Cymric: Short-tailed but Mighty

Alexandre Adomnicăi[1] **Wonseok Choi**[2] Yeongmin Lee[3]
Kazuhiko Minematsu[4] Yusuke Naito[5]

[1]Independent Researcher, Paris, France, [2]DGIST, Daegu, Korea, [3]DESILO Inc., Seoul, Korea,
[4]NEC, Kawasaki, Japan, [5]Mitsubishi Electric Corporation, Kanagawa, Japan,

Cymric?



Cymric?

Cymric cat

27 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

The **Cymric** (/ˈkɪmrɪk/ *KIM-rik*, /ˈkʌmrɪk/ *KUM-rik*) is a Canadian cat breed. Some cat registries consider the Cymric a semi-long-haired variety of the Manx breed, rather than a separate breed. Except for the length of fur, in all other respects, the two varieties are the same, and kittens of either sort may appear in the same litter. The name comes from *Cymru* (Welsh pronunciation: [ˈkəmrɨ]), the indigenous Welsh name of Wales, even though the breed is not associated with Wales. The name may have been chosen to provide a "Celtic" sounding moniker for the breed. While the breed's Manx bloodline originated from the Isle of Man, the long-haired variant is claimed to have been developed by Canada. The breed is called the **Longhair Manx** or a similar name by some registries.

History [edit]

According to the Isle of Man records, the taillessness trait of the Manx (and ultimately the Cymric) began as a mutation among the island's domestic cat population. Given the island's closed environment and small gene pool, the dominant gene that decided the cats' taillessness was easily passed from one generation to the next, along with the gene for long hair. Long-haired kittens had been born to Manx cats on the Isle of Man, but had always been discarded by

Cymric

Tortoiseshell cat

Other names	Manx Longhair, Longhair Manx, Semi-longhair Manx Variant, long-haired Manx
Origin	Canada (breeding programme), Isle of Man (Manx stock)

Cymric

- Cymric = MANX with LONGHAIR
- What is MANX?

Cymric

- Cymric = MANX with LONGHAIR
- What is MANX?

Manx?

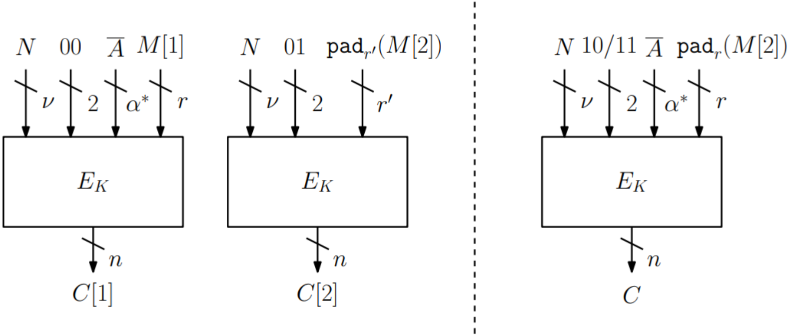


Fig. 3: Encryption of Manx2. (Left) Short message case, (Right) Tiny message case.

- Alexandre Adomnicăi, Kazuhiko Minematsu, and Junji Shikata, "Authenticated Encryption for Very Short Inputs", CT-RSA '23

Cymric vs. Manx

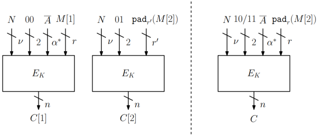


Fig. 3: Encryption of Manx2. (Left) Short message case, (Right) Tiny message case.

- Observation: Cymric is cuter.

Cymric vs. Manx

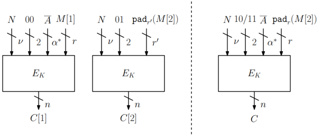


Fig. 3: Encryption of Manx2. (Left) Short message case, (Right) Tiny message case.

- Observation: Cymric is cuter.

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Cymric, More Details

- A BBB AEAD dedicated to short inputs!
- an interesting security proof!
- Achieving tightness with a matching impossibility result!
- Super fast!
- Cymric: Short-tailed but Mighty and hairy

Motivation

- Typical examples are found in low-power wireless communication because of (e.g.) limited packet length from power constraints. For example,
  - Sigfox limits packet lengths to a maximum of 12 bytes,
  - EnOcean limits packet lengths to 9 or 14 bytes, and
  - Bluetooth Low Energy (v4.0) supports payloads up to 33 bytes.
  - Electronic Product Code (EPC) specified for RFIDs has just a 12-byte payload.
  - Micro QR code can contain up to 15 bytes.
  - For healthcare applications using tiny medical sensors, Narrow-Band IoT standards work with 1 to 4-byte payloads.
  - Andreeva et al. (the Forkchipher work) present more examples.

