New Constructions of MACs from (Tweakable) Block Ciphers

Benoît Cogliati[1] Jooyoung Lee[2] Yannick Seurin[3]

[1]UL, Luxembourg

[2]KAIST, Korea

[3]ANSSI, France

March 6, 2018 — FSE 2018

Summary of the contribution

- we propose four new MAC constructions based on a (tweakable) block cipher:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all four constructions are secure beyond the birthday bound
- TBC-based constructions are provably secure in the standard model
- BC-based constructions are provably secure in the ideal cipher model
- nonce-based constructions provide graceful security degradation with the maximal number of nonce repetitions

Summary of the contribution

- we propose four new MAC constructions based on a (tweakable) block cipher:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all four constructions are secure beyond the birthday bound
- TBC-based constructions are provably secure in the standard model
- BC-based constructions are provably secure in the ideal cipher model
- nonce-based constructions provide graceful security degradation with the maximal number of nonce repetitions

Summary of the contribution

- we propose four new MAC constructions based on a (tweakable) block cipher:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all four constructions are secure beyond the birthday bound
- TBC-based constructions are provably secure in the standard model
- BC-based constructions are provably secure in the ideal cipher model
- nonce-based constructions provide graceful security degradation with the maximal number of nonce repetitions

Summary of the contribution

- we propose four new MAC constructions based on a (tweakable) block cipher:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

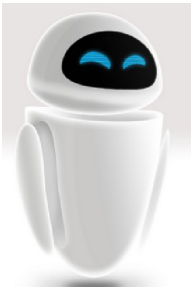
- all four constructions are secure beyond the birthday bound
- TBC-based constructions are provably secure in the standard model
- BC-based constructions are provably secure in the ideal cipher model
- nonce-based constructions provide graceful security degradation with the maximal number of nonce repetitions

Outline

Outline

MAC definition

$T = \mathrm{MAC}_K(N, M)$



$\mathrm{MAC}_K(N', M') = T'$?

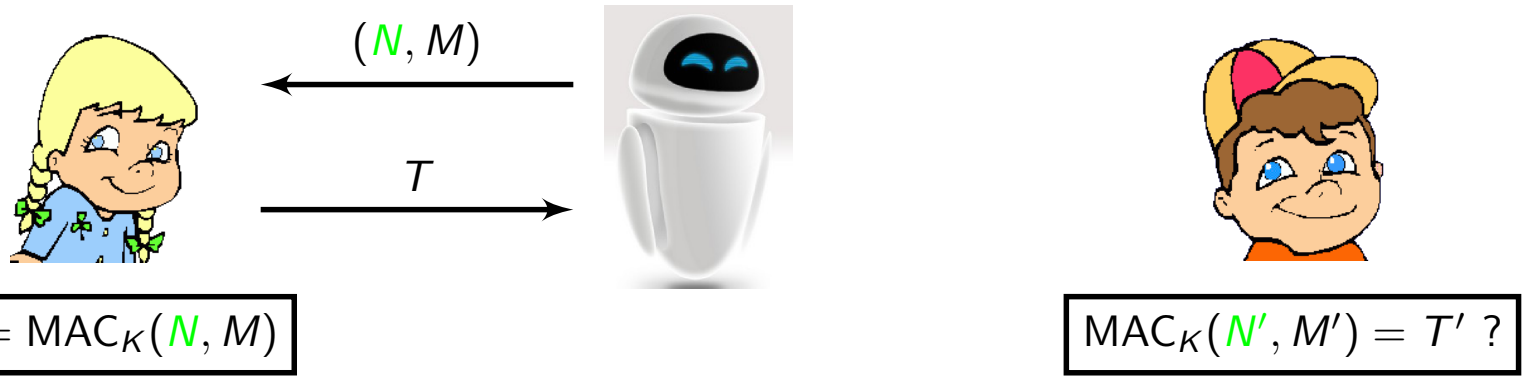
Security Definition

The adversary is allowed

- q MAC queries $T = \mathrm{MAC}_K(N, M)$
- v verification queries (forgery attempts) (N', M', T')

and is successful if one of the verification queries (N', M', T') passes and no previous MAC query (N', M') returned T'.

MAC definition



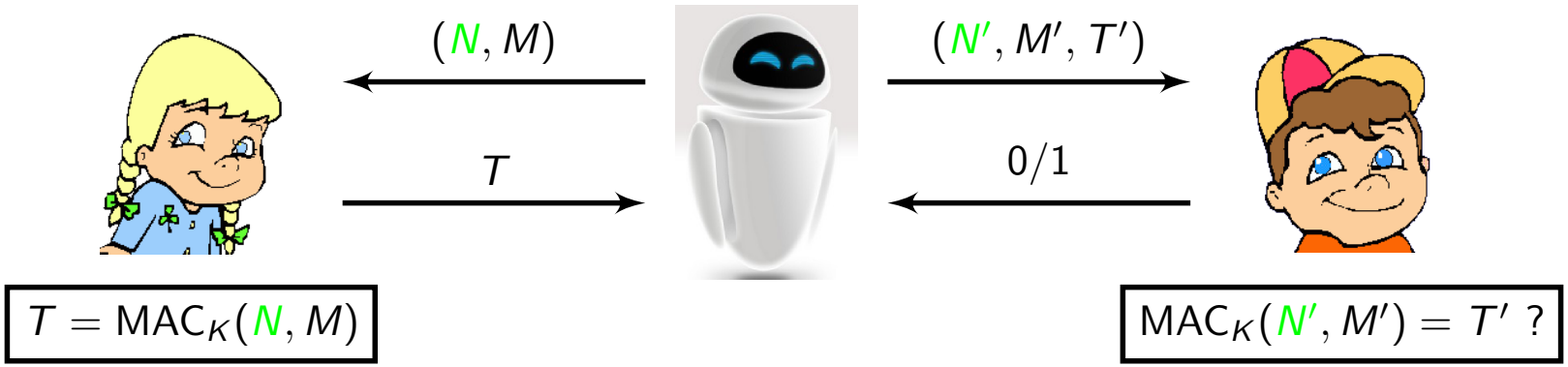
Security Definition

The adversary is allowed

- q MAC queries $T = \mathrm{MAC}_K(N, M)$
- v verification queries (forgery attempts) (N', M', T')

and is successful if one of the verification queries (N', M', T') passes and no previous MAC query (N', M') returned T'.

