Stronger Security Variants of GCM-SIV

Tetsu Iwata[1] and Kazuhiko Minematsu[2]

[1] Nagoya University, Nagoya, Japan, tetsu.iwata@nagoya-u.jp
[2] NEC Corporation, Kawasaki, Japan, k-minematsu@ah.jp.nec.com

Abstract. At CCS 2015, Gueron and Lindell proposed GCM-SIV, a provably secure authenticated encryption scheme that remains secure even if the nonce is repeated. While this is an advantage over the original GCM, we first point out that GCM-SIV allows a trivial distinguishing attack with about 2^{48} queries, where each query has one plaintext block. This shows the tightness of the security claim and does not contradict the provable security result. However, the original GCM resists the attack, and this poses a question of designing a variant of GCM-SIV that is secure against the attack. We present a minor variant of GCM-SIV, which we call GCM-SIV1, and discuss that GCM-SIV1 resists the attack, and it offers a security trade-off compared to GCM-SIV. As the main contribution of the paper, we explore a scheme with a stronger security bound. We present GCM-SIV2 which is obtained by running two instances of GCM-SIV1 in parallel and mixing them in a simple way. We show that it is secure up to $2^{85.3}$ query complexity, where the query complexity is measured in terms of the total number of blocks of the queries. Finally, we generalize this to show GCM-SIVr by running r instances of GCM-SIV1 in parallel, where $r \geq 3$, and show that the scheme is secure up to $2^{128r/(r+1)}$ query complexity. The provable security results are obtained under the standard assumption that the blockcipher is a pseudorandom permutation.

Keywords: GCM-SIV · nonce-reuse misuse-resistance authenticated encryption · provable security · beyond-birthday-bound security.

1 Introduction

AE Schemes and MRAE Schemes. An authenticated encryption (AE) scheme is a symmetric key primitive that is used for efficiently protecting both privacy and authenticity of digital data. The Galois/Counter Mode (GCM) of operation, proposed in 2004 by McGrew and Viega [MV04a, MV04b], is one of the widely used AE schemes. It is included in various standards e.g. in [Dwo07, IEE06, SCM08]. GCM is a nonce-based AE scheme, that is, it takes data called a nonce as a part of the input, and the security relies on the fact that the nonce is never repeated. Under this assumption, GCM is provably secure [IOM12, NOMI15], and when the nonce length is restricted to 96 bits, it is provably secure up to the standard birthday-bound of about $2^{n/2}$ query complexity, where n is the block length of the underlying blockcipher. When AES is used, $n = 128$, and the query complexity refers to the total number of blocks of the queries.

However, as with the case for many other nonce-based AE schemes, repeating the nonce has critical impact on the security of GCM. In fact, Joux showed that if the adversary can repeat the nonce, then a secret key of a universal hash function used in GCM called GHASH can be obtained, and the key can be used for a universal forgery attack [Jou06]. A practical threat to TLS implementations based on this attack was shown by Böck et al. [BZD+16]. While the mathematical definition of a non-repeating nonce is simple, implementation of the nonce is often non-trivial, and the assumption is repeatedly compromised in practice. For instance software to generate the nonce may contain a bug, or the nonce generation

process may rely on poor randomness. This leads to the formalization of nonce-reuse misuse-resistance AE (MRAE) by Rogaway and Shrimpton [RS06].

MRAE schemes are AE schemes that remain secure even if a nonce is repeated. Specifically, we consider AE schemes that take (N, A, M) as input, where N is a nonce, A is associated data, and M is a plaintext. N and A are supposed to be authenticated (but not encrypted), and M is authenticated and encrypted. The privacy of MRAE demands that the repetition of (N, A, M) is observed by the adversary, since the encryption algorithm returns the same output, while nothing else is revealed. The authenticity of MRAE demands that a forgery is impossible regardless of the repetition of a nonce.

The scheme we focus on in this paper is GCM-SIV, an MRAE scheme designed and proposed by Gueron and Lindell at CCS 2015 [GL15a]. The design follows the approach of SIV [RS06], and GCM-SIV can be seen as an efficient instantiation of SIV that uses components from GCM. Compared to GCM, the security advantage of GCM-SIV is that it remains secure even if the nonce is repeated. While achieving this may have efficiency penalty, the authors demonstrate that the efficiency loss is limited. GCM-SIV achieves 0.92 cycles per byte on the Broadwell architecture, showing that an MRAE scheme can achieve less than one cycle per byte, and the speed is close to that of GCM [GL15a]. The authors conclude that GCM-SIV is a viable alternative to GCM, providing full nonce-reuse misuse-resistance at little cost. An updated version of GCM-SIV was submitted as the Internet-Draft [GLL16].[1]

This Paper. We first observe that there is a subtle difference in the security bounds of GCM and GCM-SIV. Specifically, the security bound of GCM contains $O(\sigma^2/2^n)$ both for privacy and authenticity notions, while the security bound of GCM-SIV contains a term of the form $O(q^2/2^{n-k})$ instead, where σ is the total number of blocks of the queries, n denotes the block length in bits, q is the number of queries, and 2^k denotes the maximum block length of all the encryption and decryption queries, and n and k are specified as $n = 128$ and $k = 32$. That is, while GCM has a standard birthday-bound security which is expressed in terms of the total number of blocks of the queries, the security bound of GCM-SIV is in the number of queries. In this paper, we first point out that there is a trivial distinguishing attack against GCM-SIV that works with about $2^{(n-k)/2}$ queries, which amounts to 2^{48} queries with $n = 128$ and $k = 32$, and each query has one plaintext block. This merely shows the tightness of the security claim and does not contradict the provable security result. If we compare the two security bounds, $O(\sigma^2/2^n)$ and $O(q^2/2^{n-k})$, we see that $O(\sigma^2/2^n)$ is better (smaller) if $\sigma/q \leq 2^{k/2}$, i.e., if the average query length (in blocks) is smaller than $2^{k/2}$ blocks, while $O(q^2/2^{n-k})$ is better otherwise. We note that $2^{k/2}$ blocks correspond to 1 MB (megabyte), and this is larger than the maximum packet size on the Internet. Therefore, there is a practical case where achieving the security bound of the form $O(\sigma^2/2^n)$ is desirable.

This observation motivates us to design a variant of GCM-SIV that resists the above mentioned attack. We present a minor variant of GCM-SIV, which we call GCM-SIV1. We discuss that GCM-SIV1 resists the attack, but it turns out that there are cases where the security bound of GCM-SIV is better, and therefore, the security bound of GCM-SIV1 offers a trade-off compared to GCM-SIV. We point out that the security bound of GCM-SIV1 is comparable to GCM, that is, it is secure up to the standard birthday-bound security of about $2^{n/2}$ query complexity, under the assumption that the average query length is at least $\ell^{1/2}$ blocks, where ℓ denotes the maximum block length of the queries. The algorithmic modification is minimal, while on the downside, it would raise the implementation cost in particular if we reuse components of GCM.

Next, as the main contribution of the paper, we explore a scheme with a stronger security guarantee. A scheme that is secure beyond $2^{n/2}$ query complexity is often called

[1] Throughout this paper, GCM-SIV refers to the three-key construction in [GL15a].

to have the beyond-birthday-bound (BBB) security. We consider a theoretical question of designing a simple BBB secure MRAE scheme from a blockcipher, and we first present GCM-SIV2 that is obtained by running two instances of GCM-SIV1 in parallel and mixing them in a simple way. We show that it is secure up to about $2^{2n/3}$ query complexity. As with GCM-SIV and GCM-SIV1, GCM-SIV2 follows the design approach of SIV [RS06] and one of the generic constructions called A4 [NRS14], where it combines a pseudo-random function (PRF) and an IV-based encryption (ivE) scheme. Here, ivE is an encryption scheme that encrypts a plaintext using an initialization vector IV that is randomly chosen inside the encryption process. The design of GCM-SIV2 can be seen as combining a BBB secure PRF and a BBB secure ivE scheme. Previous BBB secure PRFs based on a blockcipher include SUM-ECBC [Yas10], a variant of PMAC in [Yas11], the Hash-then-Sum construction [DDN+15], and a construction in [Osa12]. The PRF in GCM-SIV2 is similar to them, with a difference being the length of the output. That is, the output length of the PRF in GCM-SIV2 is $2n$ bits, while the length is n bits in the above mentioned schemes. This is a disadvantage from a view point of communication cost, however, this plays a crucial role in proving the BBB security when combined with a BBB secure ivE scheme. We note that a generic approach of constructing a BBB secure MRAE scheme is discussed in [IY09a, IY09b]. These are either complex and inefficient, or require a stronger primitive than a blockcipher called a tweakable blockcipher [LRW11]. However, we remark that a tweakable blockcipher based AE scheme can be instantiated with a blockcipher by using a BBB secure tweakable blockcipher in [LST12].

One feature of GCM-SIV2 is that the design approach is scalable in that it allows strengthening the security bound by naturally increasing the number of instances of GCM-SIV1. We present GCM-SIVr by running r instances of GCM-SIV1 in parallel, where $r \geq 3$, and show that the scheme is secure up to about $2^{nr/(r+1)}$ query complexity. The security bound of GCM-SIVr approaches 2^n query complexity as r grows, however, this comes with costs. The efficiency significantly degrades as r grows, and reusing GCM implementations becomes more difficult than the original GCM-SIV.

We note that the construction is generic in that any universal hash function can be used, while we heavily rely on the fact that the underlying ivE scheme is CTR mode. We remark that all the provable security results are obtained under the standard assumption that the blockcipher is a pseudorandom permutation. We also emphasize that all our results hold under nonce-misuse setting, where the adversary can repeat the nonce.

Related Work. MRAE schemes provide a strong security guarantee, and they can be obtained from a deterministic AE (DAE) scheme, which addresses the problem of key wrapping, by encoding a nonce into a part of the input of the DAE scheme [RS06]. There have been proposals that offer the strong security guarantee of MRAE or DAE. The first scheme called SIV mentioned above was proposed by Rogaway and Shrimpton [RS06]. This followed by HBS [IY09b] and BTM [IY09a]. Reyhanitabar et al. proposed a hash function-based MRAE scheme called misuse-resistant OMD [RVV14]. Chakraborty and Sarkar [CS16] and Sarkar [Sar14] showed comprehensive studies on generic constructions of AE and DAE schemes from a blockcipher or a stream cipher. See also the constructions by Shrimpton and Terashima [ST13] of BBB secure AEAD schemes. Robust AE (RAE) introduced by Hoang et al. [HKR15] is closely related to MRAE, where a RAE scheme has a more flexible security guarantee than that of plain DAE. There are more recent schemes, such as Abed et al. [AFL+16] and Granger et al. [GJMN16].

Another direction is on-line AE schemes, where they are more efficient than MRAE schemes, while the security notion is weaker [FFL12]. Finally, CAESAR [CAE], a competition for AE schemes, attracted submissions of MRAE schemes, including AEZ [HKR15], HS1-SIV [Kro15], and Synthetic Counter in Tweak (SCT) mode [PS16] employed in Deoxys [JNP15a] and Joltik [JNP15b].

2 Preliminaries

Notation. We write $\{0,1\}^*$ for the set of all finite bit strings including the empty string ε. For a bit string $X \in \{0,1\}^*$, $|X|$ denotes its length in bits, and for an integer $\ell \geq 1$, $|X|_\ell = \lceil |X|/\ell \rceil$ denotes the length in ℓ-bit blocks. For an integer $\ell \geq 0$, let $\{0,1\}^\ell$ be the set of all bit strings of ℓ bits, and $\mathrm{Perm}(\ell)$ be the set of all permutations on $\{0,1\}^\ell$. For $X, Y \in \{0,1\}^*$, $X \,\|\, Y$ denotes their concatenation, which is also written as XY for simplicity. We write the ℓ-bit zero string as $0^\ell = 0 \cdots 0 \in \{0,1\}^\ell$. For $X \in \{0,1\}^*$ with $|X| \geq \ell$, $\mathsf{msb}_\ell(X)$ denotes the first (leftmost) ℓ bits of X, and $\mathsf{lsb}_\ell(X)$ denotes the last (rightmost) ℓ bits of X. For $X \in \{0,1\}^*$ and $\ell \geq 1$, $(X[1], \ldots, X[x]) \xleftarrow{\ell} X$ denotes the decomposition of X into ℓ-bit blocks, where $x = |X|_\ell$, i.e., $X[1], \ldots, X[x]$ are unique bit strings such that $X[1] \,\|\, \cdots \,\|\, X[x] = X$, $|X[1]| = \cdots = |X[x-1]| = \ell$, and $1 \leq |X[x]| \leq \ell$. We follow the convention that if $X = \varepsilon$, then $X[1] \xleftarrow{\ell} X$, where $X[1] = \varepsilon$. For a finite set $\mathcal{X}$, $X \xleftarrow{\$} \mathcal{X}$ means a uniform random sampling of an element X from $\mathcal{X}$. For integers ℓ and x such that $x < 2^\ell$, $\mathsf{str}_\ell(x)$ is the standard ℓ-bit binary representation of x.

