Radboud University

Efficient Instances of Docked Double Decker With AES, and Application to Authenticated Encryption

Christoph Dobraunig[1], Krystian Matusiewicz[2], Bart Mennink[3], Alexander Tereschenko[2]

[1]Intel USA, [2]Intel Poland, [3]Radboud University

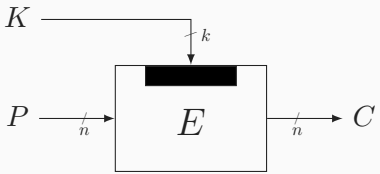
ASK 2024

December 16, 2024

ESCADA

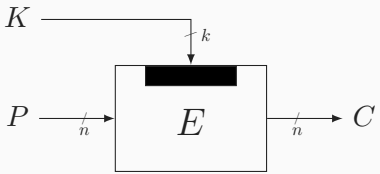
Introduction

Blockciphers



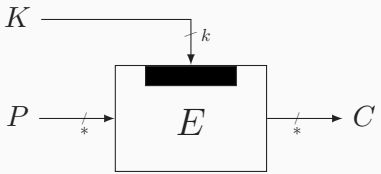
- Plaintext P encrypted to ciphertext C with secret key K
- Fixed block size

Blockciphers



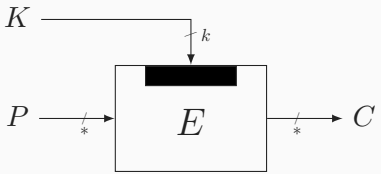
- Plaintext P encrypted to ciphertext C with secret key K
- Fixed block size
- In order to encrypt variable sized plaintexts, we need a mode of operation
  - These modes require a nonce

Wide Blockciphers



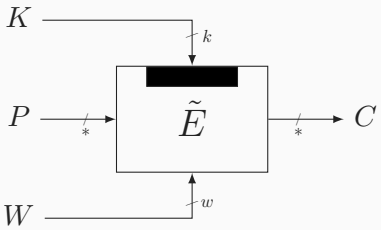
- Alternatively, we can design a wide block cipher
- A wide block cipher is a block cipher with a variable block size

Wide Blockciphers



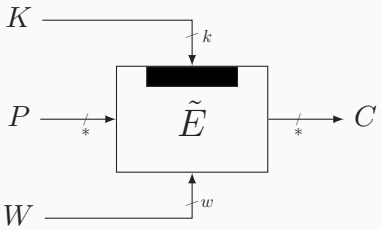
- Alternatively, we can design a wide block cipher
- A wide block cipher is a block cipher with a variable block size
- Every part of the output (ideally) depends on every part of the input

Tweakable Wide Blockciphers



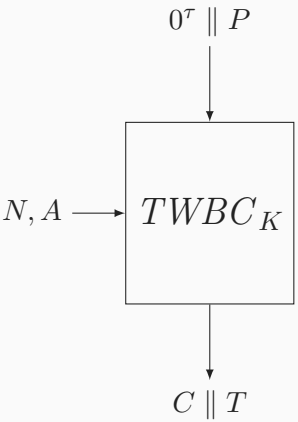
- A tweakable wide block cipher additionally has a tweak
- Tweak W public, ciphertext completely changes with a different tweak

Tweakable Wide Blockciphers



- A tweakable wide block cipher additionally has a tweak
- Tweak W public, ciphertext completely changes with a different tweak
- Applications:
  - disk encryption, where every sector gets its own tweak
  - robust authenticated encryption

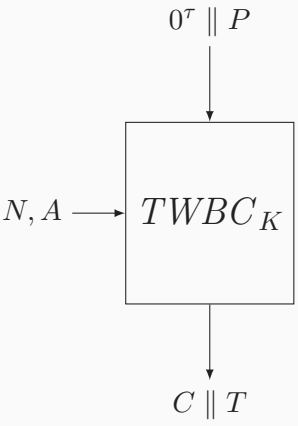
Application to Robust Authenticated Encryption [HKR15]



Robust Authenticated Encryption

- Encryption:
  - Prepend τ zeros to P
  - Evaluate with $TWBC_K$ to obtain $C \parallel T$
- Decryption:
  - Decrypt $C \parallel T$ using $TWBC_K^{-1}$
  - If result starts with τ zeros: output P

Application to Robust Authenticated Encryption [HKR15]



Robust Authenticated Encryption

- Encryption:
  - Prepend τ zeros to P
  - Evaluate with $TWBC_K$ to obtain $C \parallel T$
- Decryption:
  - Decrypt $C \parallel T$ using $TWBC_K^{-1}$
  - If result starts with τ zeros: output P

- Solves many issues present in GCM...
- ... but we first need to design $TWBC$

NIST's Incentive to Develop Accordion Mode

- March 2024: US NIST announced quest for tweakable wide blockciphers
- There was a workshop (June 2024) aimed to discuss ideas on requirements, designs, security goals, targets, ...

