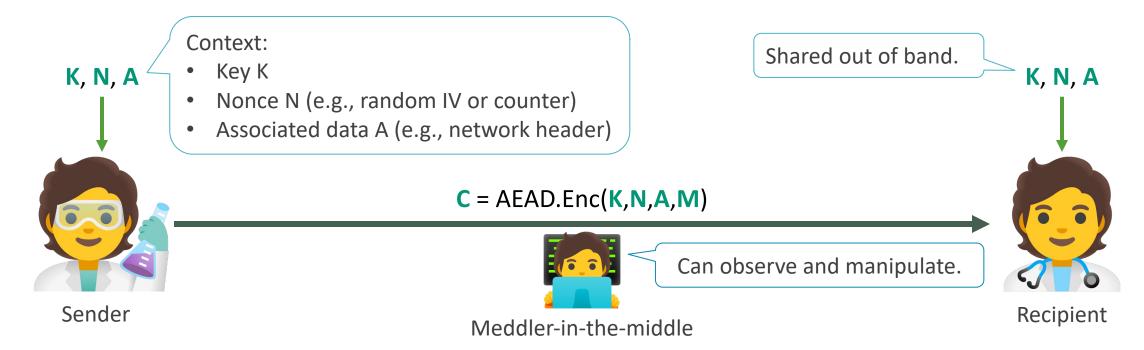
# Context Discovery and Commitment Attacks How to Break CCM, EAX, SIV, and More

Sanketh Menda, Julia Len, Paul Grubbs, and Thomas Ristenpart

Eurocrypt 2023

## **Authenticated Encryption with Associated Data (AEAD)**



#### **Standardized**

- 1. AES-GCM
- 2. ChaCha20/Poly1305
- 3. AES-GCM-SIV

#### **Provably Secure**

- 1. Confidentiality
- 2. Authenticity

#### **Recent Attacks on AEAD**

Fast Message Franking: From Invisible Salamanders to Encryptment\*

Yevgeniy Dodis<sup>1</sup>, Paul Grubbs<sup>2,†</sup>, Thomas Ristenpart<sup>2</sup>, Joanne Woodage<sup>3,†</sup>

<sup>1</sup> New York University <sup>2</sup> Cornell Tech

<sup>3</sup> Royal I

**Key Material** A Blog about Security and Cryptography About Contact CRYPTOGRAPHY Invisible Salamanders in AEC-CCM-SIV

These attacks work in new threat models!

Security 2021. This is the full version.

How to Abuse and Fix

Ange Albertini<sup>1</sup>, Thai Duong

These attacks exploit lack of *key commitment*: An adversary can find keys K<sub>1</sub>, K<sub>2</sub> and ciphertext C s.t. C can be decrypted under both keys

Abstract

Authenticated encryption (AE) is used in a wide varie applications, potentially in settings for which it was not originally designed. Recent research tries to understand what happens when AE is not used as prescribed by its designers. A question given relatively little attention is whether an AE scheme guarantees "key commitment": ciphertext should only document to a valid plaintant under the last used to concrete the

of cryptographic algorithms such as SHA-1 [SBK<sup>+</sup>17].

The vast majority of applications should default to using authenticated encryption (AE) [BN00, KY00], a well-studied primitive which avoids the pitfalls of unauthenticated SKE with relatively small performance overhead. AE schemes are used in widely adopted protocols like TLS [Res18], standardschemes are not committing with respect to their keys. We detail novel adaptive chosen ciphertext attacks that exploit partitioning oracles to efficiently recover passwords and deanonymize anonymous communications. The attacks utilize efficient key multi-collision algorithms — a cryptanalytic acks

s Ristenpart

aps because attacks exploiting lack of robustness in relatively niche applications like auction 3] or recently as an integrity issue in moderation ed messaging [22,30].