Nonce-Based AEAD. A nonce-based authenticated encryption with associated data (AEAD) scheme Π consists of an encryption algorithm $\Pi\text{-}\mathcal{E}$ and a decryption algorithm $\Pi\text{-}\mathcal{D}$, and it is associated with a set of keys $\mathcal{K}_\Pi \subseteq \{0,1\}^*$. We write $\Pi = (\mathcal{K}_\Pi, \Pi\text{-}\mathcal{E}, \Pi\text{-}\mathcal{D})$. The encryption algorithm $\Pi\text{-}\mathcal{E}$ takes a key $\boldsymbol{K} \in \mathcal{K}_\Pi$, a nonce $N \in \mathcal{N}_\Pi$, associated data $A \in \mathcal{A}_\Pi$, and a plaintext $M \in \mathcal{M}_\Pi$ as input, and returns a ciphertext $C \in \{0,1\}^*$ and a tag $T \in \{0,1\}^\tau$ for some fixed τ, where $\mathcal{N}_\Pi$ is the set of nonces, $\mathcal{A}_\Pi$ is the set of associated data, $\mathcal{M}_\Pi$ is the set of plaintexts, and $\mathcal{N}_\Pi, \mathcal{A}_\Pi, \mathcal{M}_\Pi \subseteq \{0,1\}^*$. We write $(C,T) \leftarrow \Pi\text{-}\mathcal{E}_{\boldsymbol{K}}(N,A,M)$. The decryption algorithm $\Pi\text{-}\mathcal{D}$ takes $\boldsymbol{K}$, N, A, C, and T as input, and returns M or the reject symbol $\perp$. We write $M/\perp \leftarrow \Pi\text{-}\mathcal{D}_{\boldsymbol{K}}(N,A,C,T)$. If $(C,T) \leftarrow \Pi\text{-}\mathcal{E}_{\boldsymbol{K}}(N,A,M)$, then we have $M \leftarrow \Pi\text{-}\mathcal{D}_{\boldsymbol{K}}(N,A,C,T)$.

We follow the security definition in [GL15a], which follows [NRS14, RS06]. Let $\mathcal{A}$ be an adversary against $\Pi = (\mathcal{K}_\Pi, \Pi\text{-}\mathcal{E}, \Pi\text{-}\mathcal{D})$. We define the MRAE-advantage of $\mathcal{A}$ as

$$\mathbf{Adv}^{\mathrm{mrae}}_{\Pi}(\mathcal{A}) = \Pr\left[\mathcal{A}^{\mathsf{Enc}_{\boldsymbol{K}},\mathsf{Dec}_{\boldsymbol{K}}} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$,\perp} \Rightarrow 1\right],$$

where $\mathsf{Enc}_{\boldsymbol{K}}$ is the encryption oracle that takes (N,A,M) as input and returns $(C,T) \leftarrow \Pi\text{-}\mathcal{E}_{\boldsymbol{K}}(N,A,M)$, $\mathsf{Dec}_{\boldsymbol{K}}$ is the decryption oracle that takes (N,A,C,T) as input and returns $M/\perp \leftarrow \Pi\text{-}\mathcal{D}_{\boldsymbol{K}}(N,A,C,T)$, \$ is the random-bits oracle that takes (N,A,M) as input and returns a random string $(C,T) \xleftarrow{\$} \{0,1\}^\ell$, where $\ell = |\Pi\text{-}\mathcal{E}_{\boldsymbol{K}}(N,A,M)|$ and is assumed to depend only on $|N|$, $|A|$, and $|M|$, and finally the reject oracle $\perp$ takes (N,A,C,T) as input and returns $\perp$. The probabilities are taken over $\boldsymbol{K}$, $\mathcal{A}$, and \$, and we without loss of generality assume that $\mathcal{A}$ does not repeat the same query.

IV-Based Encryption. An IV-based encryption (ivE) scheme E consists of an encryption algorithm $\mathsf{E}\text{-}\mathcal{E}$ and a decryption algorithm $\mathsf{E}\text{-}\mathcal{D}$, with a set of keys $\mathcal{K}_\mathsf{E} \subseteq \{0,1\}^*$. We write $\mathsf{E} = (\mathcal{K}_\mathsf{E}, \mathsf{E}\text{-}\mathcal{E}, \mathsf{E}\text{-}\mathcal{D})$. The encryption algorithm $\mathsf{E}\text{-}\mathcal{E}$ takes a key $K \in \mathcal{K}_\mathsf{E}$ and a plaintext $M \in \mathcal{M}_\mathsf{E}$ as input, and returns (IV,C), where $IV \xleftarrow{\$} \{0,1\}^\tau$ is an initialization vector for some fixed τ and $C \in \{0,1\}^*$ is a ciphertext. We write $(IV,C) \leftarrow \mathsf{E}\text{-}\mathcal{E}_K(M)$. The decryption algorithm $\mathsf{E}\text{-}\mathcal{D}$ takes K, IV, and C as input, and returns M. We write $M \leftarrow \mathsf{E}\text{-}\mathcal{D}_K(IV,C)$. If $(IV,C) \leftarrow \mathsf{E}\text{-}\mathcal{E}_K(M)$, then we have $M \leftarrow \mathsf{E}\text{-}\mathcal{D}_K(IV,C)$.

We use the security definition in [GL15a, NRS14, RS06]. Let $\mathcal{A}$ be an adversary against $\mathsf{E} = (\mathcal{K}_\mathsf{E}, \mathsf{E}\text{-}\mathcal{E}, \mathsf{E}\text{-}\mathcal{D})$. We define the priv\$-advantage of $\mathcal{A}$ as

$$\mathbf{Adv}^{\mathrm{priv\$}}_{\mathsf{E}}(\mathcal{A}) = \Pr\left[\mathcal{A}^{\mathsf{Enc}_K} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$} \Rightarrow 1\right],$$

where the encryption oracle Enc_K takes M as input and returns $(IV, C) \leftarrow \mathsf{E}\text{-}\mathcal{E}_K(M)$, and the random-bits oracle \$ takes M as input and returns a random string $(IV, C) \xleftarrow{\$} \{0,1\}^\ell$, where $\ell = |\mathsf{E}\text{-}\mathcal{E}_K(M)|$ depends only on $|M|$. The probabilities are taken over K, $\mathcal{A}$, and \$.

PRF. A pseudo-random function (PRF) F is a keyed function with a set of keys $\mathcal{K}_\mathsf{F}$, domain $\mathcal{D}_\mathsf{F}$, and range $\{0,1\}^\tau$ for some fixed τ. It takes $K \in \mathcal{K}_\mathsf{F}$ and $X \in \mathcal{D}_\mathsf{F}$ as input, and returns $Y = \mathsf{F}_K(X) \in \{0,1\}^\tau$. Let $\mathcal{R} \xleftarrow{\$} \mathrm{Rand}(\mathcal{D}_\mathsf{F}, \{0,1\}^\tau)$ be a random function, where $\mathrm{Rand}(\mathcal{D}_\mathsf{F}, \{0,1\}^\tau)$ is the set of all functions with domain $\mathcal{D}_\mathsf{F}$ and range $\{0,1\}^\tau$.

We define the prf-advantage of an adversary $\mathcal{A}$ as

$$\mathbf{Adv}^{\mathrm{prf}}_\mathsf{F}(\mathcal{A}) = \Pr\left[\mathcal{A}^{\mathsf{F}_K} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\mathcal{R}} \Rightarrow 1\right],$$

where the oracle F_K takes X as input and returns $Y \leftarrow \mathsf{F}_K(X)$, and the oracle $\mathcal{R}$ takes X as input and returns a random element $Y \xleftarrow{\$} \mathcal{R}(X)$. The probabilities are taken over K, $\mathcal{A}$, and $\mathcal{R}$.

Blockcipher. A blockcipher E is a keyed function that takes a key $K \in \mathcal{K}_E$ and a plaintext block $M \in \{0,1\}^n$ as input, and returns a ciphertext block $C \in \{0,1\}^n$, where $\mathcal{K}_E \subseteq \{0,1\}^*$ is a non-empty set of keys and n is the block length. We write $C \leftarrow E_K(M)$, and for each $K \in \mathcal{K}_E$, the function $E_K : \{0,1\}^n \to \{0,1\}^n$ is a permutation, i.e., $E_K \in \mathrm{Perm}(n)$. We call $P \xleftarrow{\$} \mathrm{Perm}(n)$ a random permutation.

Hash Function. We consider a keyed hash function H with a set of keys $\mathcal{K}_H \subseteq \{0,1\}^*$, domain $\mathcal{D}_H$, and range $\{0,1\}^n$, where we assume the length of the output is the same as the block length of the blockcipher. It takes a key $L \in \mathcal{K}_H$ and $X \in \mathcal{D}_H$ as input, and returns the output $Y \leftarrow H_L(X) \in \{0,1\}^n$. H is an ϵ-almost universal (ϵ-AU) hash function if for any distinct $X, X' \in \mathcal{D}_H$, it holds that $\Pr_L[H_L(X) = H_L(X')] \le \epsilon$.

3 SIV: MRAE from PRF and ivE

Here we recall a result related to SIV [NRS14, RS06]. SIV is an MRAE scheme that is obtained by combining a PRF F and an ivE scheme E. Consider a nonce-based AEAD scheme $\Pi = (\mathcal{K}_\Pi, \Pi\text{-}\mathcal{E}, \Pi\text{-}\mathcal{D})$ composed from F and E, where $\mathcal{D}_\mathsf{F} = \mathcal{N}_\Pi \times \mathcal{A}_\Pi \times \mathcal{M}_\Pi$, and $\mathcal{M}_\mathsf{E} = \mathcal{M}_\Pi$. Given $\boldsymbol{K} = (K_1, K_2) \in \mathcal{K}_\mathsf{F} \times \mathcal{K}_\mathsf{E}$, to encrypt (N, A, M) into $(C, T) \leftarrow \Pi\text{-}\mathcal{E}_{\boldsymbol{K}}(N, A, M)$, we first let $T \leftarrow \mathsf{F}_{K_1}(N, A, M)$, and then $C \leftarrow \mathsf{E}\text{-}\mathcal{E}_{K_2}(M)$, where we use the tag T as the IV in $\mathsf{E}\text{-}\mathcal{E}$. To decrypt (N, A, C, T), we first let $M \leftarrow \mathsf{E}\text{-}\mathcal{D}_{K_2}(T, C)$ and $T^* \leftarrow \mathsf{F}_{K_1}(N, A, M)$, and then return M if $T = T^*$ and $\perp$ otherwise. We call this the SIV construction using F as tag generation and E as the ivE scheme.

SIV is secure as an MRAE scheme if its tag generation is a PRF and the ivE scheme is secure. More precisely, we say that $\mathcal{A}$ is a (q, ℓ, σ)-adversary if it makes at most q_1 encryption queries (N_i, A_i, M_i), $1 \le i \le q_1$, and at most q_2 decryption queries (N'_i, A'_i, C'_i, T'_i), $1 \le i \le q_2$, where

- $q_1 + q_2 \le q$,
- $|N_i|_n + |A_i|_n + |M_i|_n \le \ell$ for all $i \in \{1, \ldots, q_1\}$,
- $|N'_i|_n + |A'_i|_n + |C'_i|_n \le \ell$ for all $i \in \{1, \ldots, q_2\}$, and
- $\sum_{1 \le i \le q_1} |M_i|_n \le \sigma$.

Algorithm GCM-SIV-$\mathcal{E}_{\boldsymbol{K}}(N, A, M)$	**Algorithm** GCM-SIV-$\mathcal{D}_{\boldsymbol{K}}(N, A, C, T)$
1. $V \leftarrow H_L(N, A, M)$	1. $IV \leftarrow \mathsf{msb}_{n-k}(T) \,\|\, 0^k$
2. $T \leftarrow E_{K'}(V)$	2. $m \leftarrow \|C\|_n$
3. $IV \leftarrow \mathsf{msb}_{n-k}(T) \,\|\, 0^k$	3. $\mathbf{S} \leftarrow \mathsf{CTR}_K(IV, m)$
4. $m \leftarrow \|M\|_n$	4. $M \leftarrow C \oplus \mathsf{msb}_{\|C\|}(\mathbf{S})$
5. $\mathbf{S} \leftarrow \mathsf{CTR}_K(IV, m)$	5. $V \leftarrow H_L(N, A, M)$
6. $C \leftarrow M \oplus \mathsf{msb}_{\|M\|}(\mathbf{S})$	6. $T^* \leftarrow E_{K'}(V)$
7. **return** (C, T)	7. **if** $T \neq T^*$ **then return** $\perp$
	8. **return** M

Figure 1: Definitions of GCM-SIV-$\mathcal{E}_{\boldsymbol{K}}(N, A, M)$ and GCM-SIV-$\mathcal{D}_{\boldsymbol{K}}(N, A, C, T)$

We note that σ does not include the lengths of associated data or nonces, but this is sufficient for the security analysis. Then the security of Π against (q, ℓ, σ)-adversaries can be proved as in the lemma below, which is shown in Theorem 2 of [RS06], Corollary 2.3 of [GL15a], and Theorem 1 of [NRS14].