MAC definition



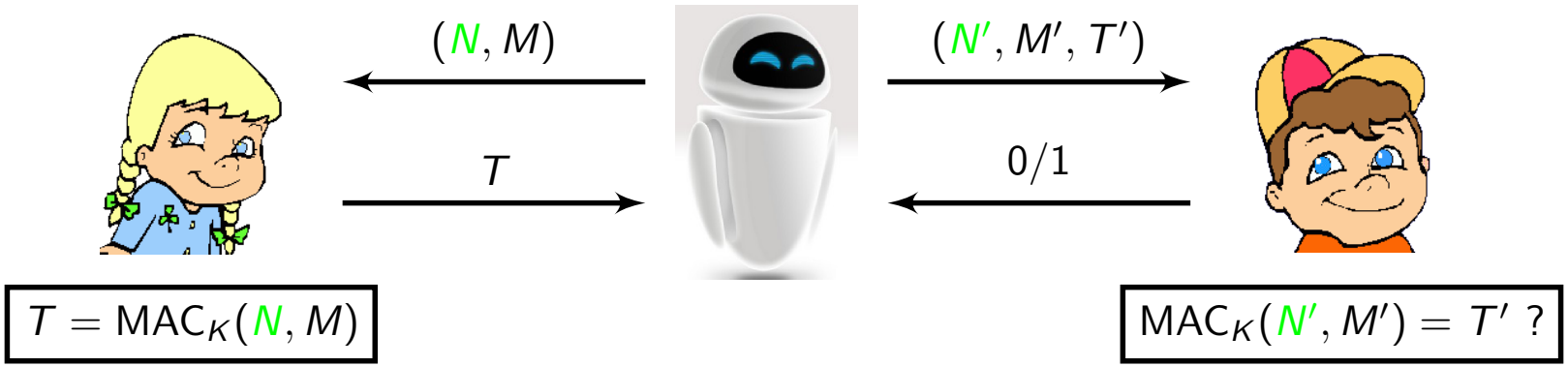
Security Definition

The adversary is allowed

- q MAC queries $T = \mathrm{MAC}_K(N, M)$
- v verification queries (forgery attempts) (N', M', T')

and is successful if one of the verification queries (N', M', T') passes and no previous MAC query (N', M') returned T'.

MAC definition



Security Definition

The adversary is allowed

- q MAC queries $T = \mathrm{MAC}_K(N, M)$
- v verification queries (forgery attempts) (N', M', T')

and is successful if one of the verification queries (N', M', T') passes and no previous MAC query (N', M') returned T'.

Three types of MAC

- stateless and deterministic: MAC function only takes the key and the message as input
  (Variable-input-length PRF $\Rightarrow$ stateless deterministic MAC)
- nonce-based:
  - MAC function takes as input a non-repeating nonce N in addition to the key and the message M
  - security model: nonces are chosen by the adversary, any nonce can be used at most μ times in MAC queries
  - $\mu = 1$: nonce-respecting adversary
  - $\mu > 1$: nonce-misusing adversary
- randomized: MAC function takes as input random coins (generated by the sender) in addition to the key and the message

Three types of MAC

- stateless and deterministic: MAC function only takes the key and the message as input
  (Variable-input-length PRF $\Rightarrow$ stateless deterministic MAC)
- nonce-based:
  - MAC function takes as input a non-repeating nonce N in addition to the key and the message M
  - security model: nonces are chosen by the adversary, any nonce can be used at most μ times in MAC queries
  - $\mu = 1$: nonce-respecting adversary
  - $\mu > 1$: nonce-misusing adversary
- randomized: MAC function takes as input random coins (generated by the sender) in addition to the key and the message

Three types of MAC

- stateless and deterministic: MAC function only takes the key and the message as input
  (Variable-input-length PRF $\Rightarrow$ stateless deterministic MAC)
- nonce-based:
  - MAC function takes as input a non-repeating nonce N in addition to the key and the message M
  - security model: nonces are chosen by the adversary, any nonce can be used at most μ times in MAC queries
  - $\mu = 1$: nonce-respecting adversary
  - $\mu > 1$: nonce-misusing adversary
- randomized: MAC function takes as input random coins (generated by the sender) in addition to the key and the message

Graceful nonce-misuse security degradation

- the security of some nonce-based MACs collapses if a single nonce is used twice (e.g. GMAC)
- ideally, security should degrade gracefully in case nonces are repeated
- any BBB-secure nonce-based MAC with graceful security degradation can be turned into a BBB-secure randomized MAC by choosing n-bit nonces uniformly at random:

$$\mathbf{Adv}_F^{\text{rand-MAC}}(q,v) \leq \underbrace{\frac{q^{\mu+1}}{2^{\mu(n+1)}}}_{\substack{\mu\text{-multicoll.}\\ \text{proba.}}} + \underbrace{\mathbf{Adv}_F^{\text{nonce-MAC}}(q,v,\mu)}_{\text{small for } \mu>1}$$

for any value of μ = maximal number of nonce repetitions.

Graceful nonce-misuse security degradation

- the security of some nonce-based MACs collapses if a single nonce is used twice (e.g. GMAC)
- ideally, security should degrade gracefully in case nonces are repeated
- any BBB-secure nonce-based MAC with graceful security degradation can be turned into a BBB-secure randomized MAC by choosing n-bit nonces uniformly at random:

$$\mathbf{Adv}_F^{\text{rand-MAC}}(q, v) \leq \underbrace{\frac{q^{\mu+1}}{2^{\mu(n+1)}}}_{\substack{\mu\text{-multicoll.} \\ \text{proba.}}} + \underbrace{\mathbf{Adv}_F^{\text{nonce-MAC}}(q, v, \mu)}_{\text{small for } \mu>1}$$

for any value of μ = maximal number of nonce repetitions.