NIST's Incentive to Develop Accordion Mode

- March 2024: US NIST announced quest for tweakable wide blockciphers
- There was a workshop (June 2024) aimed to discuss ideas on requirements, designs, security goals, targets, ...

- Quote from the website:
  NIST plans to develop a new mode of the AES that is a tweakable, variable-input-length-strong pseudorandom permutation (VIL-SPRP) with a reduction proof to the security of the underlying block cipher.

NIST's Incentive to Develop Accordion Mode

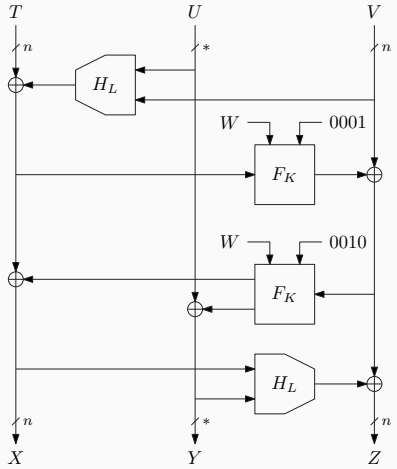
- March 2024: US NIST announced quest for tweakable wide blockciphers
- There was a workshop (June 2024) aimed to discuss ideas on requirements, designs, security goals, targets, ...

- Quote from the website:
  NIST plans to develop a new mode of the AES that is a tweakable, variable-input-length-strong pseudorandom permutation (VIL-SPRP) with a reduction proof to the security of the underlying block cipher.

Now: high-level idea of our proposals

Docked Double Decker

Docked Double Decker [GDM19]



Building Blocks

- F_K: stream cipher
- H_L: universal hash

Construction

- Feistel-like structure
- Outer lanes of fixed size
- Inner lane of variable size

Generic Security

- Assume
  - F_K is PRF-secure
  - H_L is ϵ-XOR-universal
- Adversary makes q queries and at most q_W queries per tweak W
- Docked double decker is secure up to approximately

$$\sum_{W \in \{0,1\}^w} \binom{q_W}{2} \epsilon + \mathbf{Adv}_F^{\mathrm{prf}}(2q)$$

Security of Docked Double Decker [GDM19]

Generic Security

- Assume
  - F_K is PRF-secure
  - H_L is ϵ-XOR-universal
- Adversary makes q queries and at most q_W queries per tweak W
- Docked double decker is secure up to approximately

$$\sum_{W \in \{0,1\}^w} \binom{q_W}{2} \epsilon + \mathbf{Adv}_F^{\mathrm{prf}}(2q)$$

Implications

- Birthday bound secure in n in general case
- Security significantly increases when tweaks are not used too often

Application to Disk Encryption on SSDs

- Docked double decker is very suitable for disk encryption
  - Disks are separated in sectors
  - Block size is equal to the sector size
  - Physical sector number used as tweak

Application to Disk Encryption on SSDs

- Docked double decker is very suitable for disk encryption
  - Disks are separated in sectors
  - Block size is equal to the sector size
  - Physical sector number used as tweak
- Sectors in SSDs have a limited lifetime
  - They get damaged every time data is written
- The Kingston UV500 960 GB has $N = 2^{28}$ sectors, where every sector can be written at most ≈ 500 times

Application to Disk Encryption on SSDs

- Docked double decker is very suitable for disk encryption
  - Disks are separated in sectors
  - Block size is equal to the sector size
  - Physical sector number used as tweak
- Sectors in SSDs have a limited lifetime
  - They get damaged every time data is written
- The Kingston UV500 960 GB has $N = 2^{28}$ sectors,
  where every sector can be written at most ≈ 500 times
  - Without tweak separation, secure when $\binom{500N}{2}\epsilon \ll 1$, i.e., $\epsilon \ll 2^{-73}$
  - With tweak separation this improves to $N\binom{500}{2}\epsilon \ll 1$, i.e., $\epsilon \ll 2^{-45}$

Application to Disk Encryption on SSDs

- Docked double decker is very suitable for disk encryption
  - Disks are separated in sectors
  - Block size is equal to the sector size
  - Physical sector number used as tweak
- Sectors in SSDs have a limited lifetime
  - They get damaged every time data is written
- The Kingston UV500 960 GB has $N = 2^{28}$ sectors,
  where every sector can be written at most ≈ 500 times
  - Without tweak separation, secure when $\binom{500N}{2}\epsilon \ll 1$, i.e., $\epsilon \ll 2^{-73}$
  - With tweak separation this improves to $N\binom{500}{2}\epsilon \ll 1$, i.e., $\epsilon \ll 2^{-45}$
- One could take weaker (and thus cheaper) H_L without security sacrifice!

