WEM: A New Family of White-box Block Ciphers Based on the Even-Mansour Construction

Jihoon Cho[1], Kyu Young Choi[1], Itai Dinur[2], Orr Dunkelman[3,⋆], Nathan Keller[4,⋆⋆], Dukjae Moon[1], and Aviya Veidberg[4]

[1] Security Research Group, Samsung SDS, Inc.
[2] Computer Science Department, Ben-Gurion University, Israel
[3] Computer Science Department, University of Haifa, Israel
[4] Department of Mathematics, Bar-Ilan University, Israel

Abstract. White-box cryptosystems aim at providing security against an adversary that has access to the encryption process. As a countermeasure against code lifting (in which the adversary simply distributes the code of the cipher), recent white-box schemes aim for 'incompressibility', meaning that any useful representation of the secret key material is memory-consuming.

In this paper we introduce a new family of white-box block ciphers relying on incompressible permutations and the classical Even-Mansour construction. Our ciphers allow achieving tradeoffs between encryption speed and white-box security that were not obtained by previous designs. In particular, we present a cipher with reasonably strong space hardness of 2^{15} bytes, that runs at less than 100 cycles per byte.

1 Introduction

The white-box threat model in cryptography, introduced by Chow et al. [7] in 2002, assumes that the adversary is accessible to the entire information on the encryption process, and can even change parts of it at will. The initial scenario-in-mind behind the model was the Digital Rights Management (DRM) realm, where an authorized user, who of course has full access to the encryption process, may be adversarial. The model has gained more relevance in recent years due to additional applications, such as mitigation of mass surveillance.

Numerous primitives claiming for security at the white-box model (in short: white-box primitives) were proposed in the last few years. These primitives can be roughly divided into two classes.

The first class includes algorithms which take an existing block cipher (usually AES or DES), and use various methods to 'obfuscate' the encryption process, so that a white-box adversary will not be able to extract the secret key.

[⋆] The fourth author was supported in part by the Israeli Science Foundation through grant No. 827/12 and by the Commission of the European Communities through the Horizon 2020 program under project number 645622 PQCRYPTO.

[⋆⋆] The fifth author was supported by the Alon Fellowship.

Pioneered by Chow et al. [7], this approach was followed by quite a few designers. The more common way to fortify the encryption process is using large tables and *random encodings*, as proposed in [7]. Unfortunately, most of these designs were broken by practical attacks a short time after their presentation (see [2, 17, 20]), and the remaining ones are very recent and have not been subjected to extensive cryptanalytic efforts yet. Another disadvantage of the designs in this class is their performance – all of them are orders of magnitude slower than the 'black-box' primitives they are based upon.

The second class includes new block ciphers designed with white-box protection in mind. Usually such designs are based on *key-dependent components* (e.g., S-boxes), designed in such a way that even if a white-box adversary can recover the full dictionary of such a component, he still cannot use this knowledge to recover the secret key. Recent designs of this class include the ASASA family [4], the SPACE family [6], and the WhiteKey and WhiteBlock ciphers [14]. An important advantage of these designs is their better performance and higher security (though, some instantiations of ASASA were broken, see [15, 18]).

A common property of the new white-box designs is *incompressibility* [9] (also called *weak white-box security* [4] and *space hardness* [6]), meaning that an adversary with access to the white-box implementation cannot produce a functionally equivalent program of significantly smaller size. This property is needed, as a white-box adversary can perform *code lifting*, i.e., extract the entire code and use it as an equivalent secret key. While incompressibility does not make code lifting impossible, it does make it harder to implement in practice, especially when the adversary wants to attack multiple targets, e.g., for mass surveillance purposes. The previous designs SPACE and WhiteBlock achieved incompressibility by using *key-dependent pseudo-random functions*.

In this paper we propose a new family of white-box block ciphers in which the basic S-box component is a *pseudo-random permutation*, rather than a pseudo-random function. The new ciphers are based on iterates of the classical Even-Mansour construction [12], in which instead of each key addition one applies an S-box layer, where the S-boxes are key-dependent incompressible permutations. The size of the incompressible S-box is flexible, and can be adjusted to the desired level of incompressibility, without slowing up the encryption process significantly. While the new family proposes similar security level as the SPACE and WhiteBlock ciphers, we show that it allows for additional tradeoffs between performance and white-box security level that were not achievable in previous designs. In particular, we achieve encryption speed of less than 100 cycles per byte with a reasonably strong space hardness of 2^{15} bytes.

This paper is organized as follows. In Section 2 we present the WEM family of white-box block ciphers and explain the rationale behind its design. In Section 3 we analyze the security of the new ciphers in the black-box model. In Section 4 we analyze the security of the new ciphers in the white-box model and compare them with the SPACE and WhiteBlock ciphers. We conclude the paper in Section 5.