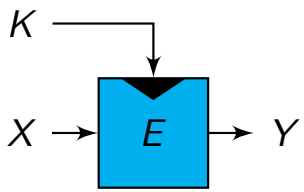
Graceful nonce-misuse security degradation

- the security of some nonce-based MACs collapses if a single nonce is used twice (e.g. GMAC)
- ideally, security should degrade gracefully in case nonces are repeated
- any BBB-secure nonce-based MAC with graceful security degradation can be turned into a BBB-secure randomized MAC by choosing n-bit nonces uniformly at random:

$$\mathbf{Adv}_F^{\text{rand-MAC}}(q,v) \leq \underbrace{\frac{q^{\mu+1}}{2^{\mu(n+1)}}}_{\substack{\mu\text{-multicoll.}\\ \text{proba.}}} + \underbrace{\mathbf{Adv}_F^{\text{nonce-MAC}}(q,v,\mu)}_{\text{small for } \mu>1}$$

for any value of μ = maximal number of nonce repetitions.

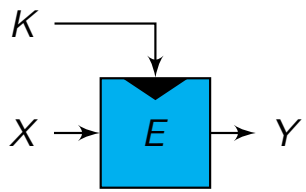
Building blocks: BCs and TBCs

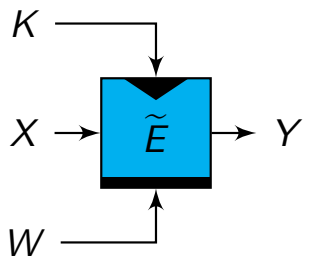


n = block size
t = tweak size

- block cipher E: for each key K, $X \mapsto E(K, X)$ is a permutation
- tweakable block cipher $\widetilde{E}$: for each key K and each tweak W, $X \mapsto \widetilde{E}(K, W, X)$ is a permutation
- one can think of a keyed TBC $\widetilde{E}_K$ as an "imperfect" PRF from $(n + t)$ bits to n bits
- if any tweak W is used at most "a few" times, $\widetilde{E}_K$ is close to a random $(n + t)$-to-n-bit function

Building blocks: BCs and TBCs

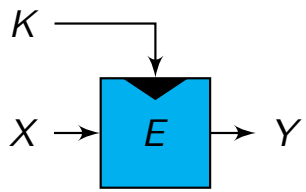


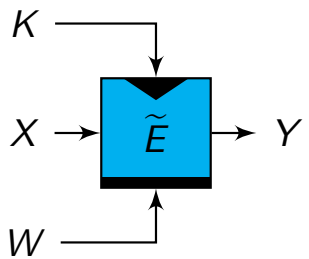


n = block size
t = tweak size

- block cipher E: for each key K, $X \mapsto E(K, X)$ is a permutation
- tweakable block cipher $\widetilde{E}$: for each key K and each tweak W, $X \mapsto \widetilde{E}(K, W, X)$ is a permutation
- one can think of a keyed TBC $\widetilde{E}_K$ as an "imperfect" PRF from $(n + t)$ bits to n bits
- if any tweak W is used at most "a few" times, $\widetilde{E}_K$ is close to a random $(n + t)$-to-n-bit function

Building blocks: BCs and TBCs

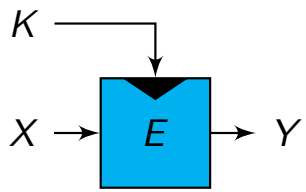


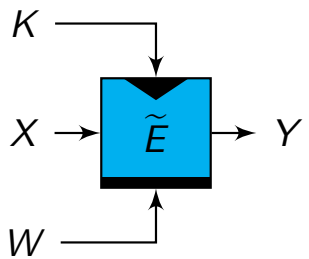


n = block size
t = tweak size

- block cipher E: for each key K, $X \mapsto E(K, X)$ is a permutation
- tweakable block cipher $\widetilde{E}$: for each key K and each tweak W, $X \mapsto \widetilde{E}(K, W, X)$ is a permutation
- one can think of a keyed TBC $\widetilde{E}_K$ as an "imperfect" PRF from $(n + t)$ bits to n bits
- if any tweak W is used at most "a few" times, $\widetilde{E}_K$ is close to a random $(n + t)$-to-n-bit function

Building blocks: BCs and TBCs



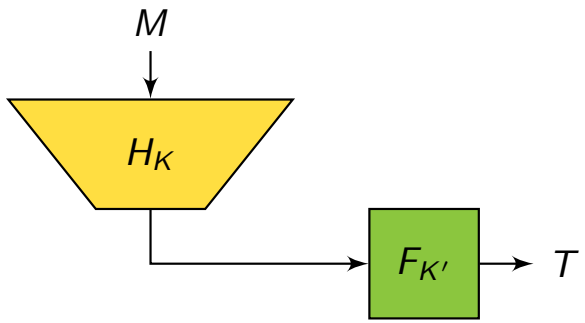


n = block size
t = tweak size

- block cipher E: for each key K, $X \mapsto E(K, X)$ is a permutation
- tweakable block cipher $\widetilde{E}$: for each key K and each tweak W, $X \mapsto \widetilde{E}(K, W, X)$ is a permutation
- one can think of a keyed TBC $\widetilde{E}_K$ as an "imperfect" PRF from $(n + t)$ bits to n bits
- if any tweak W is used at most "a few" times, $\widetilde{E}_K$ is close to a random $(n + t)$-to-n-bit function

Outline

The "standard" UHF-then-PRF Construction

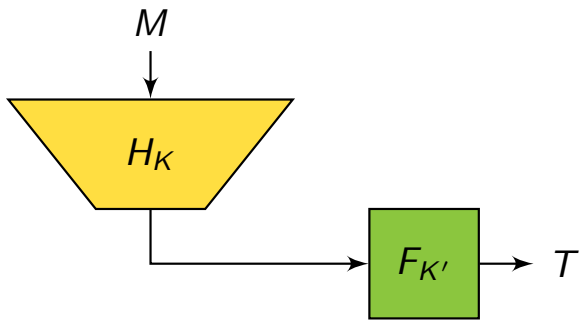


- based on a fixed-input-length PRF F and an ε-almost universal (ε-AU) hash function H:

$$\forall M \neq M', \ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) = H_K(M')] \leq \varepsilon$$

- H can be statistically secure (polynomial evaluation) or computationally secure (BC/TBC-based)
- most MACs are (variants of) this construction (UMAC, EMAC, OMAC, CMAC, PMAC, NMAC)