Efficient Instantiation

Goals

- Instantiation using components as used in NIST standardized schemes:
  - AES [DR02, DR20]
  - Operations in binary extension fields, e.g., as in GHASH [MV04]

Goals

- Instantiation using components as used in NIST standardized schemes:
  - AES [DR02, DR20]
  - Operations in binary extension fields, e.g., as in GHASH [MV04]
- Present birthday bound secure $ddd\text{-}AES$ and beyond birthday bound secure $bbb\text{-}ddd\text{-}AES$ that seamlessly fit NIST's accordion idea

Goals and Hurdles

Goals

- Instantiation using components as used in NIST standardized schemes:
  - AES [DR02, DR20]
  - Operations in binary extension fields, e.g., as in GHASH [MV04]
- Present birthday bound secure $ddd\text{-}AES$ and beyond birthday bound secure $bbb\text{-}ddd\text{-}AES$ that seamlessly fit NIST's accordion idea

Hurdles

- AES is not a tweakable blockcipher
- AES is rather small (circular reasoning?)
- AES in typical stream cipher modes only gives birthday bound security

ddd-AES

- H_L instantiated using Polyval (sibling of GHASH)
- F_K instantiated as variant of CTR: tweak used to randomize inputs to AES_K

ddd-AES

- H_L instantiated using Polyval (sibling of GHASH)
- F_K instantiated as variant of CTR: tweak used to randomize inputs to AES_K

bbb-ddd-AES

- H_L instantiated using Polyval (sibling of GHASH)
- F_K instantiated as variant of CENC: tweak used to randomize inputs to AES_K

ddd-AES

- H_L instantiated using Polyval (sibling of GHASH)
- F_K instantiated as variant of CTR: tweak used to randomize inputs to AES_K

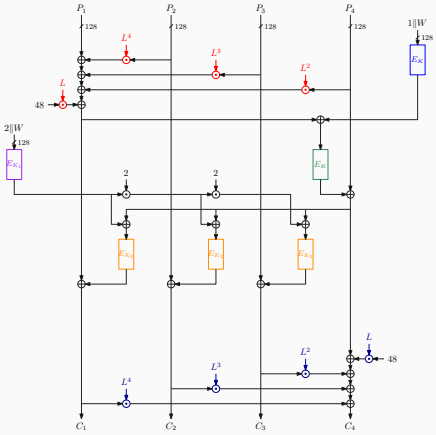
bbb-ddd-AES

- H_L instantiated using Polyval (sibling of GHASH)
- F_K instantiated as variant of CENC: tweak used to randomize inputs to AES_K

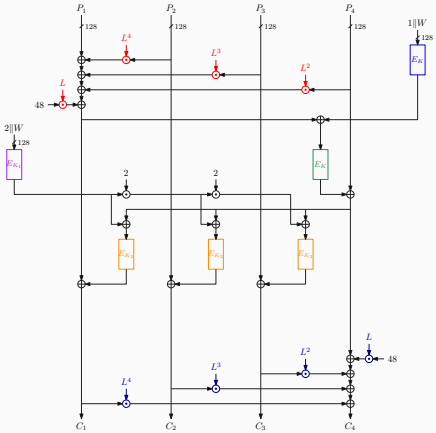
Instantiations turn out to be very competitive and well parallelizable

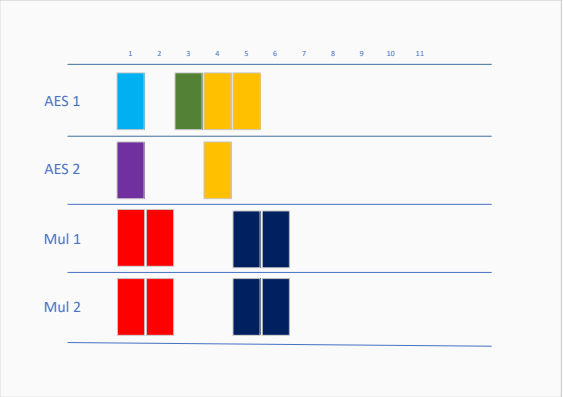
Efficiency

Implementation Design of $ddd\text{-}AES$ (512-Bit Message)

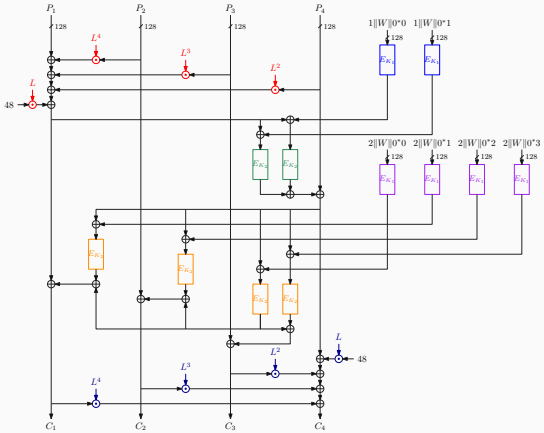


Implementation Design of $ddd\text{-}AES$ (512-Bit Message)

