SILC: SImple Lightweight CFB

Tetsu Iwata, Nagoya University
Kazuhiko Minematsu, NEC Corporation
Jian Guo, Nanyang Technological University
Sumio Morioka, NEC Europe Ltd.
Eita Kobayashi, NEC Corporation

DIAC 2014
August 23, 2014, Santa Barbara, USA

Outline

- Authenticated Encryption with Associated Data (AEAD)
- SILC, SImple Lightweight CFB, pronounced as "silk"

http://pixabay.com/en/silk-yarn-thread-spool-thread-196539/

SILC Design Goal

- Provably secure AEAD that is based on a blockcipher
  - Standard security notions for privacy and authenticity
- To improve previous schemes, CCM, EAX, and EAX-prime
  - optimizing the design to achieve a small gate size on HW implementations
- HW oriented version of CLOC [IMGM14]
  - CLOC is for embedded SW implementations

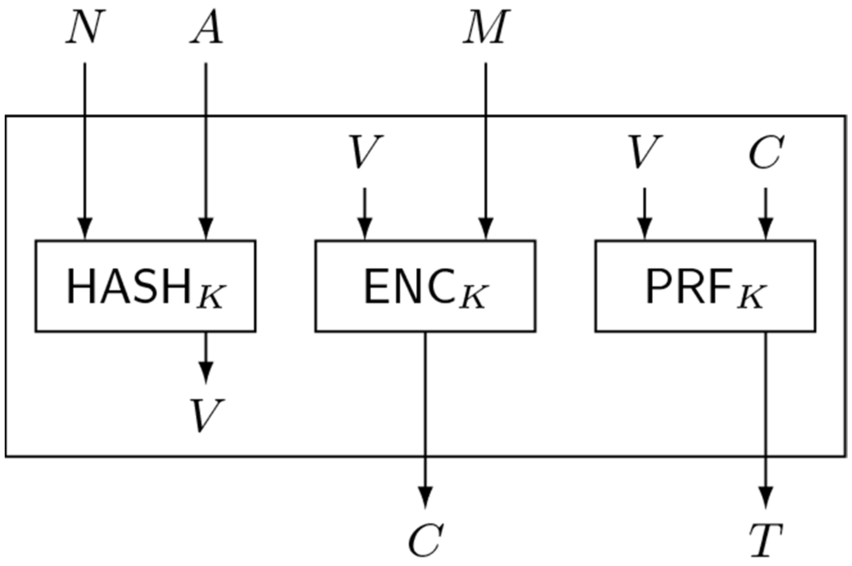
[IMGM14] Iwata, Minematsu, Guo, Morioka, CLOC: Compact Low-Overhead CFB, Submission to the CAESAR competition

Design Strategy

- CLOC optimizes the number of blockcipher calls by making various cases
  - if the input is empty, a multiple of block size, or otherwise
  - this contributes to the efficiency for short input, and well suits for embedded SW implementations
  - requires non-negligible number of logic gates
- SILC avoids making cases
  - at the cost of the constant number of increase of blockcipher calls
  - data blocks are processed consistently
  - reduces the logic gates needed to implement the cases

SILC Overview

- SILC is built upon CLOC
- It follows the Encrypt-then-PRF paradigm
- HASH, PRF: variants of CBC MAC
- ENC: a variant of CFB



Parameters

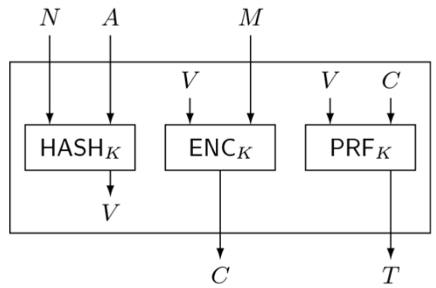
- E_K: blockcipher with an n-bit block
- l_N: nonce length in bits

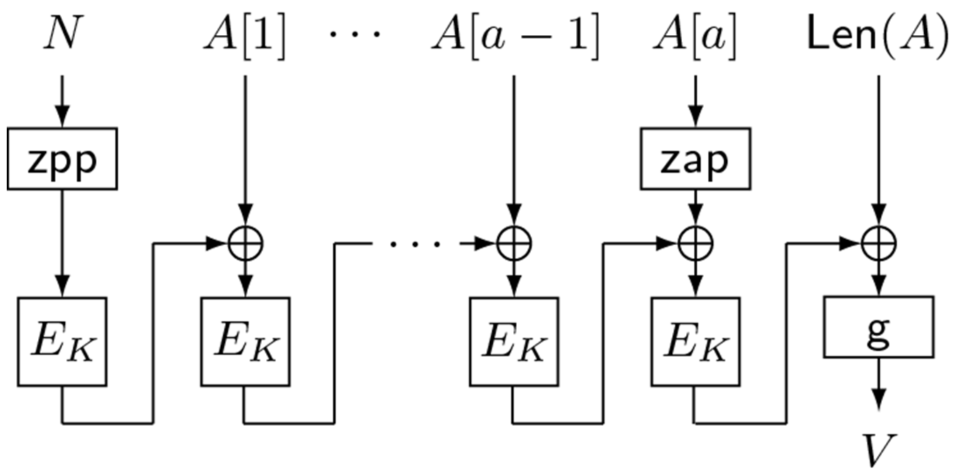
  $1 \leqq l_N \leqq n-1$
- tau: tag length in bits