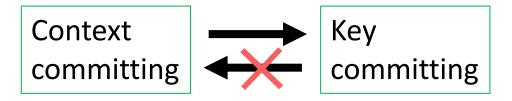
Q

oduce partitioning oracle attacks, a new type of These are similar to previous attacks considered in the password-authenticated key exchange (PAKE) literature [11, 72, 98]; we provide a unifying attack framework that transcends PAKE and show partitioning oracle attacks that exploit weaknesses in widely used non-committing AEAD schemes. Briefly, a partitioning oracle arises when an adversary can: (1) efficiently craft cinhertexts that suc-

## **Context Commitment Security [BH22]**

For AEAD, computationally efficient to find  $(K_1, N_1, A_1), (K_2, N_2, A_2)$  and C such that decryption  $M_1 \leftarrow \text{AEAD.Dec}(K_1, N_1, A_1, C)$ 

 $M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$ 



## **Commitment Attacks (Before this Paper)**

Scheme	Key Committing	Context Committing
GCM	<b>X</b> [GLR17, DGRW19]	XS
AES-SIV		
CCM	?	2
EAX	?	2
OCB3	X [ADGKLS20]	XS
PaddingZeros	✓[ADGKLS20]	
KeyHashing	✓[ADGKLS20]	2
CAU-C1	<b>✓</b> [BH22]	

#### Do CCM and EAX provide key commitment?

Asked 2 years, 2 months ago Modified 2 years, 2 months ago Viewed 171 times



10







In an interesting paper called <u>"Partitioning Oracle Attacks" by</u>
<u>Julia Len, Paul Grubbs & Thomas Ristenpart</u> an attack is
presented on 1.5 pass AEAD schemes that utilize GMAC (GCM,
AES-GCM, AES-GCM-SIV) and Poly1305 which is often used with
a ChaCha/Salsa variant.

In the paper they mention that older schemes based on HMAC authentication are not vulnerable against this attack because they provide the key commitment property.

Do CCM with CBC-MAC and EAX with AES-CMAC provide key commitment as well? Or is - for instance - the output size of the MAC constructions too small? If they don't provide full key commitment, are they susceptible to this attack?

cryptanalysis cbc-mac cmac decryption-oracle eax

Share Improve this question Follow





**Sophie, indistinguishable from random noise** @SchmiegSophie · Sep 7, 2020 Given that I've now done the work of setting up the blog, I used the chance to write a bit about invisible salamanders, i.e. ciphertexts that decrypt to two different plaintexts depending on the used key, for AES-GCM and AES-GCM-SIV.



kevmaterial.net

Invisible Salamanders in AES-GCM-SIV

By now, many people have run across the Invisible Salamander paper about the interesting property of ...

 $\bigcirc$  4

17 40

♡ 128

 $_{1}\uparrow_{1}$ 



JP Aumasson @veorq · Sep 9, 2020

Replying to @SchmiegSophie no such trick for SIV-AES I guess?

1

 $\mathcal{O}_2$ 

, **1** 



**Sophie, indistinguishable from random noise** @SchmiegSophie

Replying to @veorq

Yeah, for AES-SIV it's much harder. You can construct a collision for the CMAC, but it's non-linear, so the encrypted text would not collide easily.

Similar issue if you replace GMAC with Poly1305 in AES-GCM-SIV, because you mix characteristic.

1:44 PM · Sep 9, 2020 · Twitter for Android

## **Commitment Attacks (After this Paper)**

Scheme	Key Committing	Context Committing
GCM	<b>X</b> [GLR17, DGRW19]	XS
AES-SIV	X	XS
CCM	X	<b>X5</b>
EAX	X	XS
OCB3	X [ADGKLS20]	XG
PaddingZeros	✓[ADGKLS20]	<b>X</b>
KeyHashing	✓[ADGKLS20]	<b>X</b>
CAU-C1	<b>✓</b> [BH22]	<b>X</b>



= new result

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

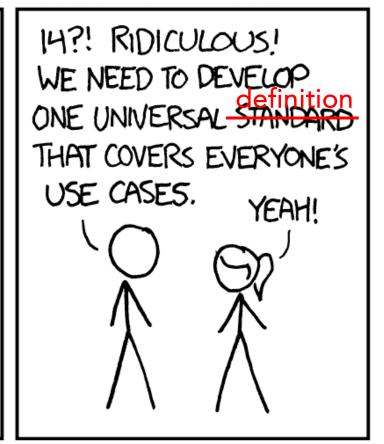
New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

## Don't we already have committing security definitions?

definitions
HOW STANDARDS PROLIFERATE;
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)