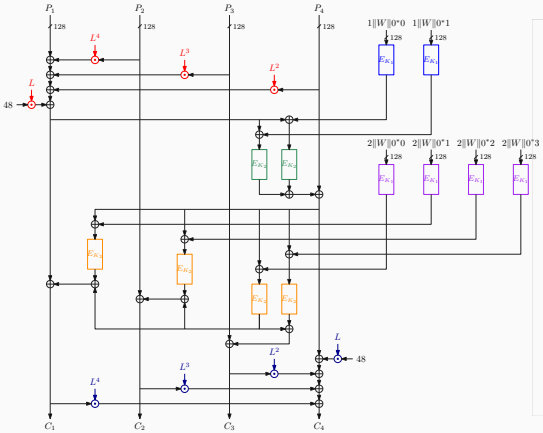


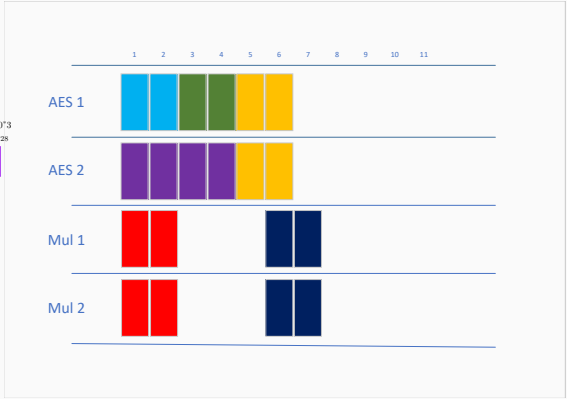


Implementation Design of $bbb\text{-}ddd\text{-}AES$ (512-Bit Message)



Implementation Design of $bbb\text{-}ddd\text{-}AES$ (512-Bit Message)





Benchmarks

- *ddd-AES* and *bbb-ddd-AES* on an Intel® Core™ i7-10610U
- C implementation using AES-NI and PCLMULQDQ

Message length (bytes)	32	48	64	96	128	256	512	1024	2048
ddd-AES x1	6	4.3	3.4	2.8	2.5	2.3	2.2	2.1	2.1
ddd-AES x2	6	3.9	3.2	2.5	2.0	1.7	1.5	1.3	1.3
ddd-AES x3	9	4.6	3.1	2.5	2.1	1.4	1.2	1.1	1.0
ddd-AES x4	7	4.3	3.5	2.6	2.3	1.6	1.3	1.1	1.0
bbb-ddd-AES x1	8	5.0	4.0	3.2	2.9	2.6	2.5	2.5	2.5
bbb-ddd-AES x2	9	5.1	3.9	3.0	2.6	1.9	1.6	1.4	1.3
bbb-ddd-AES x3	8	5.2	3.8	3.0	2.5	1.7	1.4	1.2	1.1
bbb-ddd-AES x4	8	5.0	4.1	3.0	2.8	1.9	1.4	1.2	1.1

Benchmarks

- *ddd-AES* and *bbb-ddd-AES* on an Intel® Core™ i7-10610U
- C implementation using AES-NI and PCLMULQDQ

Message length (bytes)	32	48	64	96	128	256	512	1024	2048
ddd-AES x1	6	4.3	3.4	2.8	2.5	2.3	2.2	2.1	2.1
ddd-AES x2	6	3.9	3.2	2.5	2.0	1.7	1.5	1.3	1.3
ddd-AES x3	9	4.6	3.1	2.5	2.1	1.4	1.2	1.1	1.0
ddd-AES x4	7	4.3	3.5	2.6	2.3	1.6	1.3	1.1	1.0
bbb-ddd-AES x1	8	5.0	4.0	3.2	2.9	2.6	2.5	2.5	2.5
bbb-ddd-AES x2	9	5.1	3.9	3.0	2.6	1.9	1.6	1.4	1.3
bbb-ddd-AES x3	8	5.2	3.8	3.0	2.5	1.7	1.4	1.2	1.1
bbb-ddd-AES x4	8	5.0	4.1	3.0	2.8	1.9	1.4	1.2	1.1

- For comparison, CBC encryption takes ≈ 1.4 cpb for 2048 byte messages

Conclusion

Conclusion

Instances of Docked Double Decker

- $ddd\text{-}AES$, $ddd\text{-}AES^{+}$, and $bbb\text{-}ddd\text{-}AES$
- Schemes come with security reduction to AES
- We also introduced authenticated encryption mode aaa for TWBCs
- Paper at `https://eprint.iacr.org/2024/084`

Conclusion

Instances of Docked Double Decker

- $ddd\text{-}AES$, $ddd\text{-}AES^{+}$, and $bbb\text{-}ddd\text{-}AES$
- Schemes come with security reduction to AES
- We also introduced authenticated encryption mode aaa for TWBCs
- Paper at `https://eprint.iacr.org/2024/084`

Future Research

- Turning proposal to context committing ciphers (ccc)
- $\widetilde{XORP}$ is a tweakable blockcipher based PRF used in $bbb\text{-}ddd\text{-}AES$
- Only proven $2n/3$-bit secure under limited tweak-reuse $\rightarrow$ tightness?