  $1 \leqq tau \leqq n$

V <- $HASH_K(N,A)$

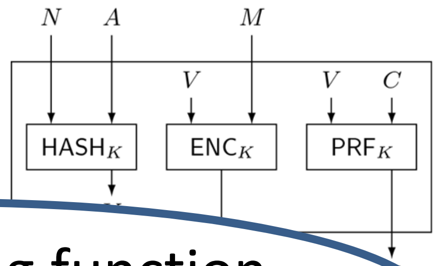
- variant of CBC MAC
- N: nonce, fixed length, $1 \leqq |N| \leqq n-1$
- A: associated data, at most $2^{n/2}$-1 bytes



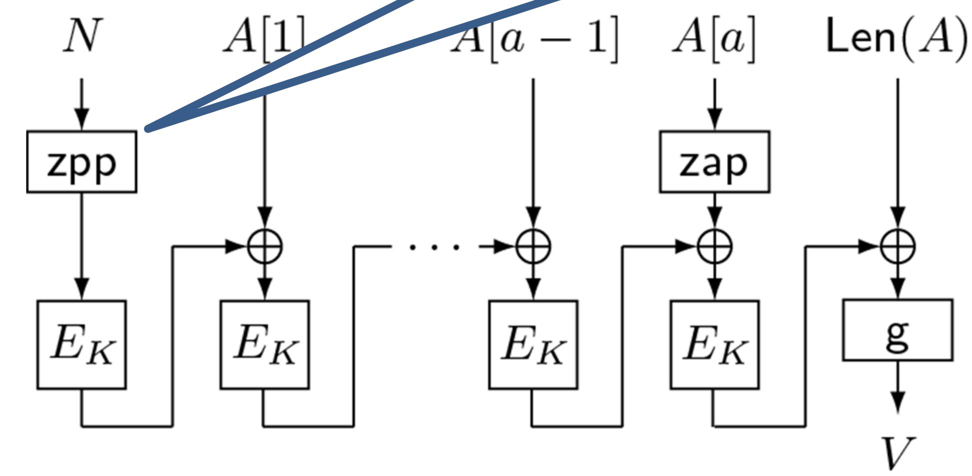


V <- $HASH_K$(N,A)

- variant of CBC MAC
- N: nonce, fixed length, 1 ≦ |N| ≦ n−1
- A: associated data, at most $2^{n/2}$-1



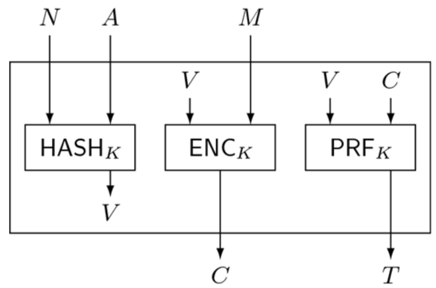
zero prepending function
zpp(N) = 0…0 || N

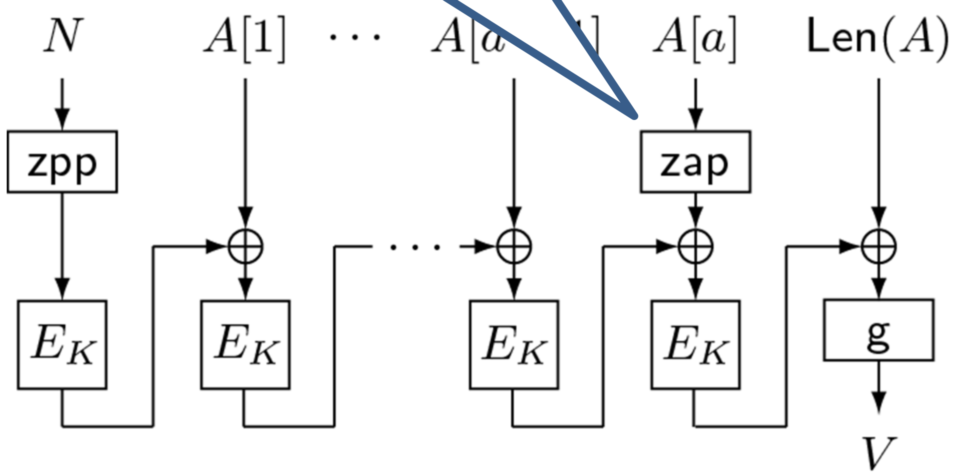


V <- HASH_K(N,A)