The "standard" UHF-then-PRF Construction

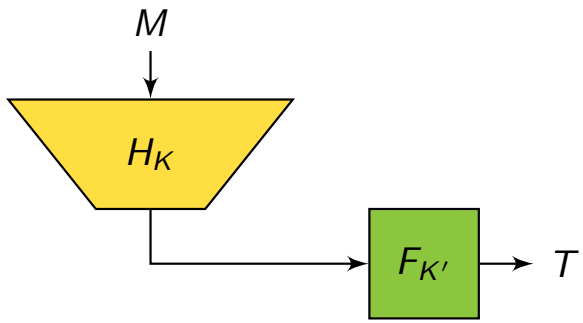


- based on a fixed-input-length PRF F and an ε-almost universal (ε-AU) hash function H:

$$\forall M \neq M', \ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) = H_K(M')] \leq \varepsilon$$

- H can be statistically secure (polynomial evaluation) or computationally secure (BC/TBC-based)
- most MACs are (variants of) this construction (UMAC, EMAC, OMAC, CMAC, PMAC, NMAC)

The "standard" UHF-then-PRF Construction

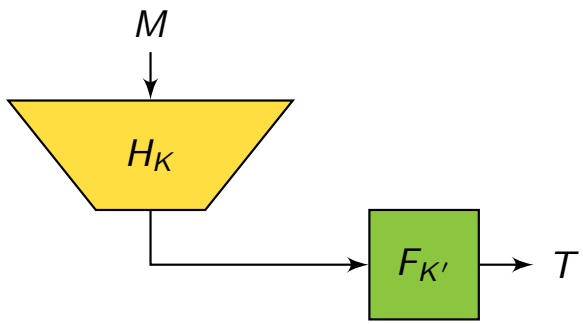


- based on a fixed-input-length PRF F and an ε-almost universal (ε-AU) hash function H:

$$\forall M \neq M', \ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) = H_K(M')] \leq \varepsilon$$

- H can be statistically secure (polynomial evaluation) or computationally secure (BC/TBC-based)
- most MACs are (variants of) this construction (UMAC, EMAC, OMAC, CMAC, PMAC, NMAC)

Security of UHF-then-PRF



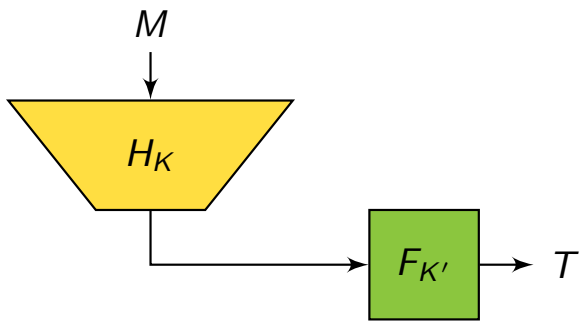
- birthday-bound-secure w.r.t. H collision probability ε

$$\mathbf{Adv}^{\mathrm{PRF}}_{F\circ H}(q) \leq \frac{q^2\varepsilon}{2} + \mathbf{Adv}^{\mathrm{PRF}}_{F}(q)$$

- typical instantiation from a block cipher E:
  - $H \leftarrow$ CBC-MAC[E] or PMAC[E] ($\varepsilon \simeq 2^{-n}$)
  - $F \leftarrow E$

  $\Rightarrow$ BB-security

Security of UHF-then-PRF



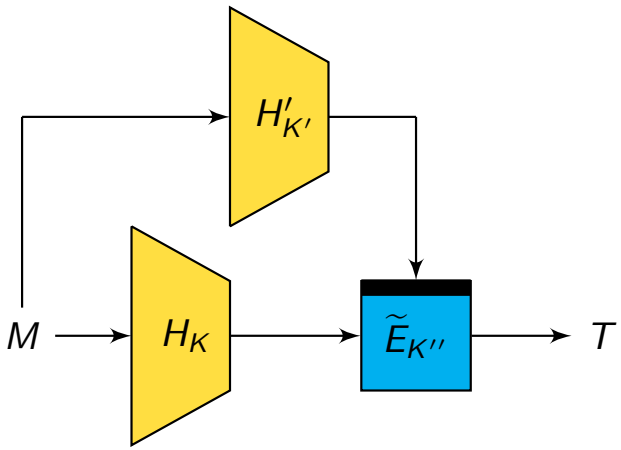
- birthday-bound-secure w.r.t. H collision probability ε

$$\mathbf{Adv}^{\mathrm{PRF}}_{F\circ H}(q) \leq \frac{q^2\varepsilon}{2} + \mathbf{Adv}^{\mathrm{PRF}}_{F}(q)$$

- typical instantiation from a block cipher E:
  - $H \leftarrow$ CBC-MAC[E] or PMAC[E] ($\varepsilon \simeq 2^{-n}$)
  - $F \leftarrow E$

  $\Rightarrow$ BB-security

Construction 1: Hash-as-Tweak (HaT)



Hash-as-Tweak (HaT)

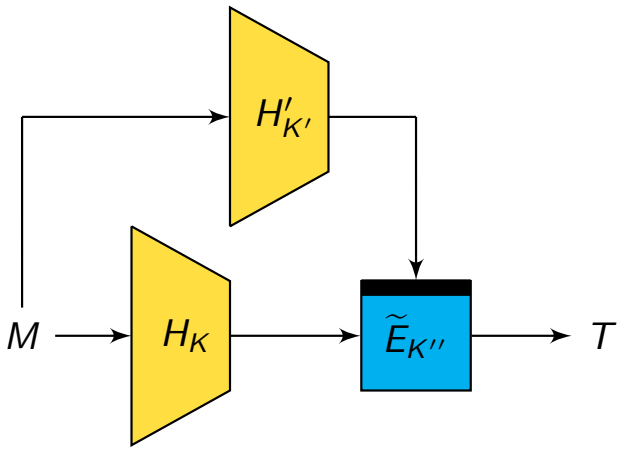
Hash-then-TBC

- BBB-secure assuming H and H' are ε-AU secure:

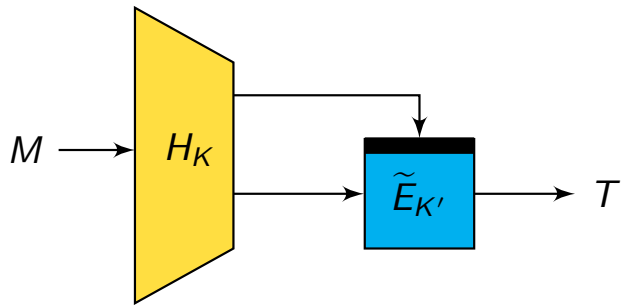
$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathrm{HaT}}(q, v) \leq q^2\varepsilon^2 + qv\varepsilon^2 + (\ldots)$$

- follow-up work: Hash-then-TBC construction [LN17], BBB-secure under more complex UHF-type properties of H

Construction 1: Hash-as-Tweak (HaT)



Hash-as-Tweak (HaT)



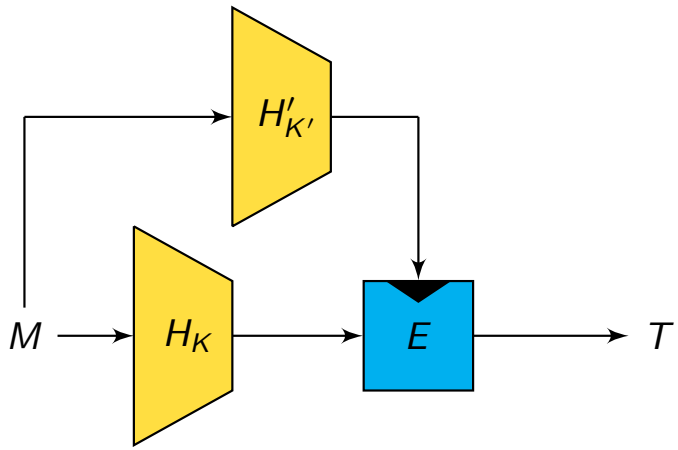
Hash-then-TBC

- BBB-secure assuming H and H' are ε-AU secure:

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathrm{HaT}}(q, v) \leq q^2\varepsilon^2 + qv\varepsilon^2 + (\ldots)$$

- follow-up work: Hash-then-TBC construction [LN17], BBB-secure under more complex UHF-type properties of H

Construction 2: Hash-as-Key (HaK)



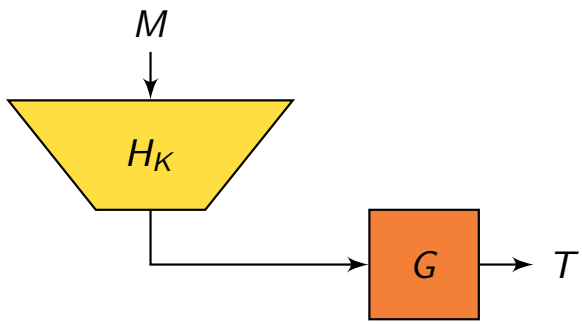
- output transformation unkeyed $\Rightarrow$ H and H' must be ε'-uniform:

$$\forall M,\ \forall Y,\ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) = Y] \leq \varepsilon'$$

- BBB-secure in the ideal cipher model assuming H and H' are ε-AU and ε'-uniform:

$$\mathbf{Adv}^{\mathrm{MAC}}_{\mathrm{HaK}}(q, v) \leq q^2\varepsilon^2 + qv\varepsilon^2 + (\ldots)$$

The UHF-then-RO construction

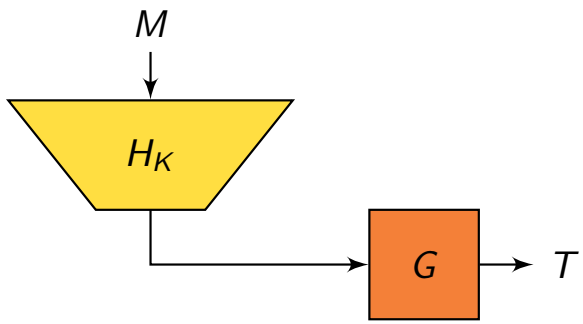


- Hash-as-Key (HaK) is a special case of the "UHF-then-RO" construction
- modeling G as a random function oracle (q_G queries), the construction is secure if H is ε-AU and ε'-uniform:

$$\mathbf{Adv}^{\mathrm{PRF}}_{G \circ H}(q, q_G) \leq \frac{q^2 \varepsilon}{2} + q q_G \varepsilon'$$

- security proof under a standard assumption on G?

The UHF-then-RO construction

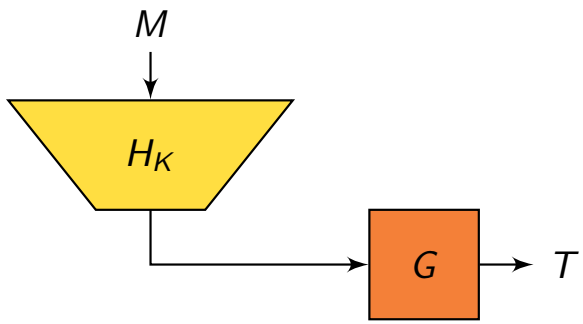


- Hash-as-Key (HaK) is a special case of the "UHF-then-RO" construction
- modeling G as a random function oracle (q_G queries), the construction is secure if H is ε-AU and ε'-uniform:

$$\mathbf{Adv}^{\mathrm{PRF}}_{G \circ H}(q, q_G) \leq \frac{q^2 \varepsilon}{2} + q q_G \varepsilon'$$

- security proof under a standard assumption on G?

The UHF-then-RO construction



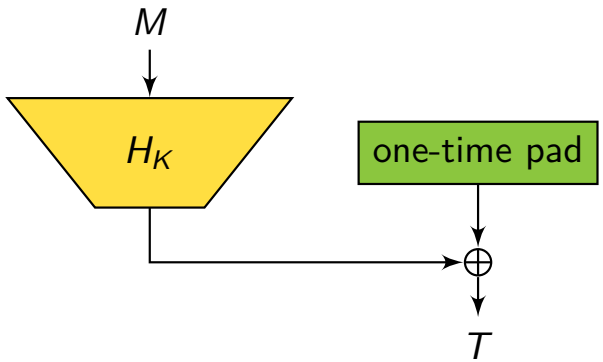
- Hash-as-Key (HaK) is a special case of the "UHF-then-RO" construction
- modeling G as a random function oracle (q_G queries), the construction is secure if H is ε-AU and ε'-uniform:

$$\mathbf{Adv}^{\mathrm{PRF}}_{G \circ H}(q, q_G) \leq \frac{q^2 \varepsilon}{2} + q q_G \varepsilon'$$

- security proof under a standard assumption on G?