Lemma 1. *For any (q, ℓ, σ)-adversary $\mathcal{A}$, we have*

$$\mathbf{Adv}^{\mathrm{mrae}}_{\Pi}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}}(\mathcal{A}') + \mathbf{Adv}^{\mathrm{priv\$}}_{\mathsf{E}}(\mathcal{A}'') + \frac{q}{2^\tau} \tag{1}$$

for some $\mathcal{A}'$ that makes at most q queries and each query is at most ℓ blocks, and some $\mathcal{A}''$ that makes at most q queries consisting of at most σ blocks in total.

Since all the schemes we consider in this paper fall into the SIV construction, their security proofs are reduced to showing the security of the underlying PRF and the ivE scheme.

4 Specification of GCM-SIV

We recall the specification of GCM-SIV [GL15a]. It uses a blockcipher $E : \mathcal{K}_E \times \{0,1\}^n \to \{0,1\}^n$ and a hash function H with the set of keys $\mathcal{K}_H$. We also fix an integer k that specifies the maximum input length. GCM-SIV-$\mathcal{E}$ takes a key $\boldsymbol{K} = (L, K', K) \in \mathcal{K}_H \times \mathcal{K}_E \times \mathcal{K}_E$, N, A, and M as input, and returns $(C, T) \leftarrow$ GCM-SIV-$\mathcal{E}_{\boldsymbol{K}}(N, A, M)$ such that $|C| = |M|$ and $|T| = n$. The algorithm is defined in Fig. 1 (left) and illustrated in Fig. 3. GCM-SIV can be defined in a more general manner, but we focus on a particular instance of GCM-SIV where we use AES as E, and we thus have $n = 128$, H is defined by using GHASH in Fig. 2, and $k = 32$, following GCM. Specifically, the hash function used in GCM-SIV is defined as $H_L(N, A, M) = \mathsf{GHASH}_L(A, M) \oplus N$, where $L \in \mathcal{K}_H = \{0,1\}^n$, $N \in \{0,1\}^n$, and $A, M \in \{0,1\}^*$ with the restriction that $|A|_n + |M|_n + 1 \leq 2^{32}$. In Fig. 2, the multiplication at line 6 is defined over $\mathrm{GF}(2^n)$. We use CTR mode based on E defined in Fig. 2, where the increment function is defined as $\mathsf{inc}(X) = \mathsf{msb}_{n-32}(X) \,\|\, \mathsf{lsb}_{32}(X) + 1 \bmod 2^{32}$. Here, we naturally interpret $\mathsf{lsb}_{32}(X)$ as an integer, and $\mathsf{lsb}_{32}(X) + 1 \bmod 2^{32}$ as a 32-bit string. GCM-SIV-$\mathcal{D}$ takes $\boldsymbol{K} = (L, K', K)$, N, A, C such that $|A|_n + |C|_n + 1 \leq 2^{32}$, and T as input, and returns $M \leftarrow$ GCM-SIV-$\mathcal{D}_{\boldsymbol{K}}(N, A, C, T)$ with $|M| = |C|$ or the reject symbol $\perp$. The algorithm is defined in Fig. 1 (right).

Algorithm $\mathsf{GHASH}_L(A, M)$

1. $\mathsf{len} \leftarrow \mathsf{str}_{n/2}(|A|) \,\|\, \mathsf{str}_{n/2}(|M|)$
2. $X \leftarrow A \,\|\, 0^{n|A|_n - |A|} \,\|\, M \,\|\, 0^{n|M|_n - |M|} \,\|\, \mathsf{len}$
3. $(X[1], \ldots, X[x]) \xleftarrow{n} X$
4. $Y \leftarrow 0^n$
5. **for** $j \leftarrow 1$ **to** x **do**
6. $\quad Y \leftarrow L \cdot (Y \oplus X[j])$
7. **return** Y

Algorithm $\mathsf{CTR}_K(IV, m)$

1. **if** $m = 0$ **then** $\mathbf{S} \leftarrow \varepsilon$
2. **else** // $m \geq 1$
3. $\quad I[1] \leftarrow IV$
4. $\quad S[1] \leftarrow E_K(I[1])$
5. $\quad$ **for** $i \leftarrow 2$ **to** m **do**
6. $\quad\quad I[i] \leftarrow \mathsf{inc}(I[i-1])$
7. $\quad\quad S[i] \leftarrow E_K(I[i])$
8. $\quad \mathbf{S} \leftarrow S[1] \,\|\, \cdots \,\|\, S[m]$
9. **return S**

Figure 2: Definitions of $\mathsf{GHASH}_L(A, M)$ and $\mathsf{CTR}_K(IV, m)$

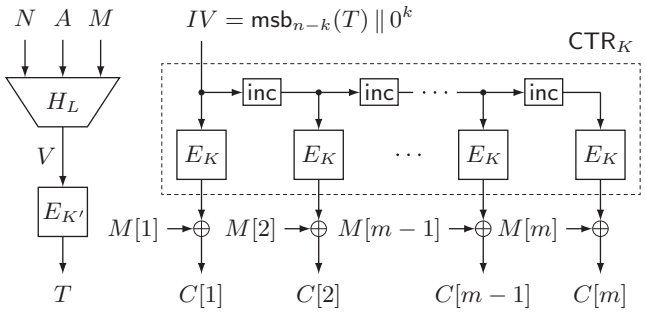

Figure 3: The encryption algorithm of GCM-SIV

5 Distinguishing GCM-SIV

Gueron and Lindell showed in [GL15a] that

$$\mathbf{Adv}^{\mathrm{mrae}}_{\text{GCM-SIV}}(\mathcal{A}) \leq 2\mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{A}') + \frac{q^2}{2^{95}} + \frac{q^2 + q'}{2^{128}}, \tag{2}$$

where $\mathcal{A}$ is an adversary that makes q encryption queries and q' decryption queries, and $\mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{A}')$ is the security of the underlying blockcipher E as a PRF, which is assumed to be small. Here, $q^2/2^{95}$ is derived as a sum of counter collision and GHASH collision probabilities, where both probabilities are bounded by $q^2/2^{n-k} = q^2/2^{96}$. See [GL15b, Sect. 4.1].

We show that the above security bound is tight by pointing out a trivial distinguishing attack on GCM-SIV. Let $q = 2^{(n-32)/2}$, $N_1, \ldots, N_q$ be q distinct nonces, $A = \varepsilon$, and $M = 0^n$. The adversary $\mathcal{A}$ first makes q encryption queries $(N_1, A, M), \ldots, (N_q, A, M)$ to receive $(C_1, T_1), \ldots, (C_q, T_q)$. Then $\mathcal{A}$ returns 1 if both $\mathsf{msb}_{n-32}(T_i) = \mathsf{msb}_{n-32}(T_{i'})$ and $C_i = C_{i'}$ hold for some $1 \leq i < i' \leq q$.

If the encryption oracle implements GCM-SIV, then with a high probability, we have $\mathsf{msb}_{n-32}(T_i) = \mathsf{msb}_{n-32}(T_{i'})$ for some $1 \leq i < i' \leq q$, in which case we also have $C_i = C_{i'}$. However, $C_i = C_{i'}$ holds with a probability of $1/2^n$ for the random-bits oracle, and hence the advantage of $\mathcal{A}$ is not small.

This observation does not violate the security claim in [GL15a]. On the other hand, for original GCM, when the nonce length is restricted to 96 bits and a random permutation is

used as the underlying blockcipher, it was shown in [IOM12] that

$$\mathbf{Adv}^{\mathrm{priv}}_{\mathrm{GCM}}(\mathcal{A}) \leq \frac{0.5(\sigma+q+1)^2}{2^n}, \tag{3}$$

where $\mathbf{Adv}^{\mathrm{priv}}_{\mathrm{GCM}}(\mathcal{A})$ measures the ability of the adversary to distinguish the ciphertexts from random bits. This value remains small with $q = 2^{(n-32)/2}$ and $\sigma = 2^{(n-32)/2}$, where σ denotes total block length of the ciphertexts. Therefore, GCM-SIV has stronger security than GCM in that it resists attacks that repeat a nonce, but there is a case where GCM-SIV is quantitatively less secure in that the distinguishing attack succeeds with the lower number of queries.

As explained earlier, the observation here is that the security bound of GCM contains $\sigma^2/2^n$ while that of GCM-SIV contains $q^2/2^{n-k}$ for the maximum input length 2^k in blocks. The former is better (smaller) if σ/q is at most $2^{k/2}$, which is around 1 MB (megabyte) for $n = 128$ and $k = 32$. We speculate that this average query length condition is practical considering standard communication protocols, where each message is at most a few KB (kilobyte), and that offline computation is required in MRAE schemes including GCM-SIV.

6 GCM-SIV1: A Variant of GCM-SIV

In this section, we consider a problem of designing a variant of GCM-SIV that resists the attack of the previous section. We make the following changes to GCM-SIV.

- The 3rd line of GCM-SIV-$\mathcal{E}_{\boldsymbol{K}}(N, A, M)$ in Fig. 1 is changed to $IV \leftarrow T$.
- The 1st line of GCM-SIV-$\mathcal{D}_{\boldsymbol{K}}(N, A, C, T)$ in Fig. 1 is changed to $IV \leftarrow T$.
- The definition of $\mathsf{inc}(X)$ used in CTR in Fig. 2 is changed to $\mathsf{inc}(X) = X + 1 \bmod 2^n$.

In other words, instead of using a part of T as an IV for CTR mode, we simply use the entire tag as the IV. We also remove the restriction $|A|_n + |M|_n + 1 \leq 2^{32}$ for GCM-SIV-$\mathcal{E}_{\boldsymbol{K}}(N, A, M)$ and $|A|_n + |C|_n + 1 \leq 2^{32}$ for GCM-SIV-$\mathcal{D}_{\boldsymbol{K}}(N, A, C, T)$. Instead, we assume the same restriction as GCM, which says $|A|_n \leq 2^{n/2}$, $|M|_n \leq 2^{32} - 2$, and $|C|_n \leq 2^{32} - 2$. This can improve the usability, which is also taken in the updated version of GCM-SIV [GLL16].

We use Lemma 1 to show the security of GCM-SIV1. It is sufficient to show that a function F that maps (N, A, M) to T using a key (L, K') is a PRF, and CTR mode is a secure ivE scheme. Consider GCM-SIV1 that uses an ϵ-AU hash function H and random permutations P' and P as blockciphers $E_{K'}$ and E_K. For a (q, ℓ, σ)-adversary $\mathcal{A}$, we have the following bound.

$$\mathbf{Adv}^{\mathrm{mrae}}_{\text{GCM-SIV1}}(\mathcal{A}) \leq 0.5q^2\epsilon + \frac{0.5q^2}{2^n} + \frac{\sigma^2}{2^n} + \frac{q}{2^n}$$

It is folklore to show that the function F is a PRF with advantage at most $0.5q^2\epsilon + 0.5q^2/2^n$, and CTR mode is secure in the sense of the ivE scheme with advantage at most $\sigma^2/2^n$ when IV is n bits. If we use GHASH in H as in GCM-SIV, then the bound becomes

$$\mathbf{Adv}^{\mathrm{mrae}}_{\text{GCM-SIV1}}(\mathcal{A}) \leq \frac{0.5q^2\ell}{2^n} + \frac{0.5q^2}{2^n} + \frac{\sigma^2}{2^n} + \frac{q}{2^n} \tag{4}$$

from $\epsilon \leq \ell/2^n$ [MV04a].

We discuss the relation between the security bounds of (2), (3), and (4). For simplicity, let us consider the security bounds of the form $q^2/2^{n-32}$ (GCM-SIV), $\sigma^2/2^n$ (GCM), and $q^2\ell/2^n + \sigma^2/2^n$ (GCM-SIV1) by taking the leading terms from (2), (3), and (4) and ignoring the constants. We make the following observations.

- We first see that $q^2/2^{n-32} \leq \sigma^2/2^n$ holds if $2^{16} \leq \sigma/q$, implying that the security bound of GCM-SIV is better (smaller) than that of GCM if the average input length is at least 2^{16} blocks.

