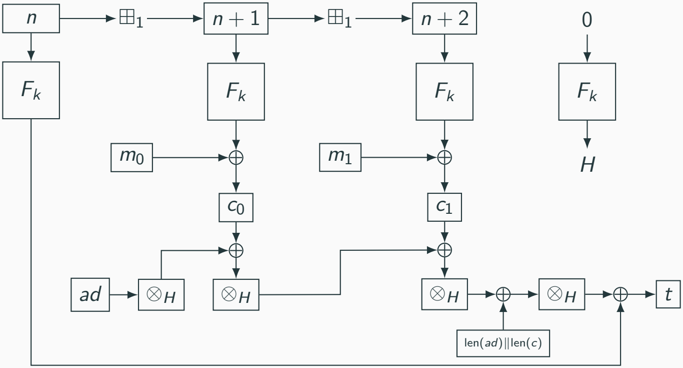
AES-GCM Before AES-PRF



$$\mathbf{Adv}^{\mathrm{conf}}_{\mathrm{GCM[AES},\tau]}(\mathcal{D}) \leq \mathbf{Adv}^{\mathrm{prp}}_{\mathrm{AES}}(\mathcal{D}') + \binom{q+\sigma+1}{2}/2^n$$

$$\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{GCM[AES},\tau]}(\mathcal{D}) \leq \mathbf{Adv}^{\mathrm{prp}}_{\mathrm{AES}}(\mathcal{D}') + \frac{q'(\ell+1)}{2^\tau} + \binom{q+q'+\sigma+1}{2}/2^n$$

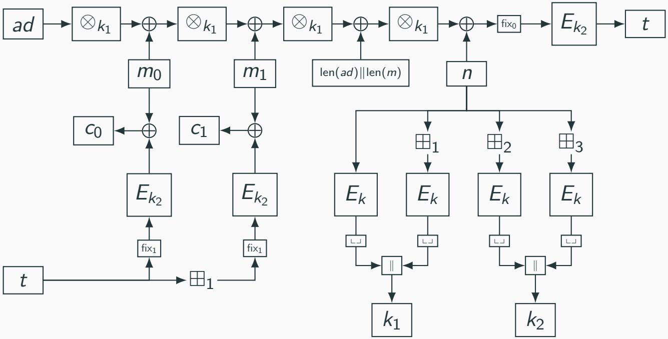
AES-GCM After AES-PRF



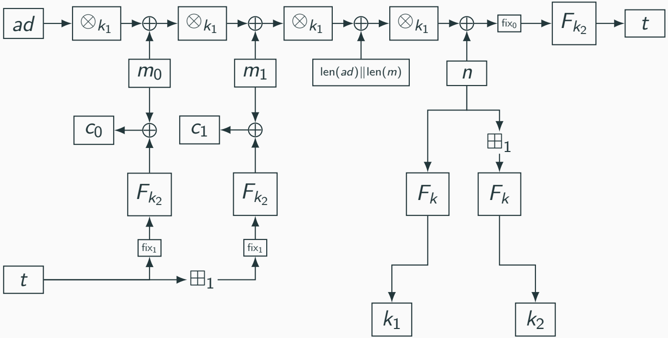
$$\mathbf{Adv}^{\mathrm{conf}}_{\mathrm{GCM[AES\text{-}PRF},\tau]}(\mathcal{D}) \leq \mathbf{Adv}^{\mathrm{prf}}_{\mathrm{AES\text{-}PRF}}(\mathcal{D}')$$

$$\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{GCM[AES\text{-}PRF},\tau]}(\mathcal{D}) \leq \mathbf{Adv}^{\mathrm{prf}}_{\mathrm{AES\text{-}PRF}}(\mathcal{D}') + \frac{q'(\ell+1)}{2^\tau}$$

AES-GCM-SIV Before AES-PRF



AES-GCM-SIV After AES-PRF



- Improved, natural, key derivation
- 2–3 fewer PRF calls
- Like GCM, birthday terms disappear

Tweakable FastPRF

Tweakable FastPRF

- FastPRF principle also applicable to tweakable blockciphers
- Draw from successful designs
  - SKINNY, MANTIS, QARMA, ...
  - E.g., SKINNY-128-256 with feed-forward after 24 rounds
- Result: compressing $\{0,1\}^{256} \to \{0,1\}^{128}$ PRF
- Simple, length-independent authenticators
- E.g., Protected counter sums
- Or PMAC1 bounded by $\mathbf{Adv}^{\mathrm{prf}}_{\widetilde{\mathrm{FastPRF}}}(\mathcal{D}') + \binom{q}{2}/2^n$
  instead of by $\mathbf{Adv}^{\mathrm{tprp}}_{\widetilde{E}}(\mathcal{D}') + \binom{q}{2}/2^n + \binom{\sigma}{2}/2^n$

Future Work

Variants

- Single-permutation (G)EDMD
  - $p(p(x)) \oplus p(x)$
  - Conjectured to be optimally secure
  - FastPRF analogous would cut key schedule cost in (at least) half
  - How secure is it?
- Public-permutation (G)EDMD
  - For usage in, e.g., sponge designs
  - “Free” forward security
  - How secure is it?

Suggestions

- Designers
  - Consider including a PRF along with your new lightweight cipher
  - Might be useful to distinguish between PRP and PRF calls
  - E.g., different constants

Suggestions

- Designers
  - Consider including a PRF along with your new lightweight cipher
  - Might be useful to distinguish between PRP and PRF calls
  - E.g., different constants
- Cryptanalysts
  - Look at AES-PRF!
  - ...or its reduced/unbalanced versions

Suggestions

- Designers
  - Consider including a PRF along with your new lightweight cipher
  - Might be useful to distinguish between PRP and PRF calls
  - E.g., different constants
- Cryptanalysts
  - Look at AES-PRF!
  - ...or its reduced/unbalanced versions
- Theorists
  - Minimal assumptions for GEDMD / FastPRF to be secure?
  - Efficient tweakable-PRF constructions from non-tweakable PRP designs?

Thank you!