Outline

The Wegman-Carter construction [GMS74, WC81]



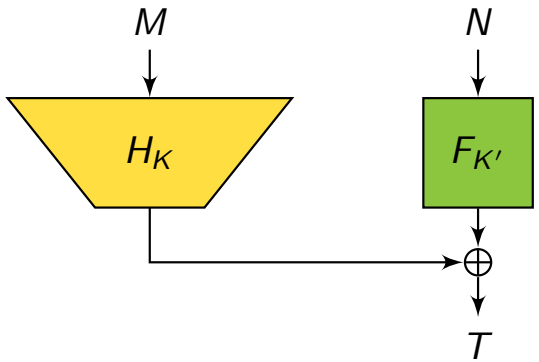
- based on an ε-almost xor-universal (ε-AXU) hash function H:

$$\forall M \neq M',\ \forall Y,\ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) \oplus H_K(M') = Y] \leq \varepsilon$$

- in practice, OTPs are replaced by a PRF applied to a nonce N
- H usually based on polynomial evaluation (GHASH, Poly1305)
- "optimal" security:

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WC}}(q, v) \leq v\varepsilon + \mathbf{Adv}^{\text{PRF}}_{F}(q + v)$$

The Wegman-Carter construction [GMS74, WC81]



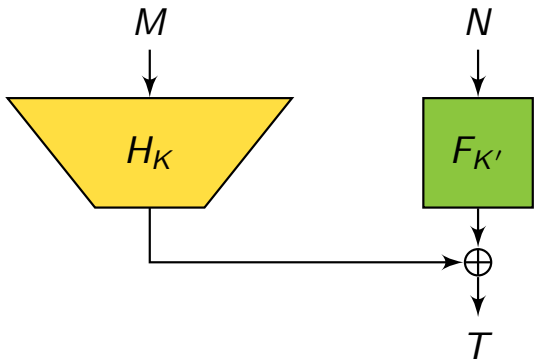
- based on an ε-almost xor-universal (ε-AXU) hash function H:

$$\forall M \neq M',\ \forall Y,\ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) \oplus H_K(M') = Y] \leq \varepsilon$$

- in practice, OTPs are replaced by a PRF applied to a nonce N
- H usually based on polynomial evaluation (GHASH, Poly1305)
- "optimal" security:

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WC}}(q, v) \leq v\varepsilon + \mathbf{Adv}^{\text{PRF}}_{F}(q + v)$$

The Wegman-Carter construction [GMS74, WC81]



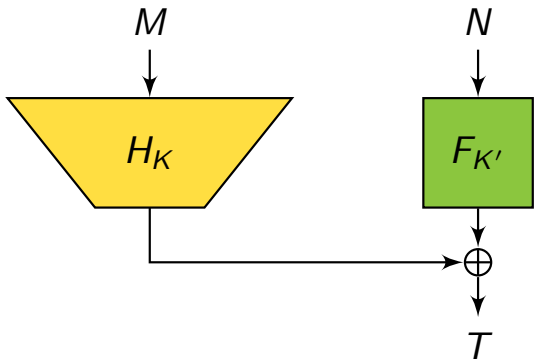
- based on an ε-almost xor-universal (ε-AXU) hash function H:

$$\forall M \neq M',\ \forall Y,\ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) \oplus H_K(M') = Y] \leq \varepsilon$$

- in practice, OTPs are replaced by a PRF applied to a nonce N
- H usually based on polynomial evaluation (GHASH, Poly1305)
- "optimal" security:

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WC}}(q, v) \leq v\varepsilon + \mathbf{Adv}^{\text{PRF}}_{F}(q + v)$$

The Wegman-Carter construction [GMS74, WC81]



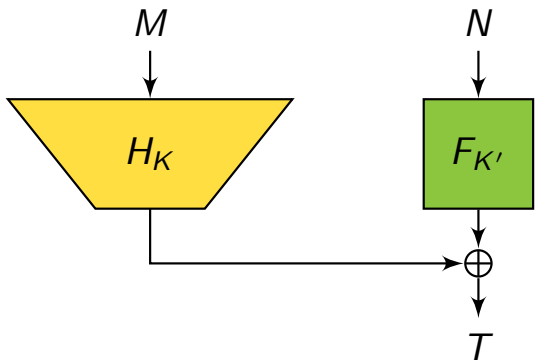
- based on an ε-almost xor-universal (ε-AXU) hash function H:

$$\forall M \neq M',\ \forall Y,\ \Pr[K \leftarrow_{\$} \mathcal{K} : H_K(M) \oplus H_K(M') = Y] \leq \varepsilon$$

- in practice, OTPs are replaced by a PRF applied to a nonce N
- H usually based on polynomial evaluation (GHASH, Poly1305)
- "optimal" security:

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WC}}(q, v) \leq v\varepsilon + \mathbf{Adv}^{\text{PRF}}_{F}(q + v)$$

Wegman-Carter weaknesses

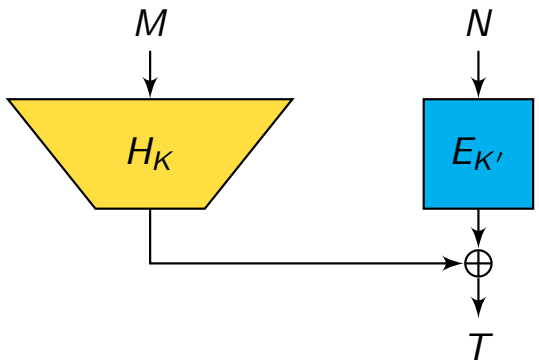


- in practice, F is replaced by a block cipher
  $\rightarrow$ Wegman-Carter-Shoup (WCS) construction
- provable security drops to birthday bound [Sho96, Ber05]

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WCS}}(q, v) \leq v\varepsilon + \frac{(q+v)^2}{2 \cdot 2^n} + \mathbf{Adv}^{\text{PRP}}_{E}(q+v)$$

- nonce-misuse problem: a single nonce repetition can completely break security [Jou06, HP08] (esp. for polynomial hashing)