SITUATION:
THERE ARE
14 COMPETING
STANDARDS.
definitions



SITUATION: THERE ARE 15 COMPETING STANDARDS. definitions

## **Ambiguity in Key Commitment Definitions**

```
\frac{\operatorname{FROB}_{\mathbf{AE}}^{\mathcal{A}}(\lambda):}{(C, K_1, K_2) \twoheadleftarrow \mathcal{A}(1^{\lambda})}
if K_1 = K_2 return 0
M_1 \leftarrow \mathbf{Dec}(K_1, C)
M_2 \leftarrow \mathbf{Dec}(K_2, C)
return (M_1 \neq \bot \land M_2 \neq \bot)
```

[FOR17] doesn't mention associated data, and implicitly requires the same nonce.

```
\frac{\operatorname{FROB}_{\mathsf{SE}}^{\mathcal{A}}}{((H,K),(H',K'),C)} \leftarrow \mathcal{A}
\operatorname{If} K = K' \text{ then Return false}
M \leftarrow \operatorname{Dec}(K,H,C)
M' \leftarrow \operatorname{Dec}(K',H',C)
\operatorname{Return} (M \neq \bot) \land (M' \neq \bot)
```

[GLR17] allows different associated data, but still implicitly requires the same nonce.

We define targeted multi-key collision resistance (TMKCR) security by the following game. It is parameterized by a scheme AEAD and a target key set  $\mathbb{K} \subseteq \mathcal{K}$ . A possibly randomized adversary  $\mathcal{A}$  is given input a target set  $\mathbb{K}$  and must produce nonce  $N^*$ , associated data  $AD^*$ , and ciphertext  $C^*$  such that  $\text{AuthDec}_K(N^*, AD^*, C^*) \neq \bot$  for all  $K \in \mathbb{K}$ . We define the advantage via

$$\mathbf{Adv}^{\mathsf{tmk\text{-}cr}}_{\mathsf{AEAD},\mathbb{K}}(\mathcal{A}) = \Pr \Big[ \, \mathsf{TMKCR}^{\mathcal{A}}_{\mathsf{AEAD},\mathbb{K}} \Rightarrow \mathsf{true} \, \Big]$$

[LGR20] requires same nonces and associated data.

```
For AEAD, computationally efficient to find (K_1, N_1, A_1), (K_2, N_2, A_2) and C such that decryption M_1 \leftarrow \text{AEAD.Dec}(K_1, N_1, A_1, C)M_2 \leftarrow \text{AEAD.Dec}(K_2, N_2, A_2, C) succeeds.
```

#### Step 1: Arbitrary Predicates

```
For AEAD, computationally efficient to find (K_1, N_1, A_1), (K_2, N_2, A_2) and C such that P((K_1, N_1, A_1), (K_2, N_2, A_2)) for predicate P and decryption M_1 \leftarrow \text{AEAD.Dec}(K_1, N_1, A_1, C) M_2 \leftarrow \text{AEAD.Dec}(K_2, N_2, A_2, C) succeeds.
```

## Step 1: Arbitrary Predicates

```
For AEAD, computationally efficient to find (K_1, N_1, A_1), (K_2, N_2, A_2) and C such that P((K_1, N_1, A_1), (K_2, N_2, A_2)) for predicate P and decryption M_1 \leftarrow \text{AEAD.Dec}(K_1, N_1, A_1, C) M_2 \leftarrow \text{AEAD.Dec}(K_2, N_2, A_2, C) succeeds.
```

Notion	Predicate
Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$

## Step 1: Arbitrary Predicates

```
For AEAD, computationally efficient to find (K_1, N_1, A_1), (K_2, N_2, A_2) and C such that P((K_1, N_1, A_1), (K_2, N_2, A_2)) for predicate P and decryption M_1 \leftarrow \text{AEAD.Dec}(K_1, N_1, A_1, C) M_2 \leftarrow \text{AEAD.Dec}(K_2, N_2, A_2, C) succeeds.
```

	Notion	Predicate
	Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
ive	CMT-k	$K_1 \neq K_2$
Permissive	CMT-n	$N_1 \neq N_2$
Per	СМТ-а	<b>A</b> <sub>1</sub> ≠ <b>A</b> <sub>2</sub>