- variant of CBC MAC
- N: n
-

zero appending function
zap(X) = X || 0…0
(possibly none)

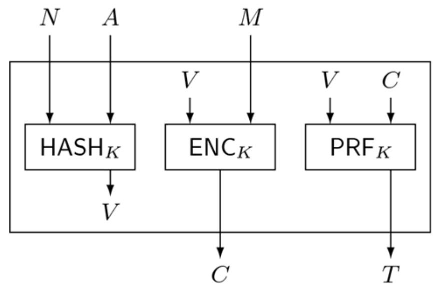


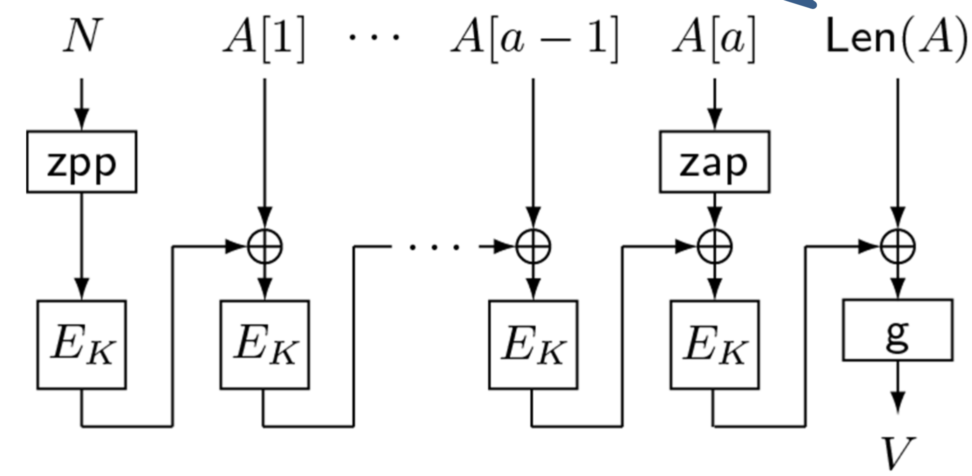


V <- HASH_K(N,A)

- variant of CBC MAC
- N: no
-

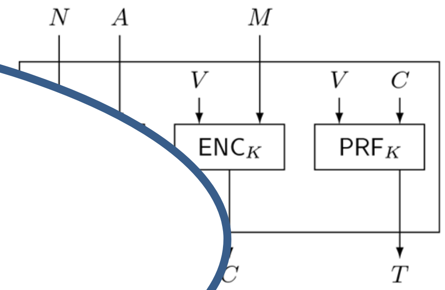
Len(A) = length of A in bytes



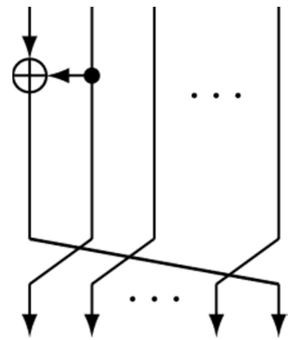


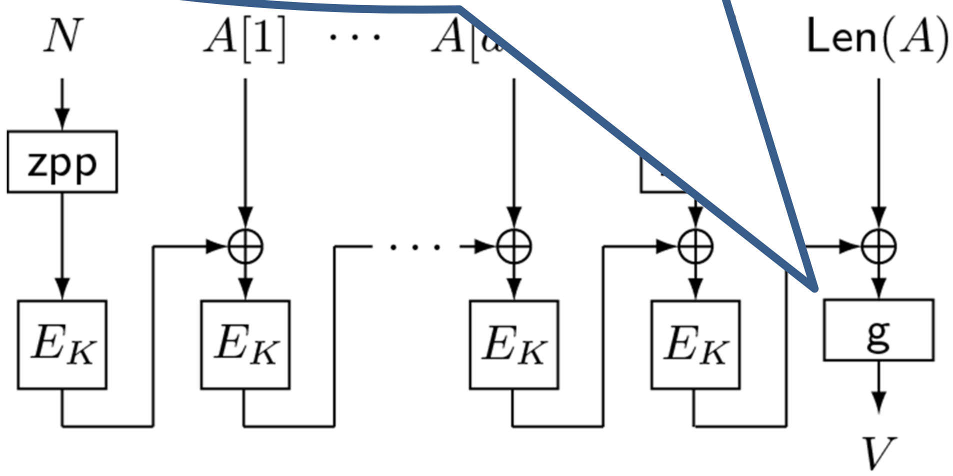
V <- HASH_K(N,A)