Motivation

- Typical examples are found in low-power wireless communication because of (e.g.) limited packet length from power constraints. For example,
  - Sigfox limits packet lengths to a maximum of 12 bytes,
  - EnOcean limits packet lengths to 9 or 14 bytes, and
  - Bluetooth Low Energy (v4.0) supports payloads up to 33 bytes.
  - Electronic Product Code (EPC) specified for RFIDs has just a 12-byte payload.
  - Micro QR code can contain up to 15 bytes.
  - For healthcare applications using tiny medical sensors, Narrow-Band IoT standards work with 1 to 4-byte payloads.
  - Andreeva et al. (the Forkchipher work) present more examples.

Cymric

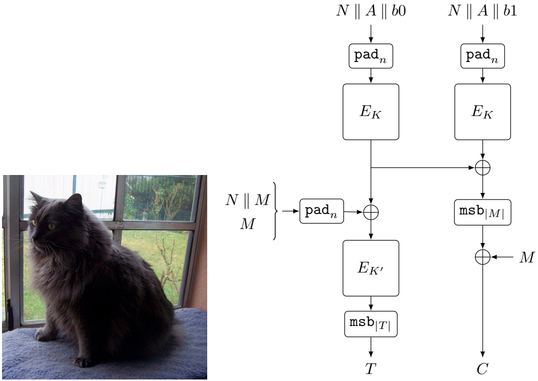


Figure 1: Left: a cat. Right: Cymric1 uses $N \| M$ for the middle XOR in the left branch whereas Cymric2 uses M. $b = 1$ iff $|N| + |M| = n$ for Cymric1 and $|M| = n$ for Cymric2.

Comparison Table

Table 1: Comparison of AE schemes based on an n-bit block cipher. MUL denotes a multiplication over $\mathrm{GF}(2^n)$. Min. calls denotes the minimum number of block cipher calls required for non-empty messages. ν and α are the predefined bit length of nonces and ADs, respectively.

Scheme	Max. message length	Primitive	Min. calls	Security	Expansion
OCB	any	SPRP	4	$n/2$	No
GCM	any	PRP, MUL	$3^{\dagger 1}$	$n/2$	No
CCM	any	PRP	4	$n/2$	No
XOCB	any	SPRP	9	$2n/3$	No
EtE	$n-\nu-\alpha$	SPRP	1	$n/2$	Yes
Manx2	n	SPRP	2	$n/2^{\dagger 2}$	Yes
Cymric1	$n-\nu$	PRP	3	n	No
Cymric2	n	PRP	3	$2n/3$	No

$\dagger 1$: additional $\mathrm{GF}(2^n)$ multiplications (two when $\nu = 96$ and four otherwise)
$\dagger 2$: optimal value achieved when nonce is $n/2$ bits

Intuition

- Amalgamating EWCDM nonce-based MAC and SoP PRF
  - both providing BBB security
- EWCDM nonce-based MAC has been analyzed via improved Mirror theory
  - Wonseok Choi, Jooyoung Lee, Yeongmin Lee, "Toward Full n-bit Security and Nonce Misuse Resistance of Block Cipher-based MACs", ASIACRYPT '24
- (Generically) composing EWCDM and SoP could be used, but
  - More key materials, more BC calls...

Theorems

- If $q_e \leq \frac{2^n}{48n^2}$, $q_d \leq 2^{t-1}$ and $n \geq 36$, then we have

$$\mathbf{Adv}^{\text{nAE}}_{\text{Cymric1}}(q_e, q_d) \leq \frac{8(q_e + q_d)}{2^n} + \frac{2q_d}{2^t},$$

- We assume $q_e + 2^{n-t} \cdot q_d \leq 2^{n-1}$ and $n \geq 36$. If $q_e \leq \frac{2^n}{48n^2}$, then we have

$$\mathbf{Adv}^{\text{nAE}}_{\text{Cymric2}}(q_e, q_d) \leq \frac{(12 + 2^{\frac{t}{2}})q_e}{2^n} + \frac{7q_d}{2^t},$$

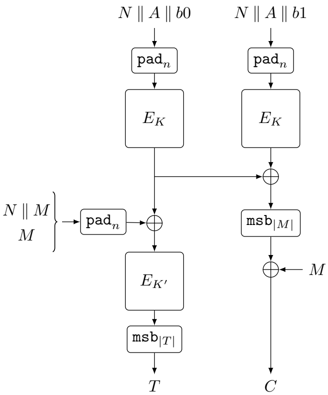
- By letting $t = 2n/3$ and $q = q_e + q_d$, we have

$$\mathbf{Adv}^{\text{nAE}}_{\text{Cymric2}}(q_e, q_d) \leq \frac{13q}{2^{2n/3}}.$$