- We next observe that $\sigma^2/2^n$ is always better than $q^2\ell/2^n + \sigma^2/2^n$, i.e., the security bound of GCM is always better than that of GCM-SIV1. However, the security bound of GCM-SIV1 can be bounded by $O(\sigma^2/2^n)$ if $\sigma/q \geq \ell^{1/2}$, i.e., if the average query length is at least $\ell^{1/2}$ blocks, in which case GCM-SIV1 achieves the standard birthday-bound security as with GCM.

- We then see that $q^2\ell/2^n + \sigma^2/2^n \leq q^2/2^{n-32}$ if $\sigma/q \leq (2^{32} - \ell)^{1/2}$. It depends on the usage of the scheme if $\sigma/q \leq (2^{32} - \ell)^{1/2}$ holds. Since the average query length is at most the maximum query length, it holds that $\sigma/q \leq \ell$, and if $\ell \leq (2^{32} - \ell)^{1/2}$, which can be approximated as $\ell \leq 2^{16}$, i.e., if the maximum query length is at most 2^{16} blocks, then the security bound of GCM-SIV1 is better than that of GCM-SIV. Therefore, even though the attack described in Sect. 5 does not work on GCM-SIV1, it does not mean that the security bound of GCM-SIV1 is always better.

- Suppose that the length of the plaintext in all the queries is about 2^{32} blocks. Then we observe that GCM-SIV remains secure as long as the number of queries is well below than 2^{48}, while GCM and GCM-SIV1 require that the number of queries is well below than 2^{32}. Therefore, in this case, GCM-SIV gives the strongest security bound.

7 GCM-SIV2: BBB Secure Scheme

Specification of GCM-SIV2. GCM-SIV1 in the previous section is secure up to about $2^{n/2}$ query complexity by using an n-bit tag. This is the best possible security among the construction in Lemma 1, since the best possible security of an ivE scheme with an n-bit IV is $O(q^2/2^n)$ priv\$-advantage, due to collisions among IVs. In this section, we present an extension that uses multiple instances of GCM-SIV1 to achieve stronger security, namely beyond-birthday-bound (BBB) security, at the cost of increased tag length. We call our BBB secure scheme GCM-SIV2. A pseudocode of GCM-SIV2 is shown in Fig. 4, and the encryption function is illustrated in Fig. 5. The global structure of GCM-SIV2 follows the SIV construction mentioned in Sect. 3, combined with $2n$-bit tag generation function using two n-bit hash functions, and an ivE scheme with $2n$-bit IV using two blockciphers. We remark that SCT [PS16] has also BBB security, however it is based on a tweakable blockcipher, a more powerful primitive than a plain blockcipher, and BBB security is achieved against nonce-respective adversaries.

We note that T in GCM-SIV2 is now $2n$ bits, the set of keys is $(\mathcal{K}_H)^2 \times (\mathcal{K}_E)^4 \times (\mathcal{K}_E)^2$, and we have $\boldsymbol{K} = (L_1, L_2, K'_1, K'_2, K'_3, K'_4, K_1, K_2) \in (\mathcal{K}_H)^2 \times (\mathcal{K}_E)^4 \times (\mathcal{K}_E)^2$.

Security Bound of GCM-SIV2. We focus on the information-theoretic security, namely all blockciphers are assumed to be independent uniform random permutations. To obtain computationally-secure counterpart, a standard technique can be applied [BDJR97], and the provable security results hold under the assumption that the blockcipher is a pseudorandom permutation.

Let $\mathsf{F2}_{L_1,L_2,P'_1,P'_2,P'_3,P'_4}$, which we abbreviate as $\mathsf{F2}$, be the tag generation function of GCM-SIV2 that takes (N, A, M) as input and outputs $(T[1], T[2])$. It is obtained by extracting lines from 1 to 4 of GCM-SIV2-$\mathcal{E}^{N,A}_{\boldsymbol{K}}(M)$ from Fig. 4, and replacing $E_{K'_1}, \ldots, E_{K'_4}$ with independent random permutations $P'_1, \ldots, P'_4$. We first present a lemma showing that $\mathsf{F2}$ is a secure PRF up to about $2^{2n/3}$ query complexity.

Algorithm GCM-SIV2-$\mathcal{E}_{\boldsymbol{K}}^{N,A}(M)$

1. $V[1] \leftarrow H_{L_1}(N, A, M)$
2. $V[2] \leftarrow H_{L_2}(N, A, M)$
3. $T[1] \leftarrow E_{K'_1}(V[1]) \oplus E_{K'_3}(V[2])$
4. $T[2] \leftarrow E_{K'_2}(V[1]) \oplus E_{K'_4}(V[2])$
5. $\mathbf{S}[1] \leftarrow \mathsf{CTR}_{K_1}(T[1], |M|_n)$
6. $\mathbf{S}[2] \leftarrow \mathsf{CTR}_{K_2}(T[2], |M|_n)$
7. $C \leftarrow M \oplus \mathsf{msb}_{|M|}(\mathbf{S}[1]) \oplus \mathsf{msb}_{|M|}(\mathbf{S}[2])$
8. $T \leftarrow T[1] \,\|\, T[2]$
9. **return** (C, T)

Algorithm GCM-SIV2-$\mathcal{D}_{\boldsymbol{K}}^{N,A}(C, T)$

1. $\mathbf{S}[1] \leftarrow \mathsf{CTR}_{K_1}(T[1], |C|_n)$
2. $\mathbf{S}[2] \leftarrow \mathsf{CTR}_{K_2}(T[2], |C|_n)$
3. $M \leftarrow C \oplus \mathsf{msb}_{|M|}(\mathbf{S}[1]) \oplus \mathsf{msb}_{|M|}(\mathbf{S}[2])$
4. $V[1] \leftarrow H_{L_1}(N, A, M)$
5. $V[2] \leftarrow H_{L_2}(N, A, M)$
6. $T[1] \leftarrow E_{K'_1}(V[1]) \oplus E_{K'_3}(V[2])$
7. $T[2] \leftarrow E_{K'_2}(V[1]) \oplus E_{K'_4}(V[2])$
8. $T^* \leftarrow T[1] \,\|\, T[2]$
9. **if** $T \neq T^*$ **then return** $\perp$
10. **return** M

Figure 4: Definitions of GCM-SIV2-$\mathcal{E}_{\boldsymbol{K}}^{N,A}(M)$ and GCM-SIV2-$\mathcal{D}_{\boldsymbol{K}}^{N,A}(C, T)$

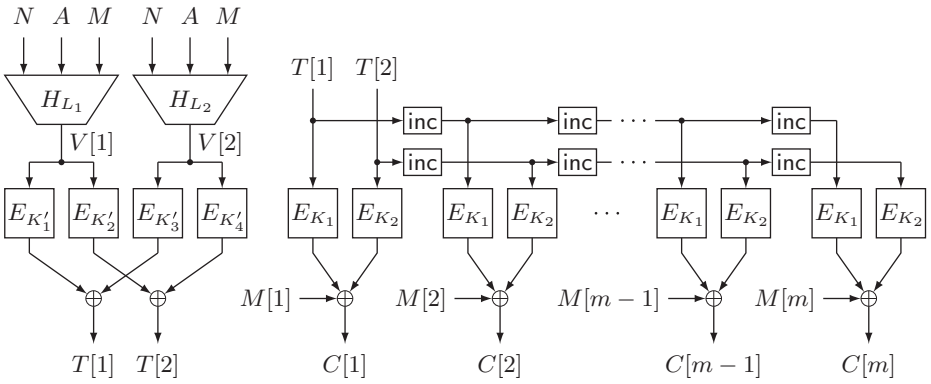

Figure 5: The encryption algorithm of GCM-SIV2

Lemma 2. *Let $\mathcal{A}$ be an adversary that makes q queries with the maximum block length ℓ. If H_{L_1} and H_{L_2} are ϵ-AU for any maximum input block length ℓ, we have*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F2}}(\mathcal{A}) \leq \frac{8q^3}{3 \cdot 2^{2n}} + 6\epsilon^2 q^3 \tag{5}$$

when $q \leq 2^{n-1}$. In particular, when H_{L_i} is defined as $H_{L_i}(N, A, M) = \mathsf{GHASH}_{L_i}(A, M) \oplus N$, it is an $\ell/2^n$-AU hash function, and we have $\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F2}}(\mathcal{A}) \leq 8.7\ell^2 q^3/2^{2n}$.

A proof is obtained from a proof that covers a more general case, which is presented in Sect. 9. Here, we briefly show the intuition of the proof. Let $(N_1, A_1, M_1), \ldots, (N_q, A_q, M_q)$ be the queries made by $\mathcal{A}$, and let $(V_1[1], V_1[2]), \ldots, (V_q[1], V_q[2])$ be the corresponding output values of H_{L_1} and H_{L_2}. For each $2 \leq i \leq q$, we make the following cases.

- Case $V_i[1] \notin \{V_1[1], \ldots, V_{i-1}[1]\}$ and $V_i[2] \notin \{V_1[2], \ldots, V_{i-1}[2]\}$. In this case, we follow the analysis of [Yas10, Luc00] and see if the xor of two output values of two independent random permutations is uniformly random.
- Case $V_i[1] \notin \{V_1[1], \ldots, V_{i-1}[1]\}$ and $V_i[2] \in \{V_1[2], \ldots, V_{i-1}[2]\}$. In this case, we rely on the randomness of P'_1 and P'_2 and see that $P'_1(V_i[1])$ and $P'_2(V_i[1])$ are random.

- Case $V_i[1] \in \{V_1[1], \ldots, V_{i-1}[1]\}$ and $V_i[2] \notin \{V_1[2], \ldots, V_{i-1}[2]\}$. This case is similar to the above, and we rely on the randomness of P'_3 and P'_4.
- Case $V_i[1] \in \{V_1[1], \ldots, V_{i-1}[1]\}$ and $V_i[2] \in \{V_1[2], \ldots, V_{i-1}[2]\}$. This case is a bad event, and we assume that the adversary succeeds in the attack.

We remark that the bound of F2 using GHASH is better than that of SUM-ECBC presented by Yasuda [Yas10], since the security bound of SUM-ECBC is $O(\ell^4 q^3/2^{2n})$, while that of F2 is $O(\ell^2 q^3/2^{2n})$. This is simply due to the use of a polynomial hash function (which is $\ell/2^n$-AU) instead of CBC-MAC (which is $\ell^2/2^n$- or $\ell^4/2^{2n}$-AU [BR00, BPR05]) for message hashing. We also remark that there are hash functions where the collision probability is independent of the input length, see e.g. [LPTY16].

We next define the ivE scheme of GCM-SIV2, which we write $\mathsf{E2}_{P_1,P_2}$. It takes $2n$-bit IV $(T[1], T[2])$ as input, and outputs the key stream $\mathbf{S} = \mathbf{S}[1] \oplus \mathbf{S}[2]$, and encrypts plaintext M as $C = \mathbf{S} \oplus M$, using two random permutations P_1 and P_2. Here we often abbreviate $\mathsf{E2}_{P_1,P_2}$ as E2. It is obtained by extracting lines from 5 to 8 of GCM-SIV2-$\mathcal{E}_{\boldsymbol{K}}^{N,A}(M)$ from Fig. 4 and replacing E_{K_i} with P_i.

Lemma 3. *For any adversary $\mathcal{A}$ that makes q queries, where the total number of blocks of the queries is at most σ blocks, we have*

$$\mathbf{Adv}^{\mathrm{priv\$}}_{\mathsf{E2}}(\mathcal{A}) \leq \frac{13\sigma^3}{3 \cdot 2^{2n}}. \tag{6}$$

A proof is obtained from a proof in Sect. 9 that covers a more general case. Here, we briefly point out that most of the analysis of F2 can be used for the analysis of E2 by treating $(T[1], T[2])$ in E2 as $(V[1], V[2])$ in F2. By combining Lemma 1, Lemma 2, and Lemma 3, we obtain the following security bound of GCM-SIV2.

Theorem 1. *For any (q, ℓ, σ)-adversary $\mathcal{A}$, we have*

$$\mathbf{Adv}^{\mathrm{mrae}}_{\text{GCM-SIV2}}(\mathcal{A}) \leq \frac{7\sigma^3}{2^{2n}} + 6\epsilon^2 q^3 + \frac{q}{2^{2n}}. \tag{7}$$

From Theorem 1, when F2 uses two independently-keyed GHASH, we have $\epsilon = \ell/2^n$, and the bound becomes

$$\frac{7\sigma^3}{2^{2n}} + \frac{6\ell^2 q^3}{2^{2n}} + \frac{q}{2^{2n}}. \tag{8}$$

This bound shows that GCM-SIV2 is secure up to about $2^{2n/3}$ query complexity.