## **Step 1: Arbitrary Predicates**

For AEAD, computationally efficient to find  $(K_1, N_1, A_1)$ ,  $(K_2, N_2, A_2)$  and C such that  $P((K_1, N_1, A_1), (K_2, N_2, A_2))$  for predicate P

$$M_1 \leftarrow AEAD.Dec(K_1, N_1, A_1, C)$$

$$M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$$

succeeds.

and decryption

	Notion	Predicate
	Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
ive Ne	CMT-k	<b>K</b> <sub>1</sub> ≠ <b>K</b> <sub>2</sub>
Permissive	CMT-n	$N_1 \neq N_2$
Pel	CMT-a	$A_1 \neq A_2$
ve	CMT-k*	$K_1 \neq K_2 \land (N_1, A_1) = (N_2, A_2)$
Restrictive	CMT-n*	$N_1 \neq N_2 \wedge (K_1, A_1) = (K_2, A_2)$
Re	CMT-a*	$A_1 \neq A_2 \wedge (K_1, N_1) = (K_2, N_2)$

## Step 2: Target Selection

For AEAD, given

$$K_1, K_2 \leftarrow \$ \{0, 1\}^k$$
 [target selection]

computationally efficient to find

$$(K_1, N_1, A_1), (K_2, N_2, A_2)$$
 and C

such that

$$P((K_1, N_1, A_1), (K_2, N_2, A_2))$$
 for predicate  $P$  and decryption

$$M_1 \leftarrow AEAD.Dec(K_1, N_1, A_1, C)$$

$$M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$$

	Notion	Predicate
	Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
ive Ne	CMT-k	<b>K</b> <sub>1</sub> ≠ <b>K</b> <sub>2</sub>
Permissive	CMT-n	$N_1 \neq N_2$
Pel	CMT-a	<b>A</b> <sub>1</sub> ≠ <b>A</b> <sub>2</sub>
Ve	CMT-k*	$K_1 \neq K_2 \land (N_1, A_1) = (N_2, A_2)$
Restrictive	CMT-n*	$N_1 \neq N_2 \wedge (K_1, A_1) = (K_2, A_2)$
Re	CMT-a*	$A_1 \neq A_2 \wedge (K_1, N_1) = (K_2, N_2)$

## Step 2: Target Selection

For AEAD, given

 $C \leftarrow AEAD.Enc(K_1, N_1, A_1, M_1)$  [target selection]

and not given

K<sub>1</sub>, N<sub>1</sub>, A<sub>1</sub> [target hiding]

computationally efficient to find

$$(K_2, N_2, A_2)$$

such that

 $P((K_1, N_1, A_1), (K_2, N_2, A_2))$  for predicate P

and decryption

 $M_1 \leftarrow AEAD.Dec(K_1, N_1, A_1, C)$ 

 $M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$ 

	Notion	Predicate
	Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
ive	CMT-k	$K_1 \neq K_2$
Permissive	CMT-n	$N_1 \neq N_2$
Per	CMT-a	<b>A</b> <sub>1</sub> ≠ <b>A</b> <sub>2</sub>
Ve	CMT-k*	$K_1 \neq K_2 \land (N_1, A_1) = (N_2, A_2)$
Restrictive	CMT-n*	$N_1 \neq N_2 \wedge (K_1, A_1) = (K_2, A_2)$
Re	CMT-a*	$A_1 \neq A_2 \wedge (K_1, N_1) = (K_2, N_2)$

## Step 2: Target Selection

For AEAD, given

 $C \leftarrow AEAD.Enc(K_1, N_1, A_1, M_1)$  [target selection]

and not given

K<sub>1</sub>, N<sub>1</sub>, A<sub>1</sub> [target hiding]

computationally efficient to find

 $(K_2, N_2, A_2)$ 

such that

 $P((K_1, N_1, A_1), (K_2, N_2, A_2))$  for predicate P

and decryption

 $M_1 \leftarrow AEAD.Dec(K_1, N_1, A_1, C)$ 

 $M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$ 

		Notion	Predicate
		Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
	Ne	CMT-k	<b>K</b> <sub>1</sub> ≠ <b>K</b> <sub>2</sub>
Restrictive Permissive	rmissi	CMT-n	$N_1 \neq N_2$
	Per	CMT-a	<b>A</b> <sub>1</sub> ≠ <b>A</b> <sub>2</sub>
	ve	CMT-k*	$K_1 \neq K_2 \land (N_1, A_1) = (N_2, A_2)$
	stricti	CMT-n*	$N_1 \neq N_2 \land (K_1, A_1) = (K_2, A_2)$
	CMT-a*	$A_1 \neq A_2 \wedge (K_1, N_1) = (K_2, N_2)$	