Security

$$\mathbf{Adv}^{\mathsf{nAE}}_{\mathsf{Cymric1}}(q_e, q_d) \leq \frac{8q_e}{2^n} + \frac{10q_d}{2^t}$$

$$\mathbf{Adv}^{\mathsf{nAE}}_{\mathsf{Cymric2}}(q_e, q_d) \leq \frac{(12 + 2^{\frac{t}{2}})q_e}{2^n} + \frac{7q_d}{2^t}$$



Security Proof Setup (Cymric 2)

- $\widehat{\text{Enc}}$: outputs a tag without truncation for encryption queries
- (A variant of) $\widehat{\mathcal{S}}_{\text{real}} = (\widehat{\text{Enc}}, \text{Dec})$
- An intermediate world: $\widehat{\mathcal{S}}_{\text{inter}} = (\widehat{\$^*}, \perp)$
  - $\widehat{\$^*}$: takes (N, A, M) and output (C, T') where
    - C is a uniformly randomly chosen string of length $|M|$ (with replacement) and
    - T' is chosen uniformly randomly from $\{0,1\}^n$ without replacement if M is the same.

Security Proof Overview

$$\begin{aligned}\left\|\mathcal{S}_{\text{real}} - \mathcal{S}_{\text{ideal}}\right\| &\leq \left\|\mathcal{S}_{\text{real}} - \mathcal{S}_{\text{inter}}\right\| + \left\|\mathcal{S}_{\text{inter}} - \mathcal{S}_{\text{ideal}}\right\|, \\ &\leq \left\|\widehat{\mathcal{S}}_{\text{real}} - \widehat{\mathcal{S}}_{\text{inter}}\right\| + \frac{q_e}{2^{n-\frac{t}{2}}}.\end{aligned}$$

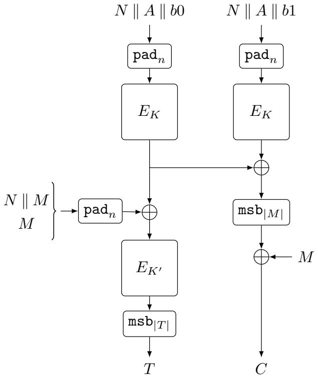
$$\left\|\widehat{\mathcal{S}}_{\text{real}} - \widehat{\mathcal{S}}_{\text{inter}}\right\| \leq \frac{12q_e}{2^n} + \frac{7q_d}{2^t}.$$

$$\left\|\widehat{\mathcal{S}}_{\mathsf{real}} - \widehat{\mathcal{S}}_{\mathsf{inter}}\right\| \le \frac{12q_e}{2^n} + \frac{7q_d}{2^t}$$

- bad$_1$ $\Leftrightarrow$ there exists $(i_1, \ldots, i_n) \in [1..q_e]^{*n}$ s.t. $T_{i_1} = \cdots = T_{i_n}$.
- bad$_2$ $\Leftrightarrow$ there exists $i \in [1..q]$ s.t. $S_i := M_i \oplus C_i = 0^n$.
- bad$_3$ $\Leftrightarrow$ there exists $(i, j) \in [1..q_e]^{*2}$ s.t. $T_i = T_j$ and

$$M'_i \oplus M'_j \in \{0^n, S_i, S_j, S_i \oplus S_j\}.$$

- Why bad?: 1) the real bad, and 2) to apply Mirror theory.
- Good analysis: use Mirror theory!



Mirror Theory

Theorem

Let Γ be a nice system over $\{0,1\}^n$ such that the number of equations is q and the number of inequalities is v. Suppose the number of variables in the largest component of $\gamma^=$ is $\xi_{\max}$. If $\xi_{\max}^2 n + \xi_{\max} \le 2^{n/2}$, $q\xi_{\max}^2 \le \frac{2^n}{12}$ and $q + v \le 2^{n-1}$, one has

$$h(\Gamma) \ge \frac{(2^n-2)_{|\mathcal{V}_1|}(2^n-2)_{|\mathcal{V}_2|}}{2^{nq}} \left(1 - \frac{2v}{2^n}\right).$$

Generic Construction

- We show the optimality of Cymric2
  - # of BC calls
  - no costly operations

- We define a generic construction of (short-input) AEs that uses linear operations and two BC calls.

- It accepts a ν-bit nonce and an n-bit plaintext, and returns an n-bit ciphertext and t-bit tag.