8 GCM-SIVr: Generalization

GCM-SIV2 achieves BBB security. However, there is still a gap from the optimal (i.e. n-bit) security bound. To fill the gap, we put forward the idea of using multiple instances of GCM-SIV1 more than two instances. The scheme is naturally defined as GCM-SIVr, where $r \geq 2$ denotes the number of instances, and by setting $r = 2$ it is exactly reduced to GCM-SIV2.

A pseudocode of GCM-SIVr is shown in Fig. 6, and the encryption function for $r = 3$ is illustrated in Fig. 7. The tag length of GCM-SIVr is rn bits, and its key is $\boldsymbol{K} = (L_1, \ldots, L_r, K'_1, \ldots, K'_{r^2}, K_1, \ldots, K_r)$, where $\boldsymbol{K} \in (\mathcal{K}_H)^r \times (\mathcal{K}_E)^{r^2} \times (\mathcal{K}_E)^r$.

The information-theoretic security bound of GCM-SIVr is described as follows.

Theorem 2. *For any (q, ℓ, σ)-adversary $\mathcal{A}$, we have*

$$\mathbf{Adv}^{\mathrm{mrae}}_{\text{GCM-SIV}r}(\mathcal{A}) \leq r \cdot (4\epsilon)^r \cdot q^{r+1} + \frac{4^r \cdot \sigma^{r+1}}{2^{nr}} + \frac{q}{2^{nr}}. \tag{9}$$

Algorithm GCM-SIVr-$\mathcal{E}_{\boldsymbol{K}}^{N,A}(M)$

1. **for** $i = 1$ **to** r **do**
2. $\quad V[i] \leftarrow H_{L_i}(N, A, M)$
3. $\quad T[i] \leftarrow 0^n$
4. **for** $i = 1$ **to** r **do**
5. $\quad$ **for** $j = 1$ **to** r **do**
6. $\quad\quad T[i] \leftarrow T[i] \oplus E_{K'_{i+r(j-1)}}(V[j])$
7. **for** $i = 1$ **to** r **do**
8. $\quad \mathbf{S}[i] \leftarrow \mathsf{CTR}_{K_i}(T[i], |M|_n)$
9. $\quad M \leftarrow M \oplus \mathsf{msb}_{|M|}(\mathbf{S}[i])$
10. $C \leftarrow M$
11. $T \leftarrow T[1] \,\|\, T[2] \,\|\, \cdots \,\|\, T[r]$
12. **return** (C, T)

Algorithm GCM-SIVr-$\mathcal{D}_{\boldsymbol{K}}^{N,A}(C, T)$

1. **for** $i = 1$ **to** r **do**
2. $\quad \mathbf{S}[i] \leftarrow \mathsf{CTR}_{K_i}(T[i], |C|_n)$
3. $\quad C \leftarrow C \oplus \mathsf{msb}_{|C|}(\mathbf{S}[i])$
4. $M \leftarrow C$
5. **for** $i = 1$ **to** r **do**
6. $\quad V[i] \leftarrow H_{L_i}(N, A, M)$
7. $\quad T[i] \leftarrow 0^n$
8. **for** $i = 1$ **to** r **do**
9. $\quad$ **for** $j = 1$ **to** r **do**
10. $\quad\quad T[i] \leftarrow T[i] \oplus E_{K'_{i+r(j-1)}}(V[j])$
11. $T^* \leftarrow T[1] \,\|\, T[2] \,\|\, \cdots \,\|\, T[r]$
12. **if** $T \neq T^*$ **then return** $\perp$
13. **return** M

Figure 6: Definitions of GCM-SIVr-$\mathcal{E}_{\boldsymbol{K}}^{N,A}(M)$ and GCM-SIVr-$\mathcal{D}_{\boldsymbol{K}}^{N,A}(C, T)$

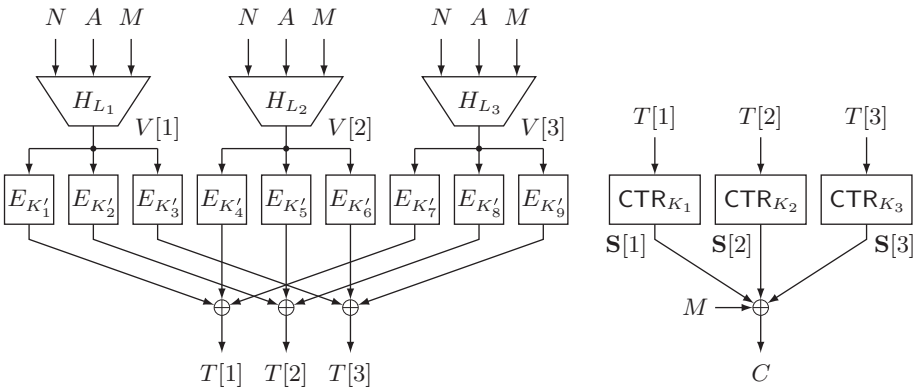

Figure 7: The encryption algorithm of GCM-SIVr for $r = 3$

If $H_{L_1}, \ldots, H_{L_r}$ used in GCM-SIVr are instantiated with independently-keyed GHASH, then the bound is

$$\frac{r \cdot (4\ell)^r \cdot q^{r+1}}{2^{nr}} + \frac{4^r \cdot \sigma^{r+1}}{2^{nr}} + \frac{q}{2^{nr}},$$

and this shows that GCM-SIVr is secure up to about $2^{rn/(r+1)}$ query complexity, and hence it asymptotically achieves the optimal security. We remark that this type of $rn/(r+1)$-bit security bound has been observed in various types of provably-secure constructions, such as [CS14, Luc00, Mau02, MP03].

9 Security Proofs of GCM-SIV2 and GCM-SIV*r*

In this section, we present the security proofs of GCM-SIVr, where the proof of GCM-SIV2 is obtained by setting $r = 2$.

Algorithm $\mathsf{F}r_{L_1,\ldots,L_r,P'_1,\ldots,P'_{r^2}}(N,A,M)$

1. **for** $i = 1$ **to** r **do**
2. $\quad V[i] \leftarrow H_{L_i}(N,A,M)$
3. $\quad$ **for** $j = 1$ **to** r **do**
4. $\quad\quad$ **if** $V[i] \notin \mathrm{Dom}(P'_{r(i-1)+j})$ **then** $x_{i,j} = 0$
5. $\quad\quad$ **else** $x_{i,j} = 1$
6. $\quad$ **end for**
7. **end for**
8. **for** $j = 1$ **to** r **do**
9. $\quad X_j \leftarrow (x_{1,j},\ldots,x_{r,j})$
10. $\quad$ **go to Case** X_j // this generates $T[j]$
11. **end for**
12. $T \leftarrow (T[1],\ldots,T[r])$
13. **return** T

Figure 8: Main Game of $\mathsf{F}r$, the PRF part of GCM-SIVr

9.1 Proving the PRF Bound of Fr

We first prove the security of the tag generation function of GCM-SIVr, which we call $\mathsf{F}r$. Let $\mathsf{F}r_{L_1,\ldots,L_r,P'_1,\ldots,P'_{r^2}} = \mathbf{P}'^{(r,r)} \circ \mathbf{H}^{(r)}$, where

$$\begin{cases} \mathbf{H}^{(a)}(N,A,M) = (H_{L_1}(N,A,M),\ldots,H_{L_a}(N,A,M)), \\ \mathbf{P}'^{(a,b)}(V[1],\ldots,V[a]) = \left(\bigoplus_{j=1}^{b} P'_{b(j-1)+1}(V[j]),\ldots,\bigoplus_{j=1}^{b} P'_{b(j-1)+a}(V[j])\right). \end{cases}$$

This corresponds to the tag generation function of GCM-SIVr using random permutations $P'_1,\ldots,P'_{r^2}$.

Let $\mathcal{R}$ be the uniform random function taking (N,A,M) as input and generating rn-bit output. We need to bound the PRF-advantage of $\mathsf{F}r$, which is $\Pr[\mathcal{A}^{\mathsf{F}r} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{R}} \Rightarrow 1]$.

We employ the Game-playing technique by Bellare and Rogaway [BR06]. Both $\mathsf{F}r$ and $\mathcal{R}$ are implemented as games, called real and ideal games, where the internal random permutations $P'_1,\ldots,P'_{r^2}$ are implemented by lazy sampling. For each P'_i, the games implicitly maintain two sets, $\mathrm{Dom}(P'_i)$ and $\mathrm{Rng}(P'_i)$, which keep the record of domain and range points that are already determined. Formally, if we have $x \in \{0,1\}^n$ which is not in $\mathrm{Dom}(P'_i)$ then we randomly sample y as $y \xleftarrow{\$} \{0,1\}^n \setminus \mathrm{Rng}(P'_i)$, and determine $y = P'_i(x)$, and then add x to $\mathrm{Dom}(P'_i)$ and y to $\mathrm{Rng}(P'_i)$.

The proofs are based on the proofs of SUM-ECBC by Yasuda [Yas10] and SUM construction by Lucks [Luc00]. We define the main game in Fig. 8 which has a number of cases (subroutines), depending on the input collisions on random permutations. Each case is specified by an r-bit variable X_j, where $j = 1,\ldots,r$, and **Case** X_j determines how $T[j]$ is produced. Let us describe the intuition how $X_j = (x_{1,j},\ldots,x_{r,j})$ works. We have $T[j] = P'_j(V[1]) \oplus P'_{r+j}(V[2]) \oplus \cdots \oplus P'_{r(r-1)+j}(V[r])$, and if the i-th bit $x_{i,j}$ of X_j is 1, where $1 \le i \le r$, then this implies that $V[i]$ was used in the previous query (and hence we detect a collision of H_{L_i}). If the i-th bit of X_j is 0, then this implies that $V[i]$ was never used previously. Observe that we have $X_1 = \cdots = X_r$.

We then describe the overview of the security proof. The proof is divided into **Case** $X_j = (0,\ldots,0)$, **Case** $X_j = (1,\ldots,1,0,\ldots,0)$, and **Case** $X_j = (1\ldots,1)$. Let $wt(X_j)$ denote the hamming weight of X_j.

- **Case** $X_j = (0,\ldots,0)$ is the case where $wt(X_j) = 0$, and all the input values of the random permutations are "new." We follow the fair set analysis of [Yas10, Luc00] to

see if the xor of r output values of r independent random permutations is uniformly random. We make further cases depending on the value of r.

- **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$ is divided into cases depending on $wt(X_j) = p$. We here only consider the case where X_j is of the form $X_j = (1, \ldots, 1, 0, \ldots, 0)$, i.e., the first p bits of X_j is 1. Due to the symmetry of the construction, other cases of weight p have the same probability. We have three cases, $1 \leq p \leq r-2$ and $r-p$ is even, $1 \leq p \leq r-2$ and $r-p$ is odd, and $p = r-1$. If $1 \leq p \leq r-2$, we apply the fair set analysis of [Yas10, Luc00] to the last $r-p$ random permutations, and see if the output yields a random value. We make further cases depending on the value of $r-p$, the number of random permutations with new input values. If $wt(X_j) = r-1$, then our analysis is based on the randomness of the single remaining random permutation.
- **Case** $X_j = (1, \ldots, 1)$, which corresponds to $wt(X_j) = r$, is considered to be a bad event, and we assume that the adversary succeeds in the attack.

The details of the proof now follow.

- For clarity we treat **Case** $X_j = (0, \ldots, 0)$ and **Case** $X_j = (1, \ldots, 1)$ as special cases. They are written in Fig. 9 and Fig. 11.
- Other cases, **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$, where $1 \leq wt(X_j) \leq r-1$, is shown in Fig. 10.
- As stated above, we show the case where X_j is of the form $X_j = (1, \ldots, 1, 0, \ldots, 0)$, i.e., the first p bits are 1, since the evaluation of other cases is the same from the symmetry.

The following proposition, which relies on the idea of *resampling* by Bellare and Rogaway [BR06], shows that Fig. 8 implements the real and ideal games.