2 A New Family of White-Box Block Ciphers Based on Incompressible Permutations

In this section we present WEM – a new family of white-box block ciphers based on iterates of the classical Even-Mansour construction [12] and on a key-less variant of a given block cipher. In order to be specific, we present the scheme with AES as the basic block cipher, but any other iterated block cipher can be used instead.

We begin this section with a brief recap of the Even-Mansour construction. Then we present the new family of block ciphers, and finally we explain the rationale behind its design.

2.1 The Even-Mansour Construction

The Even-Mansour (EM) construction was designed by Even and Mansour [12] in 1991, as an attempt to design the 'simplest possible' block cipher based upon a single public permutation. It uses a publicly-known permutation $P : \{0,1\}^n \rightarrow \{0,1\}^n$, and two independent n-bit keys K_0, K_1. The encryption function is defined simply as $EM_{K_0,K_1}(X) = K_1 \oplus P(K_0 \oplus X)$, for $X \in \{0,1\}^n$. Even and Mansour [12] showed that any attack on EM that requires D queries to the entire scheme and T queries to the permutation P must satisfy $DT = \Omega(2^n)$. On the other hand, attacks on the scheme were presented by Daemen [8], Biryukov and Wagner [5], and Dunkelman et al. [11] who showed that the lower bound of [12] is tight by devising a known-plaintext attack that requires D queries to EM and T queries to P, for any (D,T) such that $DT = \Omega(2^n)$.

As a security level of $2^{n/2}$ is considered insufficient for an n-bit block cipher, several authors proposed to enhance the security level by considering *iterates* of the EM construction. For $r \geq 1$, the r-round EM scheme is defined as

$$rEM_{K_0,K_1,\ldots,K_r}(X) = K_r \oplus P_r(K_{r-1} \oplus P_{r-1}(\cdots (P_1(K_0 \oplus X)))),$$

where $P_1, P_2, \ldots, P_r : \{0,1\}^n \rightarrow \{0,1\}^n$ are public permutations, and $K_0, K_1, \ldots, K_r$ are independent n-bit keys. The iterated EM scheme was studied in numerous papers, and multiple upper and lower bounds on its security level were obtained (see, e.g., [10]). The analysis conducted so far indicates that even for small values of r, the security level of the scheme is high. In particular, for the single-key variant in which $K_0 = K_1 = \ldots = K_r$, no attack faster than $2^n/n$ is known even for 2EM (i.e., iterated EM with 2 rounds).

2.2 The new family of block ciphers

The new family of block ciphers, WEM (standing for white-box Even-Mansour), is based on an iterated EM construction, in which the key additions are replaced by layers of incompressible key-dependent S-boxes. In order to allow flexibility, the scheme uses several parameters: n denotes the block size of the cipher, m denotes the size of the incompressible S-box, where $m|n$ is required. r denotes the number of rounds in the underlying iterated EM construction, E denotes the 'name' of the underlying block cipher (e.g., AES), and d denotes the number of rounds we take in its key-less version.

The overall structure of the cipher The WEM(n, m, r, E, d) encryption scheme is a modification of the r-round EM scheme, in which:

- A d-round reduced variant of E with the all-zero key is used as the 'public permutation' P. (The same permutation can be used in all rounds of WEM.)
- Each key addition is replaced by an S-box layer, which consists of parallel application of n/m incompressible m-to-m bit S-boxes. For this, we generate $(r+1)n/m$ independent incompressible S-boxes[5] $S_1, S_2, \ldots, S_{(r+1)n/m}$ and use S-boxes $S_{(i-1)n/m+1}, \ldots, S_{in/m}$ in the i'th S-box layer. (The generation of the S-boxes is presented below.)

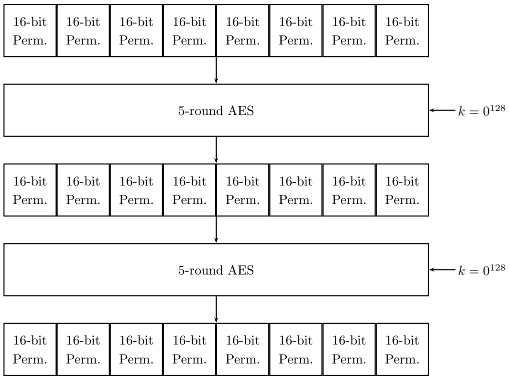

Fig. 1. The cipher WEM(128,16,2,AES-128,5)

A specific instantiation of the scheme, with $n = 128$, $m = 16$, $r = 2$, $E =$AES-128, and $d = 5$, is presented in Figure 1. As can be seen in the figure, the 128-bit plaintext is divided into eight 16-bit values. These values enter the first S-box layer. The outputs of the S-box layer are treated as a 128-bit state, to which 5-round AES with the zero key is applied.[6] Then, the value is split again and enters another S-box layer. The outputs of the S-box layer are again unified and processed with 5-round AES with the zero key, and the resulting values are passed through a final S-box layer.