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

#### SIV vs. GCM for CMT-k\* attacks

Recall: CMT-k\* means nonces and associated data of both contexts must be the same

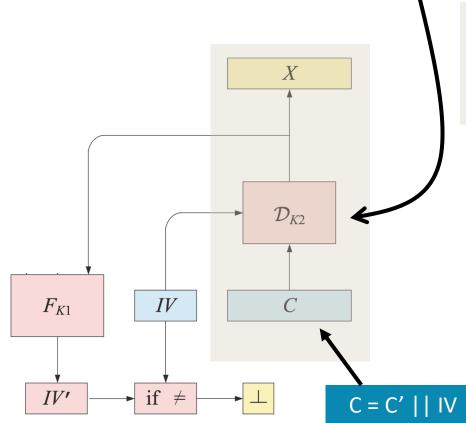
GCM uses a highly-structured polynomial MAC

Finding CMT-k\* attack is *easy*: this is just solving a simple system of 2 linear equations

SIV does not use a polynomial MAC so we can't adapt the attacks we already have...

## **The Original SIV Mode**

For simplicity, we assume no associated data

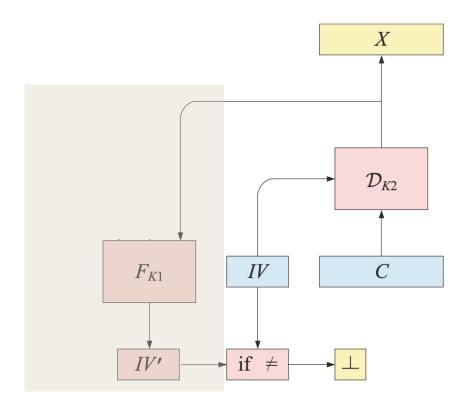


CTR mode with "synthetic IV" in ciphertext as initial counter

- 1. Use CTR mode with key K2 to decrypt the ciphertext and recover message.
- 2. Recompute the "synthetic IV" from the message using S2V[CMAC] with key K1.
- 3. Compare this computed synthetic IV with that stored as part of the ciphertext:
  - a) If they are different, then reject.
  - b) Otherwise, return message.

#### The Original SIV Mode

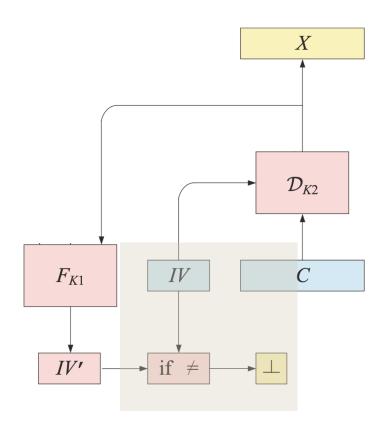
For simplicity, we assume no associated data



- 1. Use CTR mode with key K2 to decrypt the ciphertext and recover message.
- 2. Recompute the "synthetic IV" from the message using S2V[CMAC] with key K1.
- Compare this computed synthetic IV with that stored as part of the ciphertext:
  - a) If they are different, then reject.
  - b) Otherwise, return message.