- variant of C…
- N:
-



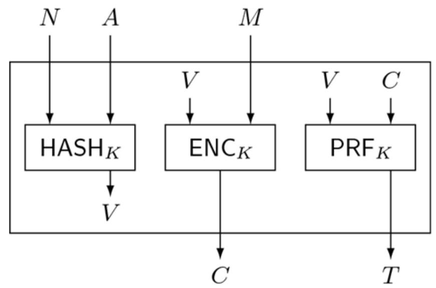
tweak function
broken into bytes

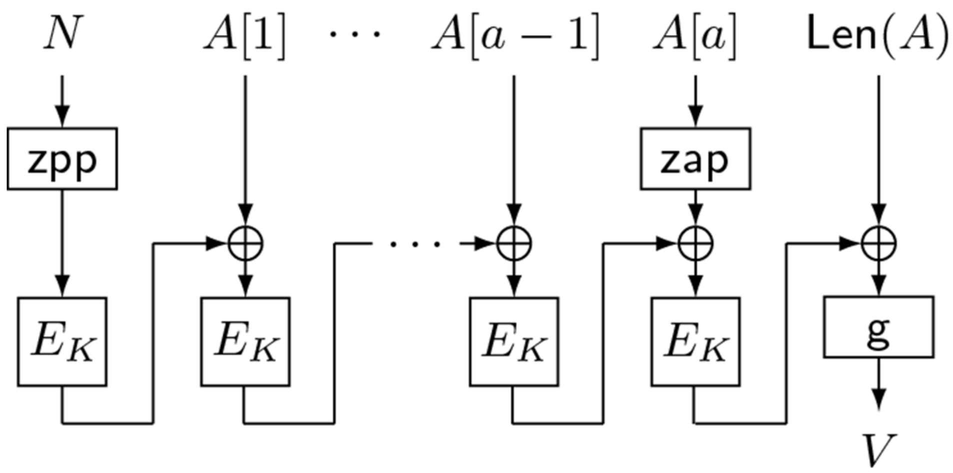




V <- $HASH_K(N,A)$

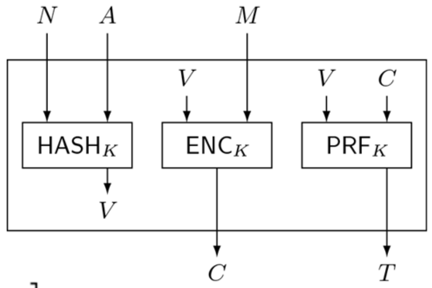
- variant of CBC MAC
- N: nonce, fixed length, $1 \leqq |N| \leqq n-1$
- A: associated data, at most $2^{n/2}$-1 bytes

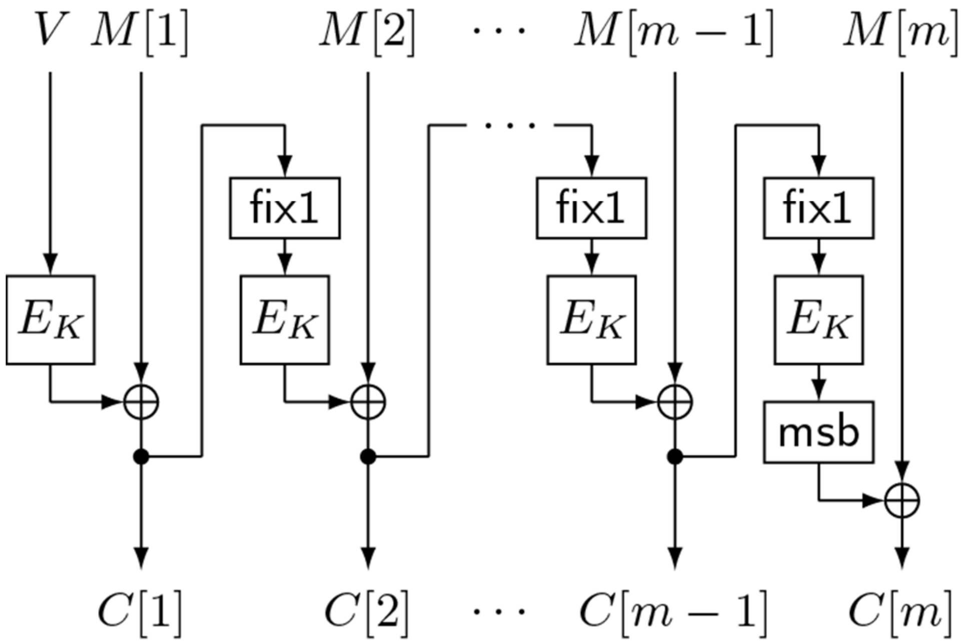




C <- ENC$_K$(V,M)