[5] We may also reuse S-boxes to obtain greater flexibility, as noted below.

[6] We note that instead, a per-domain fixed key can be used, e.g., each country gets its own key, or even a per-user key. However, we assume this key to be publicly known.

Due to the choice of parameters, the cipher presented in the figure has 24 S-boxes and each encryption has time complexity roughly equivalent to a single AES encryption plus 3 sequences of 8 parallel table lookups.

The structure of the S-box Since the S-box can be isolated by a white-box adversary, it must be a stand-alone primitive that ensures n-bit security to the key even against an attacker that has the full S-box code-book in his disposal.

To obtain this goal, we instantiate the S-box using a two-step procedure. First, we use the secret key to generate a long sequence of pseudo-random bits, and then we use the Fisher-Yates shuffle algorithm [13] to instantiate an S-box from m bits to m bits from the pseudo-random sequence. We note that similar methods were used to generate a pseudo-random function in the SPACE family [6] and in WhiteBlock [14].

The Fisher-Yates algorithm gets an array a of 2^m entries of m bits each, and outputs the designed S-box (where the S-box value on input i is simply $a[i]$). It has the following simple structure:

```
for $i = 0 \ldots 2^m - 1$ do
    $a[i] \leftarrow i$
end for
for $i = 2^m - 1 \ldots 0$ do
    $j \leftarrow$ random integer modulo $i$
    exchange $a[j]$ and $a[i]$
end for
```

The Fisher-Yates shuffle was shown in [13] to provide perfect randomness: when instantiated with a truly random sequence of bits, it generates a truly random permutation over the range $0, 1, \ldots, 2^m - 1$, meaning that each permutation $\sigma \in \mathcal{S}_{2^n}$ is obtained with probability $1/(2^n)!$.

The pseudo-random sequence is generated using the block cipher E (keyed with the master key) in counter mode. For example, in the case $E =$AES-128, we set the key of AES-CTR as our 128-bit secret master key and generate pseudo-random numbers by encrypting 128-bit plaintexts $0, 1, \ldots$ (as many numbers as required). Thus, the value of the encrypted plaintext functions as the state of the pseudo-random generator, and is incremented as more pseudo-random numbers are required.

Our construction requires generating several such S-boxes (e.g., $8 \cdot 3 = 24$ S-boxes in the above example), and the only difference between the generation of these S-boxes is in the state of the generator (the value of the plaintext encrypted). This value is initialized to 0 for the first S-box and incremented as long as pseudo-random numbers are required (namely, the state is preserved across the initializations of the different S-boxes).

2.3 Design Rationale

The design aims at achieving good performance, while at the same time providing strong security both in the black-box and white-box models (with an appropriate choice for the number of rounds).

1. Good performance and strong security in the black-box model are achieved by using the iterated EM construction as the basis of the cipher. The numerous works published so far on iterated EM suggest that even with only two rounds, the security level of the scheme (in the black-box model) is close to 2^n, and of course, the scheme becomes even stronger when simple key addition is replaced by a secret S-box layer. Furthermore, by taking a round-reduced variant of E as the public permutation of 2EM we obtain good performance without sacrificing security, as a round-reduced variant of the cipher with sufficiently many rounds already provides sufficient randomness (even if it is not secure as a stand-alone cipher).
2. The white-box security is obtained by making it very hard for an adversary to extract the master key, even if the full code-book of all S-boxes is known. (Note that the user gets the S-boxes in the form of look-up tables, so that even a white-box adversary does not have access to the generation process of the S-box.) Hence, the generation of the S-box must be 'very secure'. On the other hand, as the S-boxes are generated only rarely, we can opt for security in their generation, allowing some performance overhead. Our S-box generation satisfies the required security criterion: while an adversary that knows the full code-book can reverse the Fisher-Yates process and find the pseudo-random string that was used in the S-box generation, this only gives him knowledge of a few plaintext/ciphertext pairs of AES-CTR. Those cannot be used to recover the secret key, unless AES-CTR is insecure.
3. A main idea behind the design is to base it upon thoroughly-analyzed components, in order to gain confidence in its security. This can be seen in the previous two points, where the security of WEM is 'reduced' to the security of iterated EM (though, not in a provable manner) and AES-CTR.
4. The S-box generation process also ensures incompressibility. Indeed, recall that the Fisher-Yates shuffle provably generates a random permutation if the initial sequence is random. Hence, if an S-box (given in a form of a lookup table) has a compressed representation, this representation can be used to distinguish the pseudo-random initial sequence from a truly random sequence, or in other words, to provide a distinguisher for AES-CTR.
5. Given a desired level of incompressibility, one can choose the parameter m appropriately to obtain it. More flexibility can be obtained by allowing re-use of S-boxes. For example, one may use a single S-box for the full scheme, but then the public permutations must be made slightly different (e.g., by using another fixed key instead of the zero key) in order to avoid slide attacks.
6. As noted above, one of the main differences between WEM and the SPACE [6] and WhiteBlock [14] families is that we use secret permutation S-boxes (rather than secret pseudo-random function S-boxes) in our iterated Even-Mansour scheme.