Conclusion

Instances of Docked Double Decker

- $ddd\text{-}AES$, $ddd\text{-}AES^{+}$, and $bbb\text{-}ddd\text{-}AES$
- Schemes come with security reduction to AES
- We also introduced authenticated encryption mode aaa for TWBCs
- Paper at `https://eprint.iacr.org/2024/084`

Future Research

- Turning proposal to context committing ciphers (ccc)
- $\widetilde{XORP}$ is a tweakable blockcipher based PRF used in $bbb\text{-}ddd\text{-}AES$
- Only proven $2n/3$-bit secure under limited tweak-reuse $\rightarrow$ tightness?

Thank you for your attention!

References i

Joan Daemen and Vincent Rijmen.
The Design of Rijndael: AES - The Advanced Encryption Standard.
Information Security and Cryptography. Springer, 2002.

Joan Daemen and Vincent Rijmen.
The Design of Rijndael - The Advanced Encryption Standard (AES), Second Edition.
Information Security and Cryptography. Springer, 2020.

Aldo Gunsing, Joan Daemen, and Bart Mennink.
Deck-Based Wide Block Cipher Modes and an Exposition of the Blinded Keyed Hashing Model.
IACR Trans. Symmetric Cryptol., 2019(4):1–22, 2019.


Shay Gueron, Adam Langley, and Yehuda Lindell.
AES-GCM-SIV: Specification and Analysis.
Cryptology ePrint Archive, Report 2017/168, 2017.
http://eprint.iacr.org/2017/168.


Viet Tung Hoang, Ted Krovetz, and Phillip Rogaway.
Robust Authenticated-Encryption AEZ and the Problem That It Solves.
In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 15–44. Springer, 2015.

Tetsu Iwata.

New Blockcipher Modes of Operation with Beyond the Birthday Bound Security.

In Matthew J. B. Robshaw, editor, *Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers*, volume 4047 of *Lecture Notes in Computer Science*, pages 310–327. Springer, 2006.


David A. McGrew and John Viega.
The Security and Performance of the Galois/Counter Mode (GCM) of Operation.
In Anne Canteaut and Kapalee Viswanathan, editors, *Progress in Cryptology - INDOCRYPT 2004, 5th International Conference on Cryptology in India, Chennai, India, December 20-22, 2004, Proceedings*, volume 3348 of *Lecture Notes in Computer Science*, pages 343–355. Springer, 2004.


Phillip Rogaway.
Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC.
In Pil Joong Lee, editor, *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings*, volume 3329 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2004.

Efficient Instantiation

Polyval [GLL17]

- Operates on finite field $GF(2^{128})[x]/(x^{128} + x^{127} + x^{126} + x^{121} + 1)$
- Defined as follows, for a padded message $(I_1, I_2, \ldots, I_s)$:

$$Polyval_L(I_1, I_2, \ldots, I_s) = \sum_{i=1}^{s} \left(L^{s-i+1} \cdot I_i \cdot x^{-128 \cdot (s-i+1)} \right)$$

- We use zero-padding with length encoding

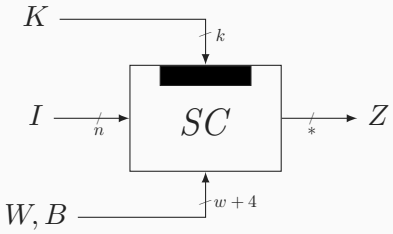
Polyval [GLL17]

- Operates on finite field $GF(2^{128})[x]/(x^{128} + x^{127} + x^{126} + x^{121} + 1)$
- Defined as follows, for a padded message $(I_1, I_2, \ldots, I_s)$:

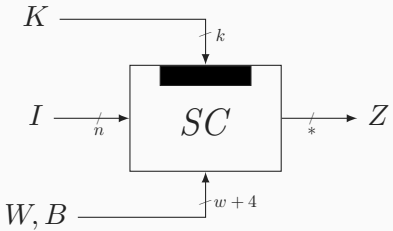
$$Polyval_L(I_1, I_2, \ldots, I_s) = \sum_{i=1}^{s} \left(L^{s-i+1} \cdot I_i \cdot x^{-128 \cdot (s-i+1)} \right)$$

- We use zero-padding with length encoding
- $Polyval$ is ϵ-XOR-universal with $\epsilon = m_{\max}/2^{128}$ [GLL17]

Recall Goal

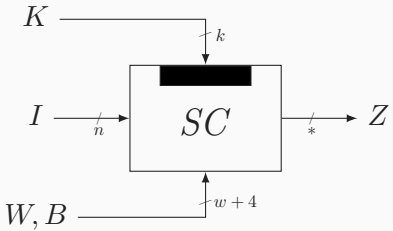


Recall Goal



- Construction should be built on top of AES

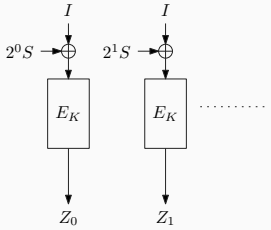
Recall Goal



- Construction should be built on top of AES
- We give one construction with birthday bound security
  one construction with beyond birthday bound security