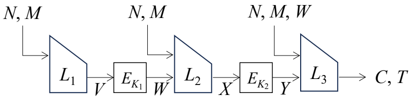


Figure 2: GAE with two block ciphers E_{K_1}, E_{K_2} and three linear functions L_1, L_2, L_3.

Impossibility results by attacking GAE

- Birthday Attack on GAE with $t = n$. There exists a $(q_e, 0)$-adversary $\mathcal{A}$ on GAE such that

$$\mathbf{Adv}^{\mathsf{nAE}}_{\mathsf{GAE}}(\mathcal{A}) = O\left(\frac{q_e^2}{2^n}\right)$$

- Birthday Attack on GAE with $t < n$. There exists a (q_e, q_d)-adversary $\mathcal{A}$ on GAE such that $q_d \leq 1$ and

$$\mathbf{Adv}^{\mathsf{nAE}}_{\mathsf{GAE}}(\mathcal{A}) = O\left(\frac{q_e^2}{2^n}\right)$$

Attacks

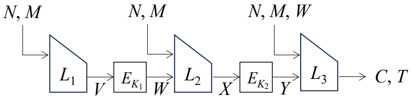


Figure 3: GAE with two block ciphers E_{K_1}, E_{K_2} and three linear functions L_1, L_2, L_3.

- WLOG, $C = a \cdot W \oplus b \cdot Y$ and $T = c \cdot W \oplus d \cdot Y$.
- Let $t = n$ and Fin $:= \begin{pmatrix} a & b \\ c & d \end{pmatrix}$.
- rank(Fin) ≤ 1.
  - There exists x s.t. $x \cdot C = T$
- rank(Fin) $= 2 \wedge \left(\exists(i,j) \text{ s.t. } (N_i, M_i) \neq (N_j, M_j) \wedge V_i = V_j\right)$.
  - Find a collision $W_i = W_j$ (computable from C and T)

Attacks (2)

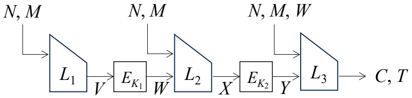


Figure 4: GAE with two block ciphers E_{K_1}, E_{K_2} and three linear functions L_1, L_2, L_3.

- rank(Fin) $= 2 \wedge \left(\forall (i,j) \text{ s.t. } (N_i, M_i) \neq (N_j, M_j) : V_i \neq V_j\right)$.
  1. Choose q_e pairs of nonce and plaintext $(N_1, M_1), \ldots, (N_{q_e}, M_{q_e})$ such that $V_1, \ldots, V_{q_e}$ are all distinct.
  2. For $i \in [1..q_e]$, make an encryption query (N_i, M_i) and receive the pair (C_i, T_i).
  3. For $i \in [1..q_e]$, recover W_i by solving the equations $C_i = a \cdot W_i \oplus b \cdot Y_i$; $T_i = c \cdot W_i \oplus d \cdot Y_i$.
  4. If $\exists i, j \in [1..q_e]$ s.t. $i \neq j \wedge W_i = W_j$, then return 0; Otherwise return 1.
- What if $t < n$?

Attacks (2)

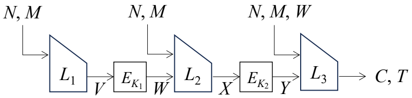


Figure 4: GAE with two block ciphers E_{K_1}, E_{K_2} and three linear functions L_1, L_2, L_3.

- $\text{rank}(\text{Fin}) = 2 \wedge \Big(\forall (i,j)$ s.t. $(N_i, M_i) \neq (N_j, M_j) : V_i \neq V_j\Big)$.
    1. Choose q_e pairs of nonce and plaintext $(N_1, M_1), \ldots, (N_{q_e}, M_{q_e})$ such that $V_1, \ldots, V_{q_e}$ are all distinct.
    2. For $i \in [1..q_e]$, make an encryption query (N_i, M_i) and receive the pair (C_i, T_i).
    3. For $i \in [1..q_e]$, recover W_i by solving the equations $C_i = a \cdot W_i \oplus b \cdot Y_i$; $T_i = c \cdot W_i \oplus d \cdot Y_i$.
    4. If $\exists i, j \in [1..q_e]$ s.t. $i \neq j \wedge W_i = W_j$, then return 0; Otherwise return 1.
- What if $t < n$?