Proposition 1. *When r is even, $\mathsf{F}r$ is implemented with the main game of Fig. 8 taking* **Case** $X_j = (0, \ldots, 0)$ *without the boxed argument,* **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$ *with the boxed argument, and* **Case** $X_j = (1, \ldots, 1)$ *without the boxed argument. In addition, $\mathcal{R}$ is implemented by taking* **Case** $X_j = (0, \ldots, 0)$ *with the boxed argument,* **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$ *without the boxed argument, and* **Case** $X_j = (1, \ldots, 1)$ *with the boxed argument.*

When r is odd, $\mathsf{F}r$ is implemented taking **Case** $X_j = (0, \ldots, 0)$ *with the boxed argument,* **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$ *with the boxed argument, and* **Case** $X_j = (1, \ldots, 1)$ *without the boxed argument. Finally, $\mathcal{R}$ is implemented by taking* **Case** $X_j = (0, \ldots, 0)$ *without the boxed argument,* **Case** $X_j = (1, \ldots, 1, 0, \ldots, 0)$ *without the boxed argument, and* **Case** $X_j = (1, \ldots, 1)$ *with the boxed argument.*

We will describe how Proposition 1 is verified in the subsequent analysis of **Case** X_j. We observe that the difference between the games, i.e. real ($\mathsf{F}r$) and ideal ($\mathcal{R}$) games, is always seen after the **bad** flag is set. This shows that the two games are equivalent until the **bad** flag is set. From the fundamental lemma of [BR06], we obtain

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}r}(\mathcal{A}) \leq \Pr[\mathcal{A}^{\mathcal{R}} \text{ sets } \mathbf{bad}] \leq \sum_{j=1}^{r} \sum_{X_j \in \{0,1\}^r} \Pr[\mathcal{A}^{\mathcal{R}} \text{ sets } \mathbf{bad} \text{ at } \mathbf{Case}\ X_j]. \qquad (10)$$

We next evaluate $\Pr[\mathcal{A}^{\mathcal{R}}$ sets **bad** at **Case** $X_j]$, which we abbreviate as $\Pr[\mathbf{bad}(X_j)]$. Since $\mathcal{A}^{\mathcal{R}}$ always receives uniform random output, any adaptive choice of inputs does not increase the chance of bad events (the same argument as [IY09a]). Therefore we focus on the non-adaptive strategies.

Case $X_j = (0, \ldots, 0)$. Our analysis follows [Luc00, Yas10]. This case first determines $Y^{(r)} \subseteq (\{0,1\}^n)^r$, which denotes the set of all possible r-tuple of outputs of r permutations in the real game. Then we use the notion of the fair set [Luc00]. Here, we say a set $S \subseteq (\{0,1\}^n)^r$ is fair if for any $z \in \{0,1\}^n$, we have

$$|\{(x_1, \ldots, x_r) \in S \mid x_1 \oplus \cdots \oplus x_r = z\}| = \frac{|S|}{2^n}.$$

Then, [Luc00] pointed out that, when r is even, there exists a set $\mathcal{C} \subset Y^{(r)}$ of size i^r such that $U_j = Y^{(r)} \setminus \mathcal{C}$ is a fair set, where i denotes the number of queries done so far. Similarly, when r is odd, there exists a set $\mathcal{C}'$ of size i^r with $\mathcal{C}' \cap Y^{(r)} = \emptyset$ such that $U_j = Y^{(r)} \cup \mathcal{C}'$ is a fair set. See Lemma 2 of [Luc00] or [Yas10] for explicit constructions of fair sets.

When r is even and without the boxed argument, line 6 of Fig. 9 ensures that y is uniformly random over $Y^{(r)}$. With the boxed argument and given $y \notin U_j$ at line 7, y is uniformly random over $\{0,1\}^n$ by the definition of the fair set. Therefore, Proposition 1 holds in this case. When r is odd, we have a similar analysis, but the boxed argument is then required to make sure that y is always uniform over $Y^{(r)}$ to implement the real game.

Based on these observations, when r is even, we have

$$\Pr[\mathbf{bad}(X_j)] \le \sum_{i=0}^{q-1} \frac{|Y^{(r)} \setminus U_j|}{|Y^{(r)}|} \le \sum_{i=0}^{q-1} \frac{i^r}{(2^n - q)^r} \le \frac{2^r}{2^{nr}} \sum_{i=0}^{q-1} i^r.$$

Similarly, when r is odd, we have the same bound since

$$\Pr[\mathbf{bad}(X_j)] \le \sum_{i=0}^{q-1} \frac{|U_j \setminus Y^{(r)}|}{|U_j|} < \sum_{i=0}^{q-1} \frac{i^r}{|Y^{(r)}|} \le \sum_{i=0}^{q-1} \frac{i^r}{(2^n - q)^r} \le \frac{2^r}{2^{nr}} \sum_{i=0}^{q-1} i^r. \tag{11}$$

Here the last term is bounded by $2^r(q-1)^{r+1}/2^{nr}$, since we have

$$\sum_{i=0}^{x} i^r \le x^{r+1} \text{ for } x \ge 0 \text{ and } r \ge 1. \tag{12}$$

Case $X_j = (1, \ldots, 1, 0, \ldots, 0)$. Let $p = wt(X_j)$ be the hamming weight of X_j. Without loss of generality, we assume that the first p bits of X_j are 1, and the game is shown in Fig. 10. Let $\bar{p} = r - p$ be the number of 0s.

At this point, we define an event which we write $\mathrm{VColl}(i, j_1, \ldots, j_p)$. Let $V_i[j]$ denote the value of $V[j]$ generated at the i-th query, i.e., $V_i[j] = H_{L_j}(N_i, A_i, M_i)$ where (N_i, A_i, M_i) denotes the i-th query. We define $\mathrm{VColl}(i, j_1, \ldots, j_p)$ as the event $(V_i[1] = V_{j_1}[1]) \wedge (V_i[2] = V_{j_2}[2]) \wedge \cdots \wedge (V_i[p] = V_{j_p}[p])$. We have

$$\begin{aligned}
\Pr[\mathbf{bad}(X_j)] &\le \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=1}^{i-1} \Pr[\mathrm{VColl}(i, j_1, \ldots, j_p) \wedge \mathbf{bad}(X_j)] \\
&\le \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=1}^{i-1} \Pr[\mathrm{VColl}(i, j_1, \ldots, j_p)] \cdot \Pr[\mathbf{bad}(X_j) \mid \mathrm{VColl}(i, j_1, \ldots, j_p)] \\
&\le \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=1}^{i-1} \epsilon^p \cdot \Pr[\mathbf{bad}(X_j) \mid \mathrm{VColl}(i, j_1, \ldots, j_p)]. \qquad (13)
\end{aligned}$$

Let us fix $i, j_1, \ldots, j_p$, and we analyze the last term of (13). We divide the analysis into three cases, $\bar{p}$ is 1, or $\bar{p}$ is a larger odd integer, or $\bar{p}$ is an even integer.

Case $X_j = (0, \ldots, 0)$ with even r

1. **for** $i = 1$ **to** r **do**
2. $\quad Y(i) \leftarrow \{0,1\}^n \setminus \mathrm{Rng}(P'_{r(i-1)+j})$
3. **end for**
4. $Y^{(r)} \leftarrow Y(1) \times \cdots \times Y(r)$
5. Choose a fair set $U_j \subset Y^{(r)}$ // see [Luc00] and texts
6. $y \leftarrow (y(1), \ldots, y(r)) \xleftarrow{\$} Y^{(r)}$
7. **if** $y \notin U_j$
8. $\quad$ **bad** $\leftarrow$ **true** $\boxed{y \leftarrow (y(1), \ldots, y(r)) \xleftarrow{\$} U_j}$
9. **end if**
10. $T[j] \leftarrow y(1) \oplus y(2) \oplus \cdots \oplus y(r)$
11. **return** $T[j]$

Case $X_j = (0, \ldots, 0)$ with odd r

1. **for** $i = 1$ **to** r **do**
2. $\quad Y(i) \leftarrow \{0,1\}^n \setminus \mathrm{Rng}(P'_{r(i-1)+j})$
3. **end for**
4. $Y^{(r)} \leftarrow Y(1) \times \cdots \times Y(r)$
5. Choose a fair set $U_j \supset Y^{(r)}$
6. $y \leftarrow (y(1), \ldots, y(r)) \xleftarrow{\$} U_j$
7. **if** $y \notin Y^{(r)}$
8. $\quad$ **bad** $\leftarrow$ **true** $\boxed{y \leftarrow (y(1), \ldots, y(r)) \xleftarrow{\$} Y^{(r)}}$
9. **end if**
10. $T[j] \leftarrow y(1) \oplus y(2) \oplus \cdots \oplus y(r)$
11. **return** $T[j]$

Figure 9: Case $X_j = (0, \ldots, 0)$. When r is even, the boxed argument is only for the ideal game, and when r is odd, the boxed argument is only for the real game.

If $\bar{p} = 1$, we have $p = r - 1$, and $\mathsf{F}r$ is implemented with the boxed argument, and $\mathcal{R}$ is implemented without the boxed argument. The probability of the bad event is bounded as

$$\Pr[\mathbf{bad}(X_j) \mid \mathrm{VColl}(i, j_1, \ldots, j_p)] \leq \frac{q}{2^n}, \tag{14}$$

since **bad** is set when a random n-bit value is in the set $\mathrm{Rng}(P'_{r(r-1)+j})$.

If $\bar{p} > 1$, the game first determines $Y^{(r-p)}$, and performs a fair set-based sampling. The correctness of Proposition 1 can be verified in the same manner to **Case** $X_j = (0, \ldots, 0)$.

If $\bar{p}$ is even, we have

$$\Pr[\mathbf{bad}(X_j) \mid \mathrm{VColl}(i, j_1, \ldots, j_p)] \leq \frac{|Y^{(r-p)} \setminus U_j|}{|Y^{(r-p)}|} \leq \frac{i^{r-p}}{(2^n - q)^{r-p}} \leq \frac{2^{r-p}}{2^{n(r-p)}} i^{r-p}. \tag{15}$$

When $\bar{p}$ is odd and larger than 1, we similarly have

$$\begin{aligned} \Pr[\mathbf{bad}(X_j) \mid \mathrm{VColl}(i, j_1, \ldots, j_p)] &\leq \frac{|U_j \setminus Y^{(r-p)}|}{|U_j|} < \frac{i^{r-p}}{|Y^{(r-p)}|} \leq \frac{i^{r-p}}{(2^n - q)^{r-p}} \\ &\leq \frac{2^{r-p}}{2^{n(r-p)}} i^{r-p}. \end{aligned} \tag{16}$$

Case $X_j = (1, \ldots, 1, 0)$ with $p = r - 1$ and $\bar{p} = r - p = 1$

1. **for** $i = 1$ **to** p **do**
2. $\quad y(i) \leftarrow P'_{r(i-1)+j}(V[i])$
3. **end for**
4. $y(r) \xleftarrow{\$} \{0,1\}^n$
5. **if** $y(r) \in \mathrm{Rng}(P'_{r(r-1)+j})$
6. $\quad$ **bad** $\leftarrow$ **true** $\boxed{y(r) \xleftarrow{\$} \{0,1\}^n \setminus \mathrm{Rng}(P'_{r(r-1)+j})}$
7. **end if**
8. $T[j] \leftarrow y(1) \oplus \cdots \oplus y(r)$
9. **return** $T[j]$

Case $X_j = (1, \ldots, 1, 0, \ldots, 0)$ with $1 \le p < r - 1$, $\bar{p} = r - p$, and even $\bar{p}$

1. **for** $i = 1$ **to** p **do**
2. $\quad y(i) \leftarrow P'_{r(i-1)+j}(V[i])$
3. **end for**
4. **for** $i = p + 1$ **to** r **do**
5. $\quad Y(i) \leftarrow \{0,1\}^n \setminus \mathrm{Rng}(P'_{r(i-1)+j})$
6. **end for**
7. $Y^{(r-p)} \leftarrow Y(p+1) \times \cdots \times Y(r)$
8. Choose a fair set $U_j \subset Y^{(r-p)}$
9. $y \leftarrow (y(p+1), \ldots, y(r)) \xleftarrow{\$} Y^{(r-p)}$
10. **if** $y \notin U_j$
11. $\quad$ **bad** $\leftarrow$ **true** $\boxed{y \leftarrow (y(p+1), \ldots, y(r)) \xleftarrow{\$} U_j}$
12. **end if**
13. $T[j] \leftarrow y(1) \oplus \cdots \oplus y(r)$
14. **return** $T[j]$

Case $X_j = (1, \ldots, 1, 0, \ldots, 0)$ with $1 \le p < r - 1$, $\bar{p} = r - p$, and odd $\bar{p}$

1. **for** $i = 1$ **to** p **do**
2. $\quad y(i) \leftarrow P'_{r(i-1)+j}(V[i])$
3. **end for**
4. **for** $i = p + 1$ **to** r **do**
5. $\quad Y(i) \leftarrow \{0,1\}^n \setminus \mathrm{Rng}(P'_{r(i-1)+j})$
6. **end for**
7. $Y^{(r-p)} \leftarrow Y(p+1) \times \cdots \times Y(r)$
8. Choose a fair set $U_j \supset Y^{(r-p)}$
9. $y \leftarrow (y(p+1), \ldots, y(r)) \xleftarrow{\$} U_j$
10. **if** $y \notin Y^{(r-p)}$
11. $\quad$ **bad** $\leftarrow$ **true** $\boxed{y \leftarrow (y(p+1), \ldots, y(r)) \xleftarrow{\$} Y^{(r-p)}}$
12. **end if**
13. $T[j] \leftarrow y(1) \oplus \cdots \oplus y(r)$
14. **return** $T[j]$

Figure 10: Case $X_j = (1, \ldots, 1, 0, \ldots, 0)$ with $wt(X_j) = p$ and $1 \le p \le r - 1$. When $\bar{p} = r - p$ is even, the boxed argument is only for the ideal game. When $\bar{p}$ is odd (including 1), the boxed arguments are only for the real game.