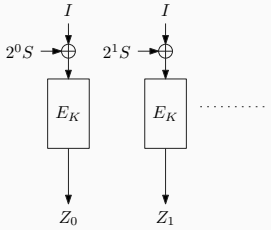
XE-style [Rog04] Tweakable Blockcipher in Counter Mode

- Let $S = E_K(B\|W)$



XE-style [Rog04] Tweakable Blockcipher in Counter Mode

- Let $S = E_K(B\|W)$



- Stream cipher (and thus $ddd\text{-}AES$) is $2^{n/2}$ PRF-secure

Bonus: Extension $ddd\text{-}AES^+$ to Accommodate Variable-Length Tweaks

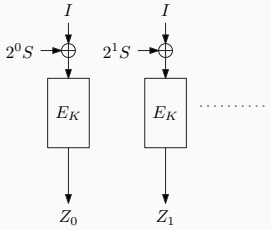
- $ddd\text{-}AES$ almost seamlessly fits NIST's accordion idea
- Only thing missing: variable-length tweaks

Bonus: Extension $ddd\text{-}AES^+$ to Accommodate Variable-Length Tweaks

- $ddd\text{-}AES$ almost seamlessly fits NIST's accordion idea
- Only thing missing: variable-length tweaks

XE^+-style [Rog04] Tweakable Blockcipher in Counter Mode

- Pad B, W into $(W_0, W_1, \ldots, W_{l-1} \| B' \| 0^*)$ with $B' = B \oplus 1000$
- Let $S = E_K(W_0 \| 0) \oplus E_K(W_1 \| 1) \oplus \cdots \oplus E_K(W_{l-1} \| B' \| 0^* \| (l-1))$

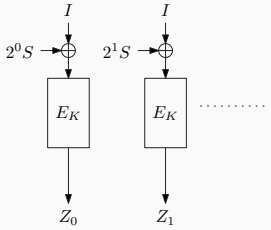


Bonus: Extension $ddd\text{-}AES^+$ to Accommodate Variable-Length Tweaks

- $ddd\text{-}AES$ almost seamlessly fits NIST's accordion idea
- Only thing missing: variable-length tweaks

XE^+-style [Rog04] Tweakable Blockcipher in Counter Mode

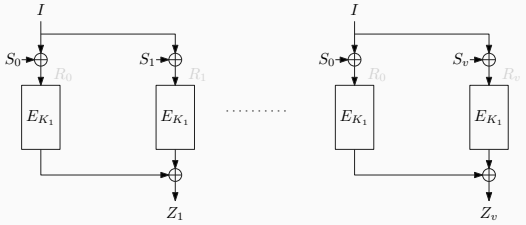
- Pad B, W into $(W_0, W_1, \ldots, W_{l-1} \| B' \| 0^*)$ with $B' = B \oplus 1000$
- Let $S = E_K(W_0 \| 0) \oplus E_K(W_1 \| 1) \oplus \cdots \oplus E_K(W_{l-1} \| B' \| 0^* \| (l-1))$



- Stream cipher (and thus $ddd\text{-}AES^+$) is $2^{n/2}$ PRF-secure

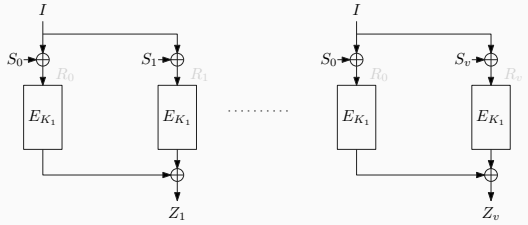
$\widetilde{XORP}$ PRF in Counter Mode

- $\widetilde{XORP}$: $XORP$ as used in CENC [Iwa06], and extended to include tweak
  - Introduction is new and comes with separate security proof
  - Let $S_j = E_{K_2}(B \| W \| c \| j)$



$\widetilde{XORP}$ PRF in Counter Mode

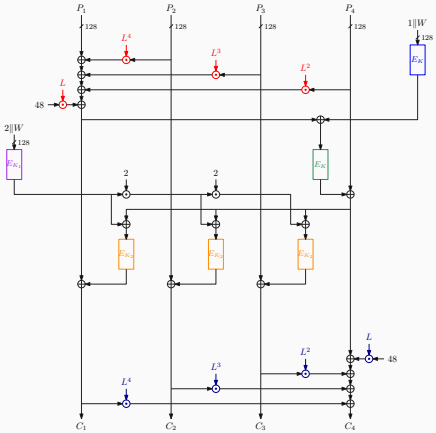
- $\widetilde{XORP}$: $XORP$ as used in CENC [Iwa06], and extended to include tweak
  - Introduction is new and comes with separate security proof
  - Let $S_j = E_{K_2}(B \| W \| c \| j)$