2.4 Performance

We implemented the cipher WEM(128,16,12,AES-128,5) (which is our main instance for white-box security) using AES rounds which are based on tables (i.e., without using the AES-NI instruction set), thus offering a relatively portable code which offers decent performance figures on 32-bit platforms. The code was compiled under g++ 4.8.4 and was run on an Intel(R) Core(TM) i7-5500U CPU @ 2.40 GHz (after being compiled using the -O2 flag). The running speed we obtained for this basic code was 96.8 cycles per byte.

Compared to WEM(128,16,12,AES-128,5), the related WhiteBlock instance HOUND with 16-bit S-boxes requires a bit more than 140 cycles per byte. However, we note that the comparison is not completely fair. First, the authors of WhiteBlock use a different platform, and in particular, they use AES-NI. On the other hand, the memory consumption of HOUND with 16-bit S-boxes is about 4 times larger than WEM(128,16,12,AES-128,5), as it uses an expanding S-box.

3 Security in the Black-box Model

In this section we present security analysis of the WEM family in the black-box model. Our conclusion is that *two* rounds of the scheme are sufficient for providing strong security, and in particular, WEM(128,16,2,AES-128,5) is expected to provide 128-bit security, basing on previous extensive analysis of its components. On the other hand, we show that *one* round of the scheme is not sufficient, by devising an attack with complexity slightly higher than $2^{n/2}$ for an n-bit block size. Due to space constraints, the more technical parts of the analysis are presented in the full version of this paper.

In order to be specific, we assume throughout the analysis that the underlying block cipher is AES-128, and focus on the variants WEM(128,8,2,AES-128,5) and WEM(128,16,2,AES-128,5) described above. When WEM is used with another underlying block cipher instead of AES-128, a separate security analysis should be conducted. For brevity, we abbreviate WEM(128,8,2,AES-128,5) and WEM(128,16,2,AES-128,5) to WEM-8 and WEM-16, respectively.

As justified in Section 2.2, for the sake of black-box analysis we may view the secret S-boxes of our construction as random permutations. The security of WEM-8 and WEM-16 is related to the security of several previously studied constructions:

1. 2-round Iterated Even-Mansour construction [10],
2. Standard AES with 128-bit key,
3. AES with secret S-boxes [19],
4. 10-round AES with random S-boxes,
5. Known-key round-reduced AES,

and results on these five constructions can be used to obtain evaluation of the security of WEM-8 and WEM-16, as described below.

When considering round-reduced variants of WEM-8 and WEM-16, we note that any such round-reduced variant employs a final secret S-box layer. For most of the attacks described below, we claim that our round-reduced construction is at least as strong as a previously studied round-reduced construction, thus establishing confidence in the security of our design. For sake of convenience, in this section we count the rounds in units of AES rounds, so WEM-8 and WEM-16 have 10 rounds each. The following is a brief assessment of the security with respect to various attack techniques.

Key recovery attacks in general. The secret S-box layers at the beginning and the end of the encryption make round-reduced variants of WEM-8 and WEM-16 significantly stronger than corresponding variants of AES with respect to key-recovery attacks. This is due to the fact that an adversary cannot 'peel off' the first/last rounds without guessing a very significant amount of key material. In this respect, the security of WEM-8 and WEM-16 can be derived from the security of AES with secret S-boxes, studied in [19]. The best currently known attack on this version of AES is on 6 rounds [19], with time complexity of 2^{96}, and it translates to an attack on 5-round WEM-8/WEM-16 with the same complexity (due to the additional MixColumns and secret Sbox layers at the end of the cipher). This is clearly far from endangering our 10-round construction. (It should be noted however that there is no direct reduction from WEM-8 or WEM-16 to AES with secret S-boxes, since in WEM, only three layers of S-boxes are secret and not all of them).