Case $X_j = (1, \ldots, 1)$

1. **bad** $\leftarrow$ **true** $\boxed{T[j] \xleftarrow{\$} \{0,1\}^n, \textbf{return } T[j]}$
2. **for** $i = 1$ **to** r **do**
3. $\quad y(i) \leftarrow P'_{r(i-1)+j}(V[i])$
4. **end for**
5. $T[j] \leftarrow y(1) \oplus \cdots \oplus y(r)$
6. **return** $T[j]$

Figure 11: Case $X_j = (1, \ldots, 1)$. This case immediately sets **bad**. The boxed argument is only for the ideal game.

Now we return to the evaluation of $\Pr[\mathbf{bad}(X_j)]$ by using (14), (15), and (16). When $\bar{p} = 1$, $\Pr[\mathbf{bad}(X_j)]$ is bounded by

$$\sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=1}^{i-1} \epsilon^p \cdot \frac{q}{2^n} \leq \sum_{i=2}^{q} \sum_{j_1=1}^{q-1} \cdots \sum_{j_p=1}^{q-1} \epsilon^p \cdot \frac{q}{2^n} \leq q^{p+1} \cdot \epsilon^p \cdot \frac{q}{2^n} \leq \frac{\epsilon^{r-1} \cdot q^{r+1}}{2^n}, \tag{17}$$

and when $\bar{p} > 1$, using (12) we obtain

$$\begin{aligned} \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_p=1}^{i-1} \epsilon^p \cdot \frac{2^{r-p}}{2^{n(r-p)}} \cdot i^{r-p} &\leq \sum_{i=2}^{q} \sum_{j_1=1}^{q-1} \cdots \sum_{j_p=1}^{q-1} \epsilon^p \cdot \frac{2^{r-p}}{2^{n(r-p)}} \cdot i^{r-p} \\ &\leq q^p \cdot \epsilon^p \cdot \frac{2^{r-p}}{2^{n(r-p)}} \cdot \sum_{i=2}^{q} i^{r-p} \leq \frac{2^{r-p} \epsilon^p q^{r+1}}{2^{n(r-p)}}. \end{aligned} \tag{18}$$

The last term is bounded by $2^r \epsilon^r q^{r+1}$ from $1/2^{n(r-p)} \leq \epsilon^{r-p}$.

Case $X_j = (1, \ldots, 1)$. In this case, it is easy to verify the correctness of Proposition 1, and we have

$$\begin{aligned} \Pr[\mathbf{bad}(X_j)] = \Pr[\mathcal{A}^{\mathcal{R}} \text{ enters } \mathbf{Case}\ X_j] &\leq \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_r=1}^{i-1} \Pr[\mathrm{VColl}(i, j_1, \ldots, j_r)] \\ &\leq \sum_{i=2}^{q} \sum_{j_1=1}^{i-1} \cdots \sum_{j_r=1}^{i-1} \epsilon^r \leq \sum_{i=2}^{q} \sum_{j_1=1}^{q-1} \cdots \sum_{j_r=1}^{q-1} \epsilon^r \leq q^{r+1} \epsilon^r. \end{aligned} \tag{19}$$

Taking the Sum. The above analysis holds for any $j = 1, \ldots, r$, and we observe that the probability of the bad event of **Case** X_j shown above is determined solely depending on $wt(X_j) = p$. Therefore, we write $f_{\mathrm{bad}}(p)$ to denote $\Pr[\mathcal{A}^{\mathcal{R}}$ sets **bad** at **Case** $X_j]$ for any X_j of weight $0 \leq p \leq r$, $j = 1, \ldots, r$. Here, $f_{\mathrm{bad}}(0)$ is given as (11), $f_{\mathrm{bad}}(r)$ is (19), $f_{\mathrm{bad}}(r-1)$ is (17), and $f_{\mathrm{bad}}(p)$ for $1 \leq p \leq r-2$ is given as (18). Since $f_{\mathrm{bad}}(p) \leq (2\epsilon)^r \cdot q^{r+1}$ for any $0 \leq p \leq r$, from (10), we obtain

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}r}(\mathcal{A}) \leq r \cdot \left[\sum_{p=0}^{r} \binom{r}{p} f_{\mathrm{bad}}(p) \right] \leq r \cdot 2^r \max_p \{ f_{\mathrm{bad}}(p) \} \leq r \cdot (4\epsilon)^r \cdot q^{r+1}, \tag{20}$$

which is $r \cdot (4\ell)^r \cdot q^{r+1}/2^{nr}$ if $\epsilon = \ell/2^n$. We remark that for given r, a slightly better bound can be obtained. From (11), (17), (18), and (19), we have

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F}r}(\mathcal{A}) \leq r \cdot \left[f_{\mathrm{bad}}(0) + f_{\mathrm{bad}}(r) + r f_{\mathrm{bad}}(r-1) + \sum_{p=1}^{r-2} \frac{r^p}{p} f_{\mathrm{bad}}(p) \right]$$

$$\leq r \cdot \left[\frac{2^r}{2^{nr}} \sum_{i=0}^{q-1} i^r + \epsilon^r \cdot q^{r+1} + \frac{r \cdot q^{r+1} \cdot \epsilon^{r-1}}{2^n} + \sum_{p=1}^{r-2} \frac{r^p}{p} f_{\mathrm{bad}}(p) \right]$$

$$\leq r \cdot \left[\frac{2^r}{2^{nr}} \sum_{i=0}^{q-1} i^r + (r+1)\epsilon^r \cdot q^{r+1} + \sum_{p=1}^{r-2} \frac{r^p \cdot 2^{r-p} \epsilon^p q^p}{p \cdot 2^{n(r-p)}} \sum_{i=2}^{q} i^{r-p} \right]. \quad (21)$$

When $r = 2$, we use

$$\sum_{i=0}^{q-1} i^2 = \frac{q(q-1)(2q-1)}{6} \leq \frac{q^3}{3} \quad (22)$$

to (21) to have $\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{F2}}(\mathcal{A}) \leq 2.7q^3/2^{2n} + 6\epsilon^2 q^3$ as the PRF bound of F2, and additionally let $\epsilon = \ell/2^n$ to derive the bound in case of using GHASH. This proves Lemma 2.

Similarly, for $r = 3$ and $r = 4$, we can apply

$$\sum_{i=0}^{q-1} i^3 = \frac{1}{4}(q^2 \cdot (q-1)^2) < \frac{q^4}{4},$$

$$\sum_{i=0}^{q-1} i^4 = \frac{1}{30}(q(q-1)(2q-1)(3(q-1)^2 + 3(q-1) - 1)) < \frac{2q^5}{5}$$

to (21) to obtain a slightly improved bound than (20).

9.2 Proving the ivE Security Bound of Er

The strategy is basically the same as in Sect. 9.1. We define the ivE scheme Er used in GCM-SIVr following the definition of E2 of GCM-SIV2. Er generates an rn-bit random IV T and encrypts a plaintext using T. Our analysis focuses on the internal key-stream generator $\mathcal{KS}^{(r)}$ of Er, which is a procedure that takes an integer m as input and outputs uniformly random T and m-bit key-stream $\mathbf{S}$. Then we have

$$\mathbf{Adv}^{\mathrm{priv\$}}_{\mathsf{E}r}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{KS}^{(r)}}(\mathcal{A}),$$

where $\mathbf{Adv}^{\mathrm{prg}}_{\mathcal{KS}^{(r)}}(\mathcal{A})$ is defined as $\Pr[\mathcal{A}^{\mathcal{KS}^{(r)}} = 1] - \Pr[\mathcal{A}^{\$} = 1]$, and the oracle \$ takes m as input and outputs a uniform random string of $rn + m$ bits.

Let $X = (X[1], \ldots, X[r]) \in (\{0,1\}^n)^r$ and $\mathbf{P}^{(r)}(X) = \bigoplus_{i=1}^{r} P_i(X[i])$. For integers i and j, let $\mathbf{R}^{(r)}(i,j) = (\mathsf{inc}^j(R_1(i)), \ldots, \mathsf{inc}^j(R_r(i))$, where for $i = 1, \ldots, r$, $R_i : \mathbb{Z} \to \{0,1\}^n$ denotes an independent random function. Let $G^{(r)} = \mathbf{P}^{(r)} \circ \mathbf{R}^{(r)}$. Then the sequence

$$G^{(r)}(i,0) \,\|\, G^{(r)}(i,1) \,\|\, \cdots \,\|\, G^{(r)}(i, m_i - 1)$$

perfectly simulates the key-stream of $\mathcal{KS}^{(r)}$ taking m_i as input.

We note that the PRG-advantage of $\mathcal{KS}^{(r)}$ can be bounded mostly in the same way as Fr with the output chopped to the first n bits, by noting that $\mathbf{R}^{(r)}$ is used instead of $\mathbf{H}^{(r)}$. We also note that $\Pr[\mathsf{inc}^j(R_h(i)) = \mathsf{inc}^{j'}(R_h(i'))] \leq 1/2^n$ for any $(i,j) \neq (i',j')$, where $h \in \{1, \ldots, r\}$. Hence each component function of $\mathbf{R}^{(r)}$ is independent and $1/2^n$-AU. Thus we can define almost the same games and bad flags as in Sect. 9.1.

These observations imply that

$$\mathbf{Adv}^{\mathrm{prg}}_{\mathcal{KS}^{(r)}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{prf}'}_{G^{(r)}}(\mathcal{A}') = \mathbf{Adv}^{\mathrm{prf}}_{G^{(r)}}(\mathcal{A}'), \quad (23)$$

where $\mathbf{Adv}^{\mathrm{prf}'}_{G^{(r)}}$ denotes the advantage of distinguishing between $\mathbf{R}^{(r)}(i,j) \,\|\, G^{(r)}(i,j)$ and $\mathbf{R}^{(r)}(i,j) \,\|\, R'(i,j)$ for an independent random function $R' : \mathbb{Z}^2 \to \{0,1\}^n$, and $\mathcal{A}'$ makes the

first m_1 queries as $(1,0),(1,1),\ldots,(1,m_1-1)$, the second m_2 queries as $(2,0),\ldots,(2,m_2-1)$ and so on, with the restriction that $\sum_{i=1}^{q} m_i \le \sigma$. To see the equality of (23), we remark that $\mathcal{A}'$ can choose m_i possibly adaptively, however, the input distribution of $\mathbf{P}^{(r)}$ does not change whether m_i is chosen adaptively or not. We can therefore focus on non-adaptive $\mathcal{A}'$, and hence ignore the existence of $\mathbf{R}^{(r)}(i,j)$ in the output.

The last term of (23) is bound by (20) or (21) without the preceding multiplication by r, and replacing q with σ and ϵ with $1/2^n$. As a result, $\mathbf{Adv}^{\mathrm{prf}}_{G^{(r)}}(\mathcal{A}')$ is bounded by

$$\frac{4^r \cdot \sigma^{r+1}}{2^{rn}}, \tag{24}$$

or more precisely, by

$$\frac{2^r}{2^{nr}} \sum_{i=0}^{\sigma-1} i^r + \frac{(r+1)\cdot\sigma^{r+1}}{2^{nr}} + \sum_{p=1}^{r-2} \frac{r^p \cdot 2^{r-p} \cdot \sigma^p}{p \cdot 2^{nr}} \sum_{i=2}^{\sigma} i^{r-p}. \tag{25}$$

For the case $r = 2$, we use (22) to (25), and this proves Lemma 3. Theorem 2 is obtained by combining (20), (24), and Lemma 1.

10 Reducing the Number of Keys

GCM-SIVr needs $r^2 + r$ blockcipher keys and r hash function keys, which can be an issue when r gets large. We present a simple solution to reduce the number of blockcipher keys. With this solution, it needs $2r$ blockcipher keys and r hash function keys, if the underlying H_{L_i} allows incremental update. The idea is also implicitly used in Sect. 9.2. For example, consider the case $H_{L_i}(N,A,M) = \mathsf{GHASH}_{L_i}(A,M) \oplus N$ and assume that the last c bits of N is always zero, where $r \le 2^c$. We define

$$\mathbf{P'}^{(r)}(V[1],\ldots,V[r]) = \bigoplus_{i=1}^{r} P'_i(V[i]),$$

and $\widetilde{\mathsf{F}r} = \mathbf{P'}^{(r)} \circ \mathbf{H}^{(r)}$. Since $\widetilde{\mathsf{F}r}$ is the first component function of $\mathsf{F}r$, it is a PRF for input (N,A,M). The overall PRF takes (N,A,M) as input and outputs $(T[1],\ldots,T[r])$, where $T[j] = \widetilde{\mathsf{F}r}(N \oplus \langle j\rangle, A, M)$, and we define $\langle j\rangle = (0^{n-c} \,\|\, \mathsf{str}_c(j-1))$, where $\mathsf{str}_c(j-1)$ denotes the c-bit binary representation of $0 \le j-1 \le r-1$. This construction is secure since each $T[j]$ is generated from $\widetilde{\mathsf{F}r}$ taking distinct inputs. Furthermore, computing $\mathbf{H}^{(r)}(N \oplus \langle j\rangle, A, M)$ is just xoring $\langle j\rangle$ to all the components of $\mathbf{H}^{(r)}(N,A,M)$ (recall that the last c bits of N are assumed to be zero), and hence is quite simple.