#### **The Original SIV Mode**

For simplicity, we assume no associated data



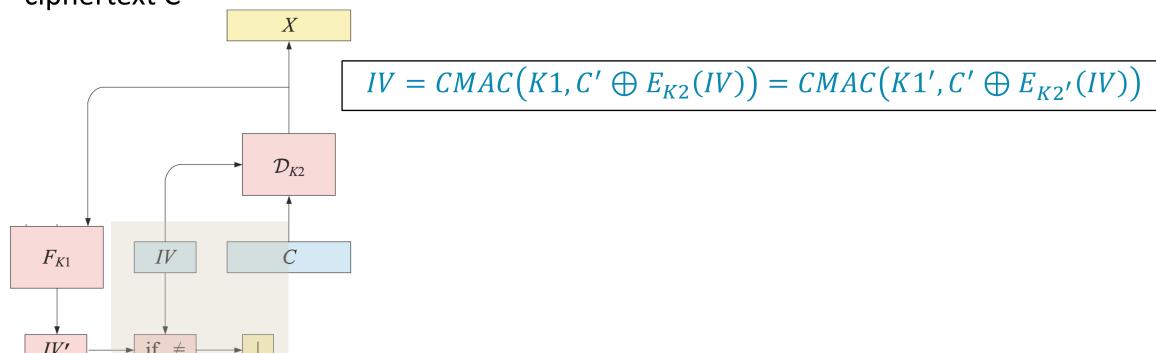
- 1. Use CTR mode with key K2 to decrypt the ciphertext and recover message.
- 2. Recompute the "synthetic IV" from the message using S2V[CMAC] with key K1.
- 3. Compare this computed synthetic IV with that stored as part of the ciphertext:
  - a) If they are different, then reject.
  - b) Otherwise, return message.

#### CMT-k\* attack on SIV

For simplicity, we'll consider 1-block ciphertexts with no associated data or nonce

**Goal**: Find two keys (K1, K2)  $\neq$  (K1', K2') and ciphertext C (of the form C' | | IV) so that decryption of C under both keys succeeds

➤ This means that the computed synthetic IV's match the stored IV that is part of ciphertext C



#### CMT-k\* attack on SIV

For simplicity, we'll consider 1-block ciphertexts with no associated data or nonce

Goal: Find two keys (K1, K2)  $\neq$  (K1', K2') and ciphertext C  $\leftarrow$ C' | | IV such that:

$$\left(E_{K1}^{-1}(IV) \oplus \left(2 \cdot E_{K1}(0^n)\right) \oplus E_{K1}\left(2 \cdot E_{K1}(0^n)\right)\right) \oplus E_{K2}(IV) 
\oplus \left(E_{K1'}^{-1}(IV) \oplus \left(2 \cdot E_{K1'}(0^n)\right) \oplus E_{K1'}\left(2 \cdot E_{K1'}(0^n)\right)\right) \oplus E_{K2'}(IV) = 0^n$$

If we model block cipher E as an ideal cipher, then this looks *very* close to the Generalized Birthday Problem!

## CMT-k\* attack on SIV: Using Generalized Birthday Problem

For simplicity, we'll consider 1-block ciphertexts with no associated data or nonce

#### **4-list Birthday Problem**:

Given lists L1, L2, L3, L4 of elements drawn uniformly and independently at random from  $\{0,1\}^n$ , find  $x1 \in L1$ ,  $x2 \in L2$ ,  $x3 \in L3$ ,  $x4 \in L4$  s.t.  $x1 \oplus x2 \oplus x3 \oplus x4 = 0^n$ 

Wagner gives the k-tree algorithm to solve this in  $O(2^{n/3})$  space and time [W02]

$$\left(E_{K1}^{-1}(IV) \oplus \left(2 \cdot E_{K1}(0^n)\right) \oplus E_{K1}\left(2 \cdot E_{K1}(0^n)\right)\right) \oplus E_{K2}(IV) \\
\oplus \left(E_{K1'}^{-1}(IV) \oplus \left(2 \cdot E_{K1'}(0^n)\right) \oplus E_{K1'}\left(2 \cdot E_{K1'}(0^n)\right)\right) \oplus E_{K2'}(IV) = 0^n$$

Problem: These values are not drawn uniformly and independently at random from



$$F_1(K1) \oplus F_2(K2) \oplus F_3(K1') \oplus F_4(K2') = 0^n$$

## CMT-k\* attack on SIV: Using Generalized Birthday Problem

For simplicity, we'll consider 1-block ciphertexts with no associated data or nonce

- We show that we can upper bound the distinguishability between the distribution formed by the values chosen to make the list and uniformly random distribution
- We then show we can apply Wagner's k-tree algorithm with the distributions we have and still have high probability of finding collisions
- We show that with high probability we can find a collision in time  $^2$ 253, making it practical and sufficiently damaging to rule out SIV as suitable for contexts where key commitment matters