Differential and linear characteristics. The analysis here is somewhat technical, and thus, is presented in the full version of this paper. The conclusion is that for WEM-8, it is expected that any 4-round differential has probability of less than 2^{-90}, and any 4-round linear hull has a bias of less than 2^{-45}. For WEM-16, we prove that the number of active S-boxes in any 4-round characteristic is at least 15 (and this is tight), and expect that any 4-round differential has probability of less than 2^{-75} and any 4-round linear hull has a bias of less than $2^{-37.5}$. As in addition, for 4 rounds of our construction that contain a secret S-box layer, the actual best differential characteristics and linear approximations are unknown to the adversary, we conclude that the full 10-round WEM-8 and WEM-16 are expected to be immune to both differential and linear cryptanalysis.

Boomerang attacks. The boomerang attack of Biryukov [3] on round-reduced AES (with at most 6 rounds) can be adapted to WEM-8 and WEM-16, with the same probability as in AES. However, the key recovery part of the attack becomes significantly more expensive, and thus, even on 6-round WEM-8/WEM-16 its complexity is expected to be extremely high.

Square attacks. The classical Square attacks on round-reduced AES are applicable to WEM-8/WEM-16 with at most 5 rounds, but their complexity becomes much higher. Actually, this is the class of attacks considered in [19] (on AES with secret S-boxes), and the best current attack of this class requires 2^{96} time for 5-round WEM-8, and a similar amount for 5-round WEM-16.

Impossible differentials. Similarly to the previous cases, the classical impossible differentials apply to our construction but with a significantly more expensive key recovery phase for the full attacks. Therefore, we do not expect these attacks to break more rounds of our construction compared to AES (where the best attack requires more than 2^{100} data and time for 7 rounds).

Collision attacks (Demirci-Selçuk attacks). It is expected that reduced WEM-8 and WEM-16 are much stronger than reduced AES with respect to these attacks, due to the secret S-box layer in the middle (which increases significantly the number of possible multisets) and the outer secret S-box layers that make key recovery more expensive. As the best known collision attack on round-reduced AES requires 2^{98} data and time for 7 rounds, we expect that WEM-8 and WEM-16 with at least 7 rounds are secure with respect to collision attacks.

Attacks on the EM construction. The added S-box layer in the middle makes the cipher a 2-round EM construction (rather than the relatively weak 1-round EM construction). The best currently known attacks on 2-round EM [10] are only slightly faster than exhaustive key search. Furthermore, WEM-8 and WEM-16 are stronger than 2-round EM, since the key-additions of EM are replaced in WEM-8/WEM-16 with secret S-box layers, which make all current attacks on 2-round EM inapplicable to WEM-8/WEM-16.

Related-key attacks. WEM-8 and WEM-16 are expected to be immune to related-key attacks, due to the key generation procedure. Indeed, as no related-key properties are known for *full AES-128*, it is expected that two related keys (even with relation chosen by the adversary) lead to two unrelated output streams of AES-CTR, and thus, the sets of secret S-boxes generated for the two keys do not have any easy-to-exploit relation. Therefore, it is expected that no related-key attacks on WEM-8/WEM-16 can target more rounds that the single-key attacks (i.e., not more than 7 of the 10 rounds).

WEM(128,8,1,AES-128,10) does not supply 128-bit security

The schemes WEM-8 and WEM-16 considered above are 'minimal', in the sense that if the underlying iterated EM construction of WEM has only one round, then the security level of the scheme is much weaker than 2^{128}. This is similar to the situation with iterated EM schemes, where the security level of 1-round EM is only $2^{n/2}$ while with $r \geq 2$ rounds the security increases significantly (so that no attack faster than $2^{128}/128$ is known).

To show this, we present a structural attack on 1-round WEM, which is a variant of the chosen plaintext attack on the Even-Mansour scheme by Daemen [8]. In the attack, we consider pools of 2^m chosen plaintexts that assume all possible values in the input of one S-box, and the same value in the input to all other S-boxes. This property is, of course, preserved by the first secret S-box layer. Then, we look at the corresponding ciphertexts, and in each S-box output of the final S-box layer, we count the number of values that occur 0 times, the number of values that occur 1 time, etc. As this property is also preserved by the S-box layer, it allows us to use comparison between 1-round WEM and key-less AES to recover the secret S-boxes. The full attack is presented in the full version of this paper. The complexity of the attack is only slightly higher than $2^{n/2}$, thus showing that 1-round WEM is not secure and should not be used.

4 Space-Hardness of the WEM Ciphers

The notion of *space-hardness* was introduced in [6] as a generalization of the notion of *weak white-box security* introduced in [4].

Definition 1. *A cipher is said to be* (M, Z)*-space hard if it infeasible for an adversary to encrypt (decrypt) a randomly chosen plaintext with probability more that* 2^{-Z} *given code (table) size less than* M*.*

There are several motivations behind this definition. One of them is that a space-hard cipher makes it more difficult for a DRM attacker in the white-box setting to distribute meaningful attack code (whose size is large). Additionally, a space-hard cipher may make it more challenging for malware (limited by communication) to leak meaningful secrets from an infected network.