11 Discussions and Conclusions

Advantages. GCM-SIV1 offers a security trade-off to GCM-SIV, and the security bound is better if the maximum input length is at most about 2^{16} blocks. The implementation of GCM-SIV1 requires handling the carry in inc function, but we expect that the efficiency impact is not significant. We emphasize the design simplicity of GCM-SIV2 and GCM-SIVr. They are essentially obtained by multiple instances of GCM-SIV1, which allows us to reuse a part of existing software libraries (e.g. OpenSSL) or hardware of GCM. In addition they are parallelizable at the level of high-level components, i.e., GHASH and CTR. For GCM-SIV2 with $n = 128$, we have about $2n/3 = 85.3$-bit security which is practically much stronger than GCM-SIV1, and for GCM-SIVr for $r > 2$, its security is even stronger.

We also remark that the security proof of $\mathsf{F}r$, the tag generation function of GCM-SIVr, is a non-trivial extension of previous results on BBB secure MAC and PRF [Yas10, Osa12,

ZWSW12] using an n-bit blockcipher. In fact, Fr is a variable-input-length blockcipher-based MAC/PRF of $rn/(r+1)$-bit security for any $r \geq 2$, while [Yas10, Osa12] only considered the case $r = 2$. Maurer's PRF [Mau02] also has the same security and works for any r, though it is based on an n-bit PRF rather than an n-bit PRP (i.e. a blockcipher).

Disadvantages. Although GCM-SIV2 and GCM-SIVr give a solution to a theoretical question of designing a simple BBB secure MRAE scheme from a blockcipher, they incurs significant loss in efficiency compared to GCM-SIV and GCM. The computation cost and tag length are precisely increased by a factor of r, which would prohibit practical use as r increases. In addition, they need many keys for the blockcipher and universal hash function, though there is a solution to mitigate this as described in Sect. 10.

Possible Directions. A natural future direction is to consider MRAE schemes having better efficiency while keeping BBB security. In particular, it would be interesting to consider if it is possible to build a scheme as secure as GCM-SIVr while using a shorter tag. In theory we could obtain full n-bit security using $2n$-bit tag using (say) a $2n$-bit blockcipher and $2n$-bit universal hash functions, however, building simple and efficient one that reuses the component of GCM seems a challenging task. Another future direction is reducing the number of keys, preferably to $O(1)$, as Datta et al. [DDN+15] studied for the case $r = 2$.

Acknowledgments

The authors would like to thank participants of Dagstuhl Seminar 16021 (Symmetric Cryptography) and the anonymous reviewers of FSE 2017 for insightful comments. The work by Tetsu Iwata was supported in part by JSPS KAKENHI, Grant-in-Aid for Scientific Research (B), Grant Number 26280045.

References

[AFL+16] Farzaneh Abed, Christian Forler, Eik List, Stefan Lucks, and Jakob Wenzel. RIV for Robust Authenticated Encryption. In Peyrin [Pey16], pages 23–42.

[BDJR97] Mihir Bellare, Anand Desai, E. Jokipii, and Phillip Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *FOCS '97*, pages 394–403. IEEE Computer Society, 1997.

[BPR05] Mihir Bellare, Krzysztof Pietrzak, and Phillip Rogaway. Improved Security Analyses for CBC MACs. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 527–545. Springer, 2005.

[BR00] John Black and Phillip Rogaway. CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions. In Mihir Bellare, editor, *CRYPTO 2000*, volume 1880 of *LNCS*, pages 197–215. Springer, 2000.

[BR06] Mihir Bellare and Phillip Rogaway. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. In Vaudenay [Vau06], pages 409–426.

[BZD+16] Hanno Böck, Aaron Zauner, Sean Devlin, Juraj Somorovsky, and Philipp Jovanovic. Nonce-Disrespecting Adversaries: Practical Forgery Attacks on GCM in TLS. Cryptology ePrint Archive, Report 2016/475, 2016. http://eprint.iacr.org/.

[CAE] CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. http://competitions.cr.yp.to/caesar.html.

[CS14] Shan Chen and John P. Steinberger. Tight Security Bounds for Key-Alternating Ciphers. In Nguyen and Oswald [NO14], pages 327–350.

[CS16] Debrup Chakraborty and Palash Sarkar. On modes of operations of a block cipher for authentication and authenticated encryption. *Cryptography and Communications*, 8(4):455–511, 2016.

[DDN+15] Nilanjan Datta, Avijit Dutta, Mridul Nandi, Goutam Paul, and Liting Zhang. Building Single-Key Beyond Birthday Bound Message Authentication Code. Cryptology ePrint Archive, Report 2015/958, 2015. http://eprint.iacr.org/2015/958.

[Dwo07] Morris Dworkin. Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. NIST Special Publication 800-38D, 2007.

[FFL12] Ewan Fleischmann, Christian Forler, and Stefan Lucks. McOE: A Family of Almost Foolproof On-Line Authenticated Encryption Schemes. In Anne Canteaut, editor, *FSE 2012*, volume 7549 of *LNCS*, pages 196–215. Springer, 2012.

[GJMN16] Robert Granger, Philipp Jovanovic, Bart Mennink, and Samuel Neves. Improved Masking for Tweakable Blockciphers with Applications to Authenticated Encryption. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 263–293. Springer, 2016.

[GL15a] Shay Gueron and Yehuda Lindell. GCM-SIV: Full Nonce Misuse-Resistant Authenticated Encryption at Under One Cycle per Byte. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 109–119. ACM, 2015.

[GL15b] Shay Gueron and Yehuda Lindell. GCM-SIV: Full Nonce Misuse-Resistant Authenticated Encryption at Under One Cycle per Byte. Cryptology ePrint Archive, Report 2015/102, 2015. http://eprint.iacr.org/2015/102.

[GLL16] Shay Gueron, Adam Langley, and Yehuda Lindell. AES-GCM-SIV: Nonce Misuse-Resistant Authenticated Encryption. CFRG Internet-Draft, draft-irtf-cfrg-gcmsiv-01, May 9, 2016.

[HKR15] 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, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 15–44. Springer, 2015.

[IEE06] IEEE Standard for Local and Metropolitan Area Networks Media Access Control (MAC) Security. IEEE Std 802.1AE-2006, 2006.

[IOM12] Tetsu Iwata, Keisuke Ohashi, and Kazuhiko Minematsu. Breaking and Repairing GCM Security Proofs. In Safavi-Naini and Canetti [SC12], pages 31–49.

[IY09a] Tetsu Iwata and Kan Yasuda. BTM: A Single-Key, Inverse-Cipher-Free Mode for Deterministic Authenticated Encryption. In Michael J. Jacobson Jr., Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *SAC 2009*, volume 5867 of *LNCS*, pages 313–330. Springer, 2009.

[IY09b] Tetsu Iwata and Kan Yasuda. HBS: A Single-Key Mode of Operation for Deterministic Authenticated Encryption. In Orr Dunkelman, editor, *FSE 2009*, volume 5665 of *LNCS*, pages 394–415. Springer, 2009.

[JNP15a] Jérémy Jean, Ivica Nikolić, and Thomas Peyrin. Deoxys v1.3. Submission to CAESAR, 2015. http://competitions.cr.yp.to/caesar.html.

[JNP15b] Jérémy Jean, Ivica Nikolić, and Thomas Peyrin. Joltik v1.3. Submission to CAESAR, 2015. http://competitions.cr.yp.to/caesar.html.

[Jou06] Antoine Joux. Authentication Failures in NIST Version of GCM. Public comments to NIST, 2006. http://csrc.nist.gov/groups/ST/toolkit/BCM/comments.html.

[Kro15] Ted Krovetz. HS1-SIV (v2). Submission to CAESAR, 2015. http://competitions.cr.yp.to/caesar.html.

[LPTY16] Atul Luykx, Bart Preneel, Elmar Tischhauser, and Kan Yasuda. A MAC Mode for Lightweight Block Ciphers. In Peyrin [Pey16], pages 43–59.

[LRW11] Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable block ciphers. *J. Cryptology*, 24(3):588–613, 2011.

[LST12] Will Landecker, Thomas Shrimpton, and R. Seth Terashima. Tweakable Blockciphers with Beyond Birthday-Bound Security. In Safavi-Naini and Canetti [SC12], pages 14–30.

[Luc00] Stefan Lucks. The Sum of PRPs Is a Secure PRF. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 470–484. Springer, 2000.

[Mau02] Ueli M. Maurer. Indistinguishability of Random Systems. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 110–132. Springer, 2002.

[MP03] Ueli M. Maurer and Krzysztof Pietrzak. The Security of Many-Round Luby-Rackoff Pseudo-Random Permutations. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 544–561. Springer, 2003.

[MV04a] 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, *INDOCRYPT 2004*, volume 3348 of *LNCS*, pages 343–355. Springer, 2004.

[MV04b] David A. McGrew and John Viega. The Security and Performance of the Galois/Counter Mode of Operation (Full Version). Cryptology ePrint Archive, Report 2004/193, 2004. http://eprint.iacr.org/.

[NO14] Phong Q. Nguyen and Elisabeth Oswald, editors. *EUROCRYPT 2014*, volume 8441 of *LNCS*. Springer, 2014.

[NOMI15] Yuichi Niwa, Keisuke Ohashi, Kazuhiko Minematsu, and Tetsu Iwata. GCM Security Bounds Reconsidered. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 385–407. Springer, 2015.

[NRS14] Chanathip Namprempre, Phillip Rogaway, and Thomas Shrimpton. Reconsidering Generic Composition. In Nguyen and Oswald [NO14], pages 257–274.

[Osa12] Yasushi Osaki. A Study on Deterministic Symmetric Key Encryption and Authentication. Master's thesis, Nagoya University, 2012.

[Pey16] Thomas Peyrin, editor. *FSE 2016*, volume 9783 of *LNCS*. Springer, 2016.

[PS16] Thomas Peyrin and Yannick Seurin. Counter-in-Tweak: Authenticated Encryption Modes for Tweakable Block Ciphers. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 33–63. Springer, 2016.

[RS06] Phillip Rogaway and Thomas Shrimpton. A Provable-Security Treatment of the Key-Wrap Problem. In Vaudenay [Vau06], pages 373–390.

[RVV14] Reza Reyhanitabar, Serge Vaudenay, and Damian Vizár. Misuse-Resistant Variants of the OMD Authenticated Encryption Mode. In Sherman S. M. Chow, Joseph K. Liu, Lucas Chi Kwong Hui, and Siu-Ming Yiu, editors, *ProvSec 2014*, volume 8782 of *LNCS*, pages 55–70. Springer, 2014.

[Sar14] Palash Sarkar. Modes of operations for encryption and authentication using stream ciphers supporting an initialisation vector. *Cryptography and Communications*, 6(3):189–231, 2014.

[SC12] Reihaneh Safavi-Naini and Ran Canetti, editors. *CRYPTO 2012*, volume 7417 of *LNCS*. Springer, 2012.

[SCM08] Joseph Salowey, Abhijit Choudhury, and David A. McGrew. AES Galois Counter Mode (GCM) Cipher Suites for TLS. IETF RFC 5288, 2008.

[ST13] Thomas Shrimpton and R. Seth Terashima. A Modular Framework for Building Variable-Input-Length Tweakable Ciphers. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 405–423. Springer, 2013.

[Vau06] Serge Vaudenay, editor. *EUROCRYPT 2006*, volume 4004 of *LNCS*. Springer, 2006.

[Yas10] Kan Yasuda. The Sum of CBC MACs Is a Secure PRF. In Josef Pieprzyk, editor, *CT-RSA 2010*, volume 5985 of *LNCS*, pages 366–381. Springer, 2010.

[Yas11] Kan Yasuda. A New Variant of PMAC: Beyond the Birthday Bound. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 596–609. Springer, 2011.

[ZWSW12] Liting Zhang, Wenling Wu, Han Sui, and Peng Wang. 3kf9: Enhancing 3GPP-MAC beyond the Birthday Bound. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 296–312. Springer, 2012.