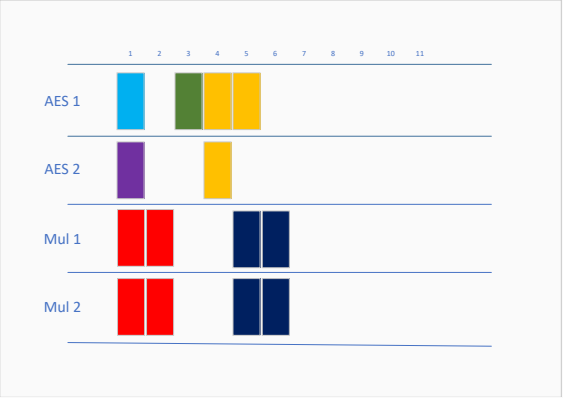


- Corresponding stream cipher runs $\widetilde{XORP}$ in counter mode
- Stream cipher (and thus $bbb\text{-}ddd\text{-}AES$) is $2^{2n/3}$ PRF-secure when tweaks are not used too often

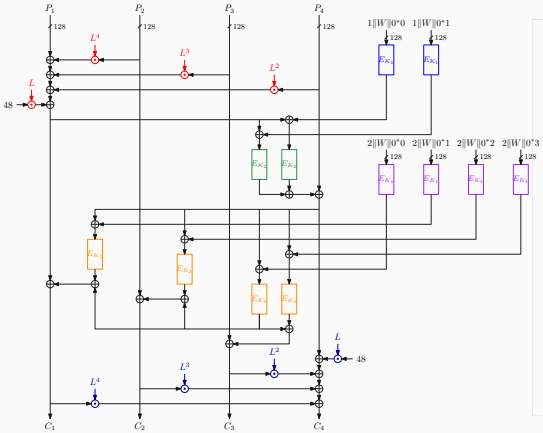
Efficiency

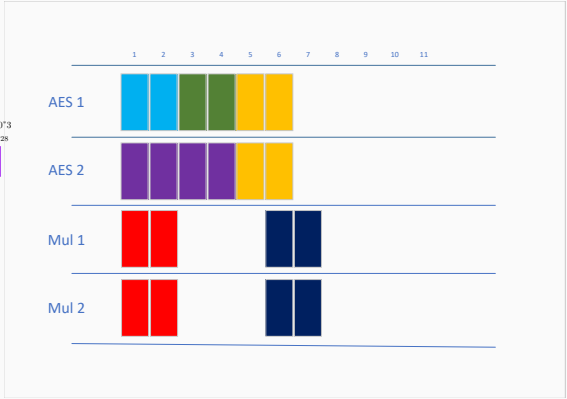
Implementation Design of $ddd\text{-}AES$ (512-Bit Message)



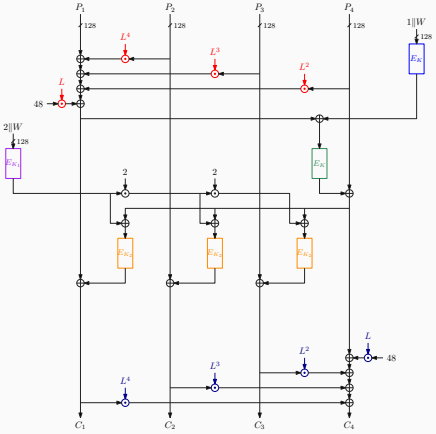


Implementation Design of $bbb\text{-}ddd\text{-}AES$ (512-Bit Message)

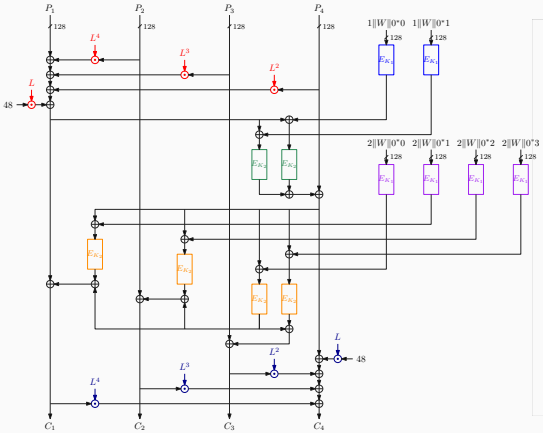




Implementation Design of $ddd\text{-}AES$ (1024-Bit Message)

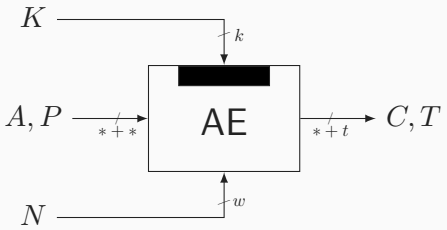


Implementation Design of $bbb\text{-}ddd\text{-}AES$ (1024-Bit Message)