In this section, we compute the number of rounds required for the WEM ciphers to achieve space-hardness, and compare the space-hardness security and performance of WEM to that of the schemes proposed in [6, 14]. For sake of simplicity and for comparison with previous work [6, 14], we only consider in this section instantiations of our schemes with a single secret S-box. We note that a more formal treatment of space-hardness was published in [14] by Fouque et al., using the notion of *weak incompressibility* (formulated as a cryptographic game with the aim of obtaining provable security). Since our motivation is more practical, we will refer to the less formal space-hardness definition of [6]. Hence, our security analysis will be cryptanalytic in nature (focusing on or the best algorithm for breaking our scheme rather than on provable security).[7]

Nevertheless, we point out two issues mentioned in [14] that are relevant for our space-hardness security analysis. First, for an n-bit block cipher, given T words of memory, one cannot hope for space-hardness security better than $Z = n - \log(T)$. The reason is that the memory can simply be utilized to store plaintext/ciphertext pairs, allowing the adversary to correctly encrypt (or decrypt) a fraction at least $2^{\log(T)-n}$ of the code-book. Even when we restrict the adversary's memory to contain entries of the secret S-box in our scheme, it is still possible to store the particular entries that are accessed in the encryption procedure of about T plaintexts (up to some multiplicative factor which depends on the number of times the S-box is accessed in an encryption). A second issue is that our analysis will indeed assume that the adversary's memory contains only secret S-box entries. While we are not aware of significantly better attacks that store other types of information, these attacks are generally much harder to analyze.[8]

[7] Interestingly, it is shown in [14] that for certain types of schemes, the gap between the number of rounds required to resist the best known attack and the the number of rounds required to obtain provable security is not large.

[8] Resistance to such attacks is addressed by the *strong incompressibility* definition of [14], which also gives a scheme (called WhiteKey) that provably achieves this security notion. However, WhiteKey is a key generator rather than a block cipher, and hence is incomparable to our scheme.

4.1 Previous Space-Hard Block Ciphers

There are two previous space-hard block cipher designs. The first one is SPACE, introduced in [6]. SPACE is a 128-bit generalized Feistel structure with a secret expanding S-box of input size m (where m is a parameter of the block cipher instance) and output size $128 - m$.[9]

The second space-hard block cipher design is WhiteBlock, introduced in [14]. The general structure of WhiteBlock is more similar to our scheme. It is a 128-bit block cipher family, where each round contains a secret S-box layer followed by several rounds of standard AES (an AES layer). There are several differences between our scheme and WhiteBlock. The most relevant one in terms of space-hardness is the structure of the secret S-box layer. In WhiteBlock, the secret S-box layer is a single-round Feistel-like structure. For a secret S-box of input size m bits (where m is a parameter of the block cipher instance) and $k = \lfloor 64/m \rfloor$, the 128-bit state is partitioned into two parts, where the 'right part' contains km bits and the 'left part' contains the remaining $128 - km$ bits. The output size of each S-box is $128 - km$ bits (equal to the size of the left part), hence it is an expanding S-box. The secret S-box layer applies k parallel S-boxes to the km bits of the right part of the state and XORs their outputs (in some arbitrary order) to the right part (hence the km bits of the right part are left unchanged).

WhiteBlock has two variants: PuppyCipher, which was designed with provable security in mind, and Hound which is optimized for performance. The difference between the two schemes is in the AES layer (but not in the secret S-box layer). As our main goal is to resist cryptanalysis while optimizing performance, our scheme is more comparable to Hound. We note that it should be possible to tweak our AES layer and apply similar provable security arguments to our scheme as in PuppyCipher, but this is out of the scope of this paper.

4.2 Space-Hardness of our Scheme

We evaluate the space-hardness of our proposal and show that it can be achieved using less rounds than the previous schemes of [6, 14] (thus, resulting in a faster cipher with the same level of white-box security). We start by analyzing our scheme WEM(128,16,r,AES-128,5) and assume that the adversary obtained a fraction of 1/4 of the S-box entries (the value 1/4 is chosen to be comparable to the analysis of [6, 14]). We then generalize the analysis.