Wegman-Carter weaknesses

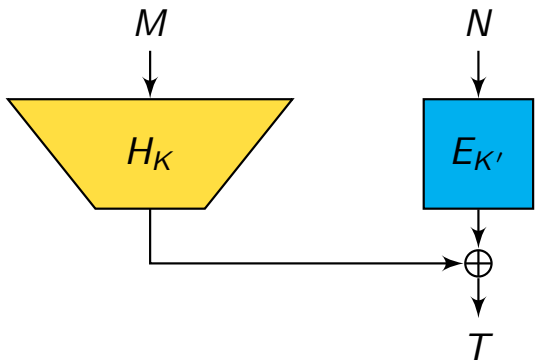


- in practice, F is replaced by a block cipher
  $\rightarrow$ Wegman-Carter-Shoup (WCS) construction
- provable security drops to birthday bound [Sho96, Ber05]

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WCS}}(q, v) \leq v\varepsilon + \frac{(q+v)^2}{2 \cdot 2^n} + \mathbf{Adv}^{\text{PRP}}_{E}(q+v)$$

- nonce-misuse problem: a single nonce repetition can completely break security [Jou06, HP08] (esp. for polynomial hashing)

Wegman-Carter weaknesses

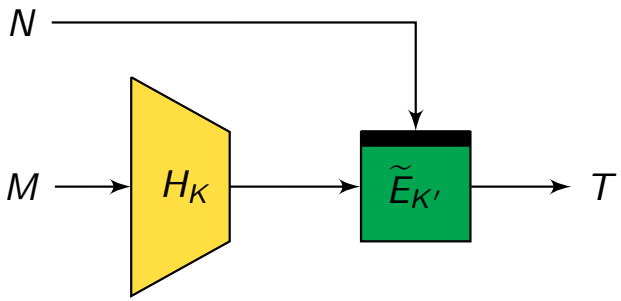


- in practice, F is replaced by a block cipher
  $\rightarrow$ Wegman-Carter-Shoup (WCS) construction
- provable security drops to birthday bound [Sho96, Ber05]

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{WCS}}(q, v) \leq v\varepsilon + \frac{(q+v)^2}{2 \cdot 2^n} + \mathbf{Adv}^{\text{PRP}}_{E}(q+v)$$

- nonce-misuse problem: a single nonce repetition can completely break security [Jou06, HP08] (esp. for polynomial hashing)

Construction 3: Nonce-as-Tweak (NaT)

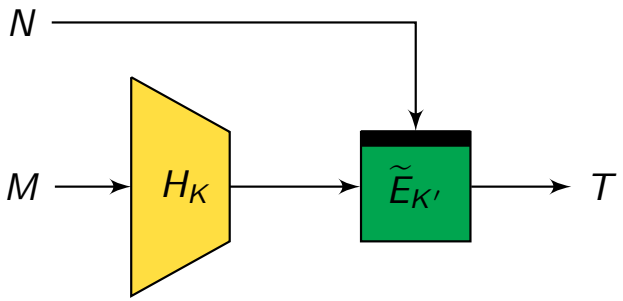


- if nonces don't repeat to often, $\widetilde{E}_{K'}$ is close to a perfect PRF
- graceful security degradation with maximal nonce multiplicity μ

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaT}}(q, v) \leq 2(\mu - 1)q\varepsilon + \mu v\varepsilon + (\ldots)$$

- can be seen as a special case of the (PRF-based) WMAC construction [BC09]

Construction 3: Nonce-as-Tweak (NaT)

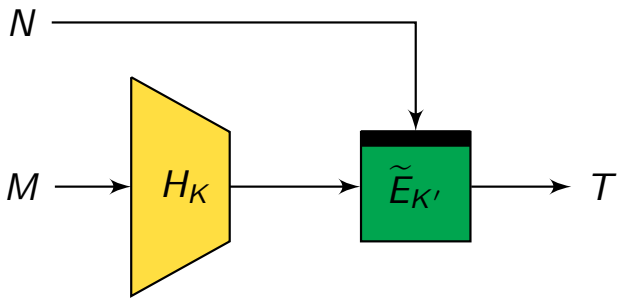


- if nonces don't repeat to often, $\widetilde{E}_{K'}$ is close to a perfect PRF
- graceful security degradation with maximal nonce multiplicity μ

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaT}}(q, v) \leq 2(\mu - 1)q\varepsilon + \mu v\varepsilon + (\ldots)$$

- can be seen as a special case of the (PRF-based) WMAC construction [BC09]

Construction 3: Nonce-as-Tweak (NaT)

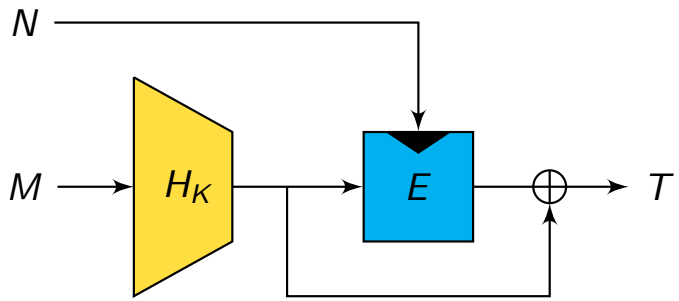


- if nonces don't repeat to often, $\widetilde{E}_{K'}$ is close to a perfect PRF
- graceful security degradation with maximal nonce multiplicity μ

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaT}}(q, v) \leq 2(\mu - 1)q\varepsilon + \mu v\varepsilon + (\ldots)$$

- can be seen as a special case of the (PRF-based) WMAC construction [BC09]

Construction 4: Nonce-as-Key (NaK)

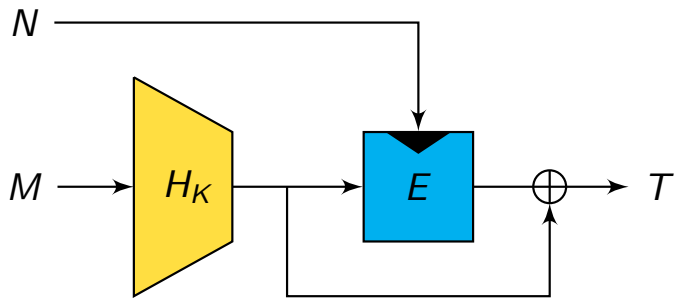


- provably secure in the ideal cipher model, assuming H is ε-AXU and ε'-uniform

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaK}}(q, v) \leq \mu q \varepsilon + (\ldots)$$

- graceful security degradation with maximal nonce multiplicity μ
- Davies-Meyer mode required to make the output function non-invertible!