Performance

Platform	Mode	Security (bits)	Speed (cycles) Scenario 1	Speed (cycles) Scenario 2	Memory (bytes) Key	Memory (bytes) Stack	Memory (bytes) Code
ATmega128 (AVR)	Cymric1	**128**	9 881	-	32	**238**	**2 152**
	Cymric2	85.3	**9 687**	**10 084**	32	**238**	2 766
	XOCB	85.3	26 699	26 989	**16**	295	7 632
	AES-GCM-SIV	64	52 211	42 126	**16**	537	5 656
	OCB	64	12 871	10 910	32	270	7 378
	GCM	64	39 239	59 628	32	490	4 466
STM32F407 (Cortex-M4)	Cymric1	**128**	9 644	-	32	472	**3 246**
	Cymric2	85.3	9 584	9 697	32	**464**	3 648
	XOCB	85.3	17 676	17 942	**16**	648	5 348
	AES-GCM-SIV	64	20 775	19 472	**16**	788	5 102
	OCB	64	**8 533**	**8 599**	32	640	4 996
	GCM	64	9 917	11 306	32	676	4 314

Table 2: Benchmark of various AE modes all instantiated with AES-128 as the underlying block cipher.

Benchmark of lightweight AE

AEAD	Security (bits)	Implementation	Speed (cycles)		Memory (bytes)		
			Scenario 1	Scenario 2	Key	Stack	Code
LEA128-Cymric1	**128**	Ours	19 163	-	32	491	1 590
			11 276*	-	768*	107*	**1 128***
LEA128-Cymric2	85.3		19 305	19 400	32	488	2 208
			11 416*	**11 517***	768*	104*	1 746*
GIFT128-Cymric1	**128**		31 609	-	32	427	5 162
			19 139*	-	640*	107*	2 252*
GIFT128-Cymric2	85.3		31 764	31 842	32	423	5 780
			19 293*	19 379*	640*	104*	2 870*
Ascon-AEAD128	**128**	`ascon/ascon-c`	24 143	18 661	**16**	122	4 036
Xoodyak	**128**	`rweather/lwc-finalists`	43 441	43 640	**16**	**98**	2 542
Romulus-N	**128**	`rweather/lwc-finalists`	30 364	30 525	**16**	165	5 592
PHOTON-Beetle-AEAD[128]	121	`rweather/lwc-finalists`	60 357	40 675	**16**	131	7 840
GIFT-COFB	64	`aadomn/gift`	27 224	26 993	**16**	398	9 192

* Using pre-computed round keys.

Table 3: Benchmark of lightweight AE schemes on AVR ATmega128.

Benchmark of lightweight AE

AEAD	Security (bits)	Implementation	Speed (cycles) Scenario 1	Speed (cycles) Scenario 2	Memory (bytes) Key	Memory (bytes) Stack	Memory (bytes) Code
LEA128-Cymric1	**128**	Ours	2 274	-	32	160	**1 052**
LEA128-Cymric2	85.3	Ours	**2 198**	**2 316**	32	**152**	1 390
GIFT128-Cymric1	**128**	Ours	8 218	-	32	800	2 224
			4 500*	-	640*	160*	1 268*
GIFT128-Cymric2	85.3	Ours	8 157	8 276	32	792	2 582
			4 438*	4 560*	640*	**152***	1 606*
Ascon-AEAD128	**128**	`ascon/ascon-c`	3 054	2 457	**16**	160	1 368
Xoodyak	**128**	`XKCP/XKCP`	3 572	3 669	**16**	240	3 304
Romulus-N	**128**	`aadomn/skinny`	11 061	11 199	**16**	980	9 868
PHOTON-Beetle-AEAD[128]	121	`rweather/lwc-finalists`	30 897	20 702	**16**	284	6 746
GIFT-COFB	64	`aadomn/gift`	6 600	6 405	**16**	496	3 970

* Using pre-computed round keys.

Table 4: Benchmark of lightweight AE schemes on ARM Cortex-M4.

Conclusion

- Recall: An intermediate world: $\widehat{\mathcal{S}}_{\text{inter}} = (\widehat{\$^*}, \perp)$
  - $\widehat{\$^*}$: takes (N, A, M) and output (C, T') where
    - C is a uniformly randomly chosen string of length $|M|$ (with replacement) and
    - T' is chosen uniformly randomly from $\{0,1\}^n$ without replacement if M is the same.
- Lower bounds for constructing encryption modes/MACs/AEAD
- Thank you for listening!

Conclusion

- Recall: An intermediate world: $\widehat{\mathcal{S}}_{\text{inter}} = (\widehat{\$^*}, \perp)$
  - $\widehat{\$^*}$: takes (N, A, M) and output (C, T') where
    - C is a uniformly randomly chosen string of length $|M|$ (with replacement) and
    - T' is chosen uniformly randomly from $\{0,1\}^n$ without replacement if M is the same.
- Lower bounds for constructing encryption modes/MACs/AEAD
- Thank you for listening!