- variant of CFB mode
- M: plaintext, at most $2^{n/2}$-1 bytes

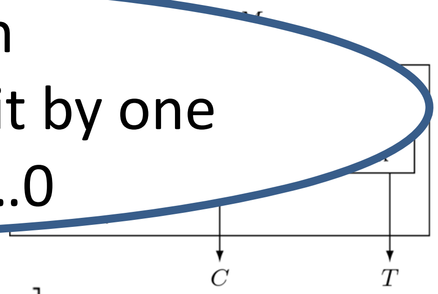


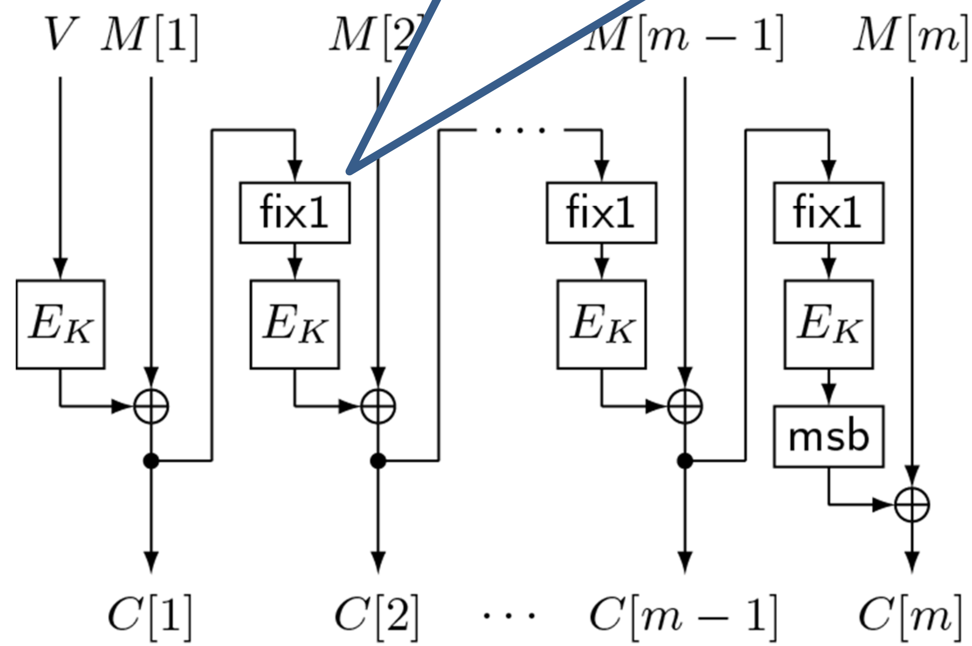


C <- ENC_K(V,M)

- variant of CFB m
- M: plaintext, a

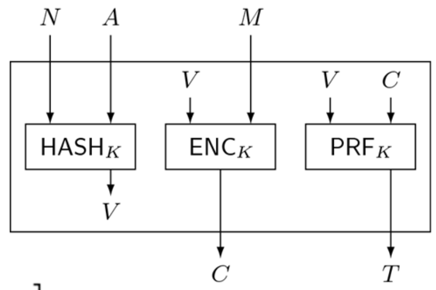
bit fixing function
fix the most significant bit by one
fix1(X) = X OR 10…0