We consider r rounds of our scheme and determine the minimal value of r such that it achieves $(T/4, 112)$-space hardness, where T is the size of the 16-bit S-box in 16-bit words (and we aim for the maximal achievable security of 112 bits for a 128-bit cipher with a 16-bit S-box). Our analysis is related to the one of [14] for WhiteBlock, although less formal. The goal is to show that given an arbitrary set of (only) 1/4 of the S-box entries, the adversary cannot guess the encryption of any plaintext with probability which is significantly higher than 2^{-128}. Note

[9] For the sake of convenience, we rename the block cipher instance parameters for both previous space-hard designs [6, 14].

that the set of S-box entries is chosen by the adversary and is not arbitrary (in particular, it can correlate with the encryption/decryption procedure of several plaintexts/ciphertexts). However, roughly speaking, a set of $2^{16}/4$ of the S-box entries can be chosen to reveal information about the encryption of (no more than) 2^{16} plaintexts, whereas for the rest of the plaintexts the analysis below will apply.[10]

The encryption procedure of WEM(128,16,r,AES-128,5) contains $8r$ S-boxes of 16 bits. Therefore, an adversary can encrypt a random plaintext if he is given the corresponding $8r$ S-box entries, which occurs with probability $2^{-2\cdot 8r}$ (assuming that the known S-box entries are arbitrary). Hence, taking $r = 9$ such that $2^{-2\cdot 8r} < 2^{-128}$ should prevent the adversary from correctly encrypting any of the 2^{128} plaintexts. However, the adversary can still miss the entries of several S-boxes and succeed in encrypting the plaintext with probability better than 2^{-128} simply by guessing the S-box outputs. A guess for an S-box output is correct with probability $1/(2^{16} - 2^{14}) < 2^{-15}$ (the adversary has 2^{14} entries of the permutation). Hence, we require that the adversary misses only 8 S-box entries[11] with very low probability (but we do not mind if the adversary misses 9 S-box entries, as $2^{15\cdot 9} < 2^{-128}$).

Overall, to predict the encryption of a plaintext with probability better than 2^{-128}, the adversary should have $8r - 8$ S-box entries which can occur at $\binom{8r}{8}$ places. Therefore, we require that $2^{-2(8r-8)} \cdot \binom{8r}{8} < 2^{-128}$, which is satisfied for $r \geq 12$.

More generally, for a block cipher with an m-bit S-box and $k = n/m$ S-boxes in a round, we apply a similar line of arguments to analyze the number of rounds required to obtain $(2^{-\alpha} \cdot T, n - \log(T))$-space hardness (where T is the S-box size). If the adversary has a $2^{-\alpha}$ fraction of the 2^m possible S-box entries, then we require $2^{-\alpha\cdot k(r-1)} \cdot \binom{k\cdot r}{k} < 2^{-k\cdot m}$ (namely, the adversary misses only k S-box inputs with very low probability). Since $\binom{k\cdot r}{k} < (k \cdot r)^k$, it is sufficient to require

$$-\alpha \cdot k(r-1) + k\log(k) + k\log(r) < -k \cdot m.$$

Dividing by αk , we get $-r + 1 + \log(r)/\alpha + \log(k)/\alpha < -m/\alpha$ or

$$r - \log(r)/\alpha > m/\alpha + \log(k)/\alpha + 1.$$

In other words, the required number of rounds r is larger than m/α by an additive logarithmic factor.

[10] We note that in terms of provable security, it was shown in [14] for WhiteBlock (and similar arguments can be applied to our scheme) that the analysis for an arbitrary set of S-box entries should give a close estimation to the number of rounds required to achieve the desired security level of 112 bits.

[11] We point out that the adversary can reduce the number of guesses in case of common missed S-boxes entries. We do not expect this to give the adversary a significant advantage, as the adversary can only miss a small number of S-box entries in the encryption which are likely to be distinct. Nevertheless, this is a shortcoming of our analysis (which is also present in the analysis of [14]).

Next, we compare our scheme to the previous proposals of [6, 14]. For the sake of simplicity, we focus on S-box sizes m which divide the block size n.[12]

Comparison to WhiteBlock [14] According to the provable security analysis of [14], for an S-box size of m bits and $\alpha = 2$ (namely, assuming that the adversary has 1/4 of the code) WhiteBlock should have $r = m + 2$ rounds (assuming that m divides the block size n). Therefore, for $\alpha = 2$, our scheme requires half the number of rounds, up to an additive logarithmic factor. However, this comparison is not completely fair since it was obtained using different analysis methods (even though they are related). Hence, we redo our analysis for WhiteBlock, and show that it gives similar results as the related analysis [14].

As in Section 4.1, we denote the number of S-boxes in a round of WhiteBlock by k, giving $km = 64$, namely each S-box maps m bits to 64 bits. Similarly to the previous section, we require that the adversary cannot guess the encryption of any plaintext with probability which is significantly higher than 2^{-128} given an arbitrary set of the S-box entries of size $2^{m-\alpha}$. The encryption procedure of r rounds contains kr S-boxes of m bits, and since the output of each S-box is 64 bits, we require that for all plaintexts, the adversary misses at least one S-box entry in at least 2 different rounds. As $2^{-64 \cdot 2} \leq 2^{-128}$, this should suffice to prevent the adversary from predicting the output of an encryption. Note that we require that the missed entries occur in distinct rounds, since even if the adversary misses several S-box entries in a single round, he can directly guess the 64-bit output of the left part of the secret S-box layer (rather than guessing the output of each S-box separately).