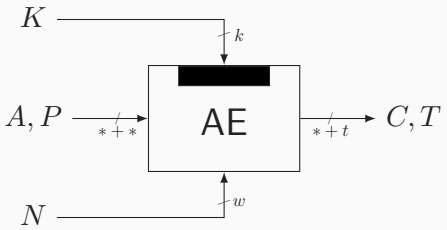
Application to Authenticated Encryption

Authenticated Encryption



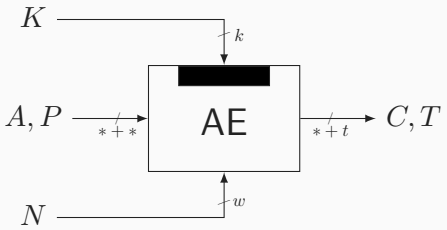
- Using key K:
  - Plaintext P is encrypted in ciphertext C
  - Associated data A and plaintext P are authenticated using T

Authenticated Encryption



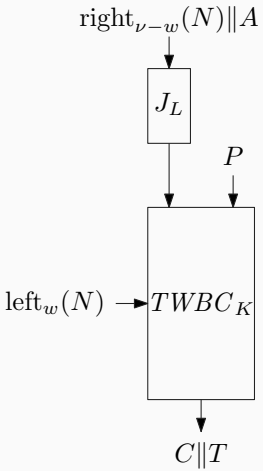
- Using key K:
  - Plaintext P is encrypted in ciphertext C
  - Associated data A and plaintext P are authenticated using T
- Nonce N randomizes the scheme

Authenticated Encryption



- Using key K:
  - Plaintext P is encrypted in ciphertext C
  - Associated data A and plaintext P are authenticated using T
- Nonce N randomizes the scheme
- Decryption outputs message if and only if tag is correct

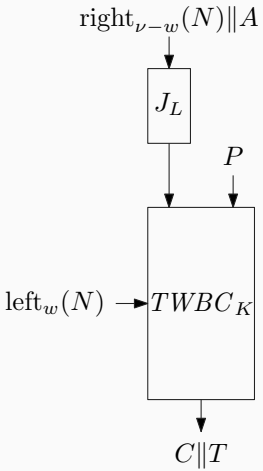
Advanced Authenticated Encryption with Associated Data (aaa)



Building Blocks

- $TWBC_K$: tweakable wide blockcipher
- J_L: universal hash

Advanced Authenticated Encryption with Associated Data (aaa)

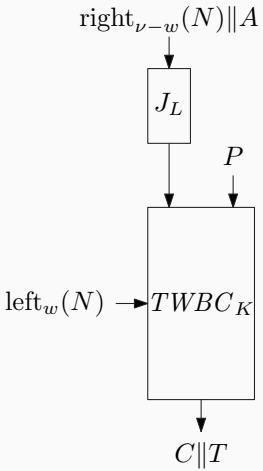


Building Blocks

- $TWBC_K$: tweakable wide blockcipher
- J_L: universal hash

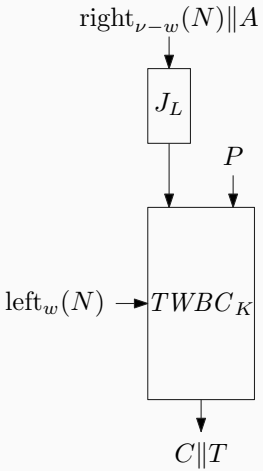
Rationale

- N partially entered into tweak
- Rest of N and A hashed into τ-bit string



Nonce-Respecting Setting

- $\mathrm{left}_w(N)$ unique for each *encryption* query
- Security analysis relies on fact that tweak to $TWBC_K$ is always new



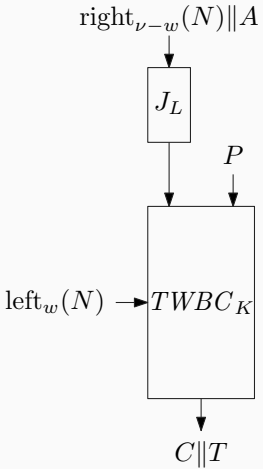
Nonce-Respecting Setting

- $\mathrm{left}_w(N)$ unique for each *encryption* query
- Security analysis relies on fact that tweak to $TWBC_K$ is always new

Random Nonce Setting

- N is random for each *encryption* query
- Security analysis relies on multicollision bound on the left w bits of the nonce

Advanced Authenticated Encryption with Associated Data (aaa): Security



Nonce-Respecting Setting

- $\mathrm{left}_w(N)$ unique for each *encryption* query
- Security analysis relies on fact that tweak to $TWBC_K$ is always new

Random Nonce Setting

- N is random for each *encryption* query
- Security analysis relies on multicollision bound on the left w bits of the nonce

Nonce-Misusing Setting

- Birthday bound security retained