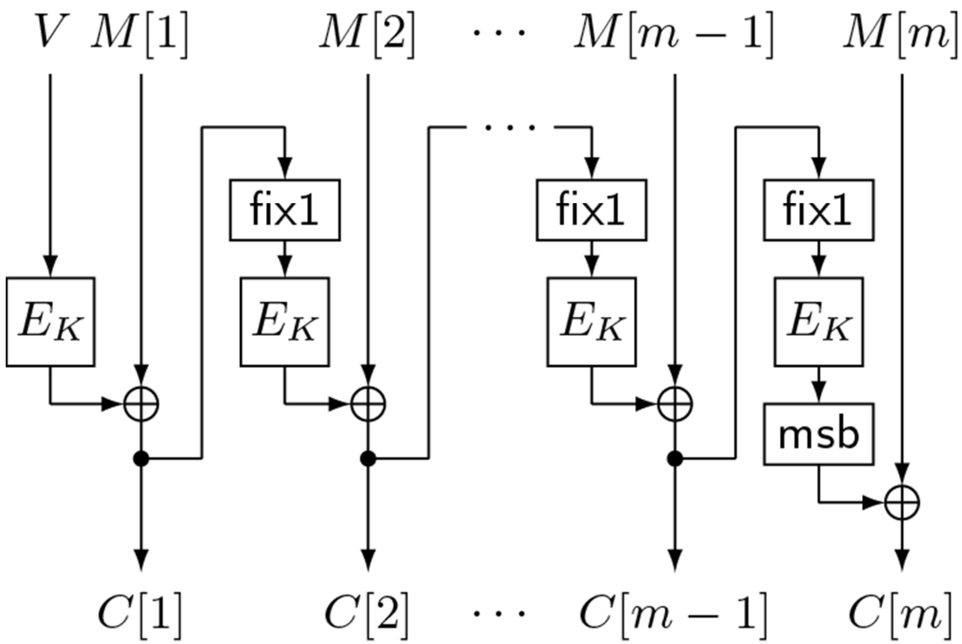




C <- $ENC_K(V,M)$

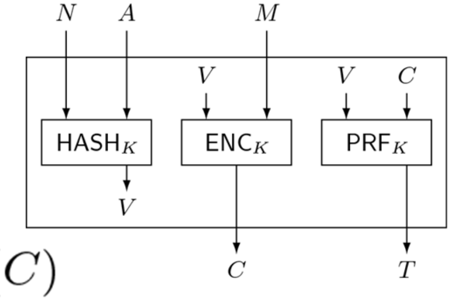
- variant of CFB mode
- M: plaintext, at most $2^{n/2}$-1 bytes