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

#### **Our Contributions**

New granular framework for context commitment

Key commitment attack against the original SIV mode

New context commitment security notion: context discovery

Context discovery attacks against GCM, OCB3, EAX, CCM, SIV

## **Revisiting the Framework for Context Committing Security**

For AEAD, given

 $C \leftarrow AEAD.Enc(K_1, N_1, A_1, M_1)$  [target selection]

and not given

K<sub>1</sub>, N<sub>1</sub>, A<sub>1</sub> [target hiding]

computationally efficient to find

 $(K_2, N_2, A_2)$ 

such that

 $P((K_1, N_1, A_1), (K_2, N_2, A_2))$  for predicate P

and decryption

 $M_1 \leftarrow AEAD.Dec(K_1, N_1, A_1, C)$ 

 $M_2 \leftarrow AEAD.Dec(K_2, N_2, A_2, C)$ 

		Notion	Predicate
		Context Commitment	$(K_1, N_1, A_1) \neq (K_2, N_2, A_2)$
	Ne	CMT-k	<b>K</b> <sub>1</sub> ≠ <b>K</b> <sub>2</sub>
Restrictive Permissive	missi	CMT-n	$N_1 \neq N_2$
	Pel	CMT-a	$A_1 \neq A_2$
	ve	CMT-k*	$K_1 \neq K_2 \land (N_1, A_1) = (N_2, A_2)$
	stricti	CMT-n*	$N_1 \neq N_2 \wedge (K_1, A_1) = (K_2, A_2)$
	Re	CMT-a*	$A_1 \neq A_2 \wedge (K_1, N_1) = (K_2, N_2)$

#### **Context Discoverability Security**

#### **CDY Security**

For AEAD, given

**C** [target selection]

computationally efficient to find

K, N, A

such that decryption

 $M \leftarrow AEAD.Dec(K, N, A, C)$ 

#### **Context Discoverability Security**

#### **CDY Security**

For AEAD, given

**C** [target selection]

computationally efficient to find

such that decryption

$$M \leftarrow AEAD.Dec(K, N, A, C)$$

- Context Discoverability security is to Context Committing security for AEAD as preimage resistance is to collision-resistance for hash functions
- We also show that if an AEAD scheme is "context compressing" (meaning: ciphertexts are decryptable under more than one context), then Context Committing security implies Context Discoverability security
- We show Context Discoverability attacks for CCM, EAX, SIV, GCM, and OCB3
- Context Discoverability allows us to better communicate attacks and threat models

#### **Conclusion**

#### Full version available on eprint:

https://ia.cr/2023/526

Scheme	Key Committing	Context Committing
GCM	[GLR17, DGRW19]	<b>G</b>
AES-SIV	*	<b>5</b>
CCM	*	<b>5</b>
EAX	*	<b>5</b>
OCB3	[ADGKLS20]	<b>G</b>
PaddingZeros	[ADGKLS20]	**
KeyHashing	[ADGKLS20]	*
CAU-C1	[BH22]	*



= new result

## References

[W02]	David Wagner. A Generalized Birthday Problem. https://www.iacr.org/archive/crypto2002/24420288/24420288.pdf
[FOR17]	Pooya Farshim, Claudio Orlandi, and Răzvan Roşie. Security of Symmetric Primitives under Incorrect Usage of Keys. ia.cr/2017/288
[GLR17]	Paul Grubbs, Jiahui Lu, and Thomas Ristenpart. Message franking via committing authenticated encryption. ia.cr/2017/664
[DGRW19]	Yevgeniy Dodis, Paul Grubbs, Thomas Ristenpart, and Joanne Woodage. Fast message franking: From invisible salamanders to encryptment. ia.cr/2019/016
[LGR20]	Julia Len, Paul Grubbs, and Thomas Ristenpart. Partitioning Oracle Attacks. ia.cr/2020/1491
[ADGKLS20]	Ange Albertini, Thai Duong, Shay Gueron, Stefan Kölbl, Atul Luykx, and Sophie Schmieg. How to Abuse and Fix Authenticated Encryption Without Key Commitment. ia.cr/2020/1456
[BH22]	Mihir Bellare and Viet Tung Hoang. Efficient schemes for committing authenticated encryption. ia.cr/2022/268
[RS07]	Philip Rogaway and Thomas Shrimpton. The SIV Mode of Operation for Deterministic Authenticated-Encryption (Key Wrap) and Misuse-Resistant Nonce-Based Authenticated-Encryption. https://web.cs.ucdavis.edu/~rogaway/papers/siv.pdf