Construction 4: Nonce-as-Key (NaK)

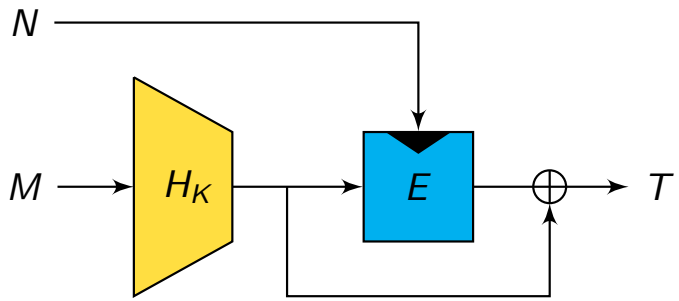


- provably secure in the ideal cipher model, assuming H is ε-AXU and ε'-uniform

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaK}}(q, v) \leq \mu q \varepsilon + (\ldots)$$

- graceful security degradation with maximal nonce multiplicity μ
- Davies-Meyer mode required to make the output function non-invertible!

Construction 4: Nonce-as-Key (NaK)



- provably secure in the ideal cipher model, assuming H is ε-AXU and ε'-uniform

$$\mathbf{Adv}^{\text{nonce-MAC}}_{\text{NaK}}(q, v) \leq \mu q \varepsilon + (\ldots)$$

- graceful security degradation with maximal nonce multiplicity μ
- Davies-Meyer mode required to make the output function non-invertible!

Conclusion

- we proposed four new MAC constructions secure beyond the birthday bound:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all security proofs rely on the standard H-coefficients technique [Pat08, CS14]
- our work does not address how to construct the UHF from a BC or TBC but many existing constructions can be used (PMAC/PMAC1 [BR02, Rog04], ZHASH [IMPS17], etc.)
- Nonce-as-Tweak (NaT) used in CAESAR candidate Deoxys v1.4

Conclusion

- we proposed four new MAC constructions secure beyond the birthday bound:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all security proofs rely on the standard H-coefficients technique [Pat08, CS14]
- our work does not address how to construct the UHF from a BC or TBC but many existing constructions can be used (PMAC/PMAC1 [BR02, Rog04], ZHASH [IMPS17], etc.)
- Nonce-as-Tweak (NaT) used in CAESAR candidate Deoxys v1.4

Conclusion

- we proposed four new MAC constructions secure beyond the birthday bound:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all security proofs rely on the standard H-coefficients technique [Pat08, CS14]
- our work does not address how to construct the UHF from a BC or TBC but many existing constructions can be used (PMAC/PMAC1 [BR02, Rog04], ZHASH [IMPS17], etc.)
- Nonce-as-Tweak (NaT) used in CAESAR candidate Deoxys v1.4

Conclusion

- we proposed four new MAC constructions secure beyond the birthday bound:

	stateless and deterministic	nonce-based/randomized
TBC-based	Hash-as-Tweak (HaT)	Nonce-as-Tweak (NaT)
BC-based	Hash-as-Key (HaK)	Nonce-as-Key (NaK)

- all security proofs rely on the standard H-coefficients technique [Pat08, CS14]
- our work does not address how to construct the UHF from a BC or TBC but many existing constructions can be used (PMAC/PMAC1 [BR02, Rog04], ZHASH [IMPS17], etc.)
- Nonce-as-Tweak (NaT) used in CAESAR candidate Deoxys v1.4

The end...

Thanks for your attention!

Comments or questions?

References I

John Black and Martin Cochran. MAC Reforgeability. In Orr Dunkelman, editor, *Fast Software Encryption - FSE 2009*, volume 5665 of *LNCS*, pages 345–362. Springer, 2009.

Daniel J. Bernstein. Stronger Security Bounds for Wegman-Carter-Shoup Authenticators. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 164–180. Springer, 2005.

John Black and Phillip Rogaway. A Block-Cipher Mode of Operation for Parallelizable Message Authentication. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 384–397. Springer, 2002.

Shan Chen and John Steinberger. Tight Security Bounds for Key-Alternating Ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 327–350. Springer, 2014. Full version available at `http://eprint.iacr.org/2013/222`.

References II


Edgar N. Gilbert, F. Jessie MacWilliams, and Neil J. A. Sloane. Codes which detect deception. *Bell System Technical Journal*, 53(3):405–424, 1974.


Helena Handschuh and Bart Preneel. Key-Recovery Attacks on Universal Hash Function Based MAC Algorithms. In David Wagner, editor, *Advances in Cryptology - CRYPTO 2008*, volume 5157 of *LNCS*, pages 144–161. Springer, 2008.


Tetsu Iwata, Kazuhiko Minematsu, Thomas Peyrin, and Yannick Seurin. ZMAC: A Fast Tweakable Block Cipher Mode for Highly Secure Message Authentication. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 (Proceedings, Part III)*, volume 10403 of *LNCS*, pages 34–65. Springer, 2017.


Antoine Joux. Authentication Failures in NIST Version of GCM. Comments submitted to NIST Modes of Operation Process, 2006. Available at `http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/800-38_Series-Drafts/GCM/Joux_comments.pdf`.

References III


Eik List and Mridul Nandi. ZMAC+ - An Efficient Variable-output-length Variant of ZMAC. *IACR Trans. Symmetric Cryptol.*, 2017(4):306–325, 2017.


Jacques Patarin. The "Coefficients H" Technique. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography - SAC 2008*, volume 5381 of *LNCS*, pages 328–345. Springer, 2008.


Phillip Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In Pil Joong Lee, editor, *Advances in Cryptology - ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 16–31. Springer, 2004.


Victor Shoup. On Fast and Provably Secure Message Authentication Based on Universal Hashing. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96*, volume 1109 of *LNCS*, pages 313–328. Springer, 1996.


Mark N. Wegman and Larry Carter. New Hash Functions and Their Use in Authentication and Set Equality. *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.