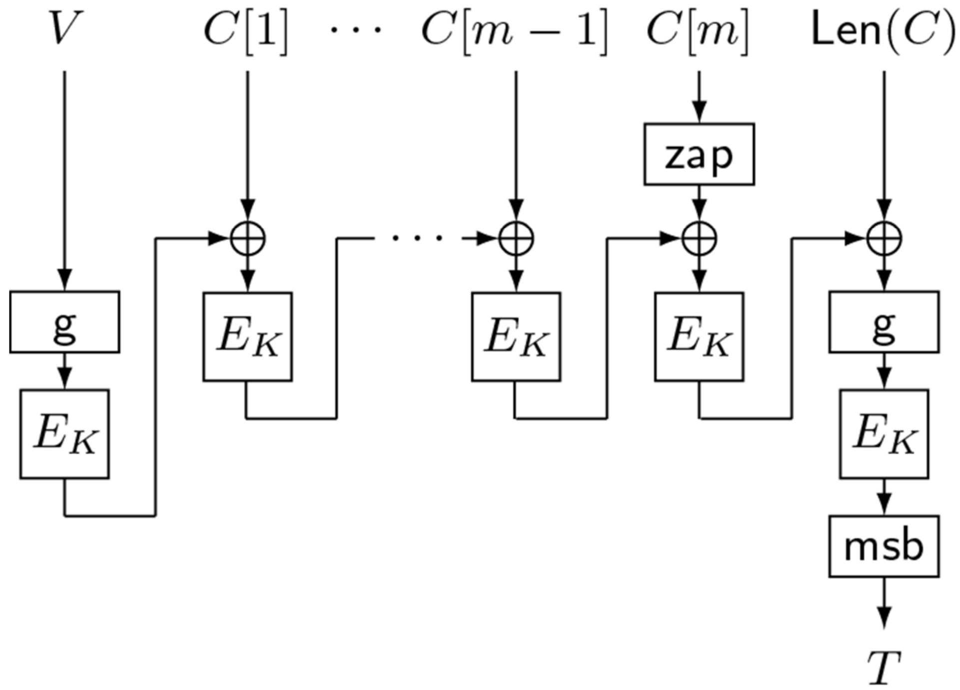




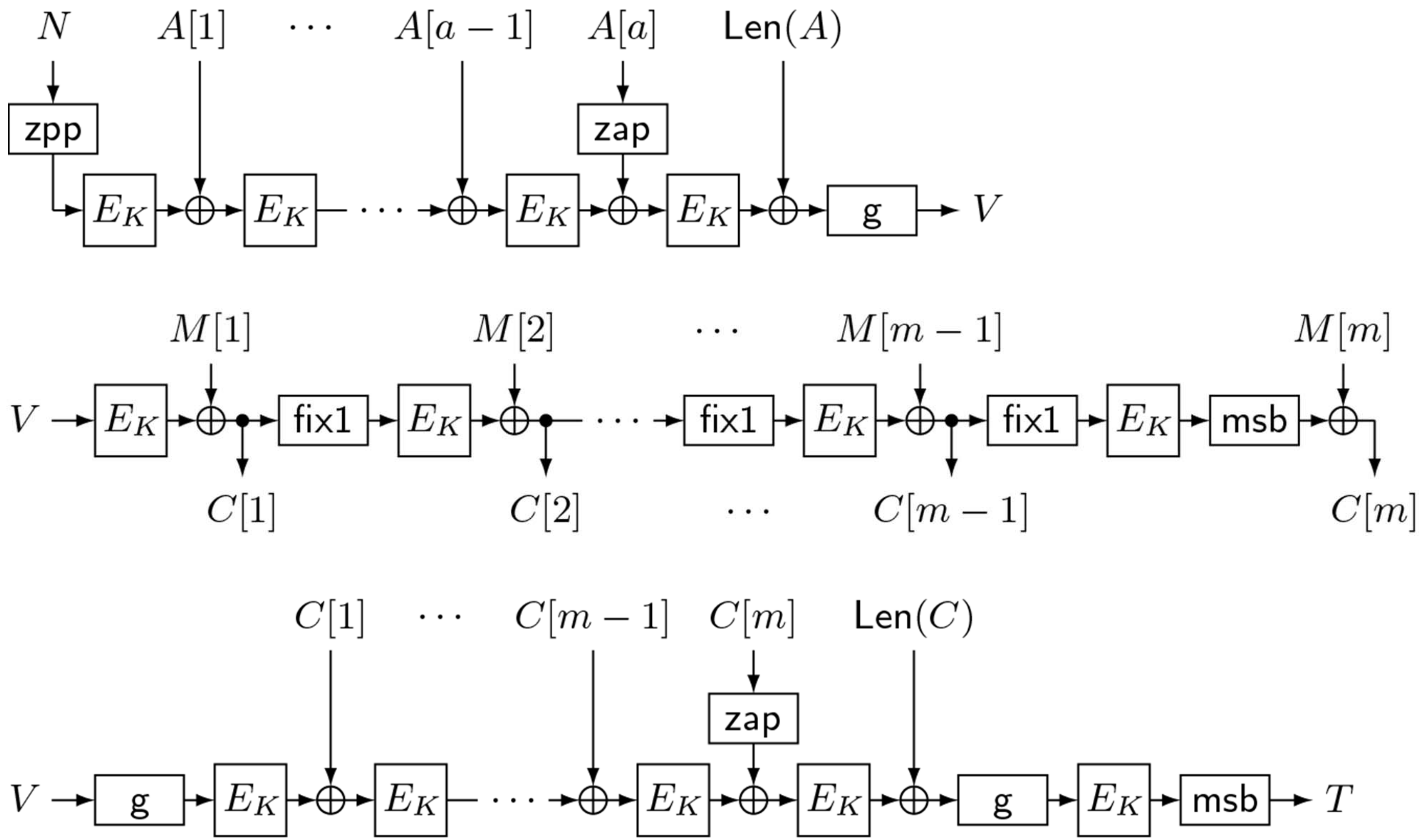
T <- $PRF_K(V,C)$

- variant of CBC MAC





Works with Two State Blocks



SILC Properties

- Nonce-based AEAD
- uses only the encryption of the blockcipher both for encryption and decryption
- It makes $|N|_n + |A|_n + 2|M|_n + 2$ blockcipher calls for a nonce N, associated data A, and a plaintext M
  - where $|X|_n$ is the length of X in n-bit blocks
  - $1 \leqq |N| \leqq n-1$, so $|N|_n = 1$
  - blockcipher key scheduling can be precomputed
  - No precomputation beyond that (blockcipher calls, generation of key dependent tables, . . .) is needed

Limitations

- Static associated data cannot be handled efficiently
  - nonce is processed before associated data
- For long plaintexts, it needs 2 blockcipher calls per one block
- HASH, ENC, and PRF are all sequential
  - blockcipher calls in ENC and PRF are parallelizable

Security

- Privacy:

  Indistinguishability of ciphertexts from random bits against nonce-respecting adversaries in a chosen plaintext attack setting

- $\mathbf{Adv}^{\mathrm{priv}}_{\mathrm{SILC}[E,\ell_N,\tau]}(\mathcal{A}) \stackrel{\mathrm{def}}{=} \Pr\left[\mathcal{A}^{\mathrm{SILC}\text{-}\mathcal{E}_K(\cdot,\cdot,\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot,\cdot,\cdot)} \Rightarrow 1\right]$

- $\mathbf{Adv}^{\mathrm{priv}}_{\mathrm{SILC}[\mathrm{Perm}(n),\ell_N,\tau]}(\mathcal{A}) \leq \dfrac{5\sigma^2_{\mathrm{priv}}}{2^n}$, where $\sigma_{\mathrm{priv}} = 3q + \sigma_A + 2\sigma_M$

Security

- Authenticity:

  Unforgeability against **nonce-reusing** adversaries in a chosen ciphertext attack setting

- $\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{SILC}[E,\ell_N,\tau]}(\mathcal{A}) \stackrel{\mathrm{def}}{=} \Pr\left[\mathcal{A}^{\mathrm{SILC}\text{-}\mathcal{E}_K(\cdot,\cdot,\cdot),\mathrm{SILC}\text{-}\mathcal{D}_K(\cdot,\cdot,\cdot,\cdot)} \text{ forges}\right]$

- $\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{SILC}[\mathrm{Perm}(n),\ell_N,\tau]}(\mathcal{A}) \le \dfrac{5\sigma^2_{\mathrm{auth}}}{2^n} + \dfrac{q'}{2^\tau},$

  where $\sigma_{\mathrm{auth}} = 3q + \sigma_A + 2\sigma_M + 3q' + \sigma_{A'} + \sigma_{C'}$

Security

- Authenticity:

  Unforgeability against **nonce-reusing** adversaries in a chosen ciphertext attack setting

- $\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{SILC}[E,\ell_N,\tau]}(\mathcal{A}) \stackrel{\mathrm{def}}{=} \Pr\left[\mathcal{A}^{\mathrm{SILC}\text{-}\mathcal{E}_K(\cdot,\cdot,\cdot),\mathrm{SILC}\text{-}\mathcal{D}_K(\cdot,\cdot,\cdot,\cdot)} \text{ forges}\right]$

- $\mathbf{Adv}^{\mathrm{auth}}_{\mathrm{SILC}[\mathrm{Perm}(n),\ell_N,\tau]}(\mathcal{A}) \leq \frac{5\sigma^2_{\mathrm{auth}}}{2^n} + \frac{q'}{2^\tau},$

  where $\sigma_{\mathrm{auth}} = 3q + \sigma_A + 2\sigma_M + 3q' + \sigma_{A'} + \sigma_{C'}$

- Standard birthday bounds, proofs are similar to those of CLOC

Recommended Parameter Sets

- E_K: blockcipher with an n-bit block
  - n: 64 or 128
  - AES-128 for n = 128, and PRESENT-80 or LED-80 for n = 64
- l_N: nonce length in bits
  - 96 or 64 for n = 128, and 48 for n = 64
- tau: tag length in bits
  - 64 for n = 128, and 32 for n = 64

Recommended Parameter Sets

- E_K: blockcipher with an n-bit block
  - n: 64 or 128
  - AES-128 for n = 128, and PRESENT-80 or LED-80 for n = 64
- l_N: nonce length in bits
  - 96 or 64 for n = 128, and 48 for n = 64
- tau: tag length in bits
  - 64 for n = 128, and 32 for n = 64

- 64-bit blockciphers are not for general purpose applications
  - for applications that can ensure the total amount of data processed with one key
  - low data transmission rate, limited battery lifetime

HW Implementation

- We evaluated AES-SILC for ASIC using a 90 nm standard cell library
- HW reference implementation AES-SILC
  - to see the basic performance
- Compared it with AES-CLOC, AES-OTR, and AES-EAX
  - Unit = Gate Equivalent (GE)
  - AES is round-based, where S-box uses the composite-field expression
  - single AES core

HW Implementation

- Scenario 1
  - Frequency is fixed to 100 MHz

	AES	SILC	CLOC	OTR	EAX
Gates (GE)	10207.75	15675.5	17137.75	21862.5	28662.25
Ratio (AES)	1	1.54	1.68	2.14	2.81
Throughput (Mbit/sec)	1163.63	764.12	685.71	1134.18	794.48

- SILC is the smallest (x 1.54 of AES size)
- no significant change if the freq. ~= 20 MHz
- Throughput is an estimation

HW Implementation

- Scenario 2
  - The same RTL (Register Transfer Level) as Scenario 1
  - find the maximum frequency

	SILC	CLOC	OTR	EAX
Max freq. (MHz)	344.8	312.5	333.3	277.8
Gates (GE)	23135	25287.25	29080.75	35305
Ratio (AES)	1.57	2.01	2.07	3.16
Throughput (Mbit/sec)	2634.88	2142.85	3780.21	2207.07

- Ratio: compared with AES of the corresponding freq.
- SILC is again the smallest (x 1.57 of AES size)
- Throughput is an estimation

SW Implementation

- Not the main focus of SILC
- General purpose CPU
  - Intel(R) Core(TM) i5-3427U CPU, 1.80GHz (Ivy Bridge)
  - with a long plaintext (more than 2^{20} blocks) and empty associated data, and with parallelism P

	AES-SILC	PRESENT-SILC	LED-SILC
Speed (cpb)	4.9	42	40
Remarks	AES-NI, P=1	bit-sliced, P=16	bit-sliced, P=32

- In AES-SILC, E_K in ENC and PRF are computed in parallel
- AES-CLOC: about 4.9 cpb (P = 1)
- serial AES-128 encryption: about 4.3 cpb

LED Reference Code

- Inconsistency in the description of LED in the submission document and the LED reference code
  - The LED reference code will be updated soon
  - The reference code of SILC remains unchanged

Conclusions

- Designed SILC and analyzed the security and the efficiency
- SILC is suitable for use within constrained HW devices

http://pixabay.com/en/silk-yarn-thread-spool-thread-196539/