To predict the encryption of a plaintext with probability better than 2^{-128}, the adversary should have all the $k(r-1)$ S-box entries which can occur in $r-1$ rounds, and there are $\binom{r}{r-1} = r$ options to choose this round. Overall, we require that $2^{-\alpha k(r-1)} \cdot r < 2^{-2km}$ or $-\alpha k(r-1) + \log(r) < -2ms$. Rearranging, we obtain

$$r - \log(r)/(\alpha k) > 2m/\alpha + 1.$$

In other words, the required number of rounds r is about $2m/\alpha$ and is twice the number of rounds required by our scheme up to additive logarithmic factors. This may seem obvious since the S-boxes of WhiteBlock encrypt only half of the state in each S-box layer, whereas they cover the full state in our scheme. However, this simplistic argument does not take into account the fact that each S-box of WhiteBlock is expanding and hence in order to predict the encryption of a plaintext with good probability, the adversary is allowed to miss less S-box entries (while guessing their values) in WhiteBlock compared to our scheme. Our analysis shows that the use of half as many expanding S-boxes in WhiteBlock compared to our scheme increases the number of rounds required to achieve the same space-hardness security, and thus, generally leads to slower encryption speed. Nevertheless, if one seeks to minimize the number of secret S-box look-ups

[12] It is also possible to instantiate our scheme for values of m that do not divide n, as briefly discussed in Section 4.4.

in the encryption process, then WhiteBlock is superior to our scheme (which has more table look-ups, but evaluated in parallel).

Comparison to SPACE [6] Unlike the case of WhiteBlock whose structure is similar to WEM (both using interleaved applications of a secret S-box layer and an AES layer), the SPACE family differs from WEM significantly, having a generalized Feistel structure. Of course, we can directly compare performance figures, but SPACE was designed with a large security margin and hence, is expectedly much slower. Thus, comparing design strategies will be more interesting.

If we ignore the fact that there are no AES layers in SPACE and redo the security analysis presented above, we get that the SPACE design strategy requires the smallest number of table lookups to achieve space-hardness, but the largest number of secret S-box layers. This is a direct continuation of the trend we previously observed: as we reduce the number of S-boxes applied in a single round, we can use S-boxes with larger output sizes, and thus we need fewer secret table look-ups in the cipher to achieve space-hardness. On the other hand, we still need more secret S-box layers since the reduction in the number of S-boxes is not sufficient to reduce the number of rounds.

4.3 Space-Hardness using Permutation S-boxes

While previous space-hard designs were built using randomly chosen S-boxes, our scheme was built using permutation S-boxes. This has some impact on the space hardness of our scheme, as a permutation on m-bit words can be represented using less memory compared to a random function mapping m-bit words to m-bit words. However, the difference is only by a small multiplicative factor of about $1 - 1.44/m$, since by Stirling's approximation, $\log((2^m)!) > m \cdot 2^m - \log(e)2^m \approx (1 - 1.44/m)(m \cdot 2^m)$. For example, representing a 16-bit random function requires $16 \cdot 2^{16}$ bits, while a 16-bit random permutation requires about $16 \cdot 2^{16} - (1.44/16)(16 \cdot 2^{16}) = 14.56 \cdot 2^{16}$ bits.[13]

4.4 Concrete Instances

Our main instance uses 16-bit S-boxes. It has 12 rounds and is claimed to have $(1/4 \cdot 14.56 \cdot 2^{16}, 112)$-space hardness or $(2^{14.86}, 112)$-space hardness in bytes.

We note that additional instances can be picked by choosing additional S-boxes sizes (e.g., we can define an instance with a 21-bit S-box, where the S-box layer contains 6 S-boxes and 2 bits are left unchanged), although that requires a slightly more technical security analysis.

[13] This factor is even smaller when considering representation of a fraction of the S-box entries.

5 Conclusions

In this paper we presented a new family of white-box block ciphers, called WEM, which combines the iterated Even-Mansour construction with incompressible S-boxes and a round-reduced key-less variant of a ‘standard’ block cipher (e.g., the AES). The structure of WEM allows obtaining good performance, while basing the security confidence in the black-box model on the extensive analysis of the cipher’s components, and the security in the white-box model on the provable randomness of the Fisher-Yates shuffle algorithm.

Our cipher is an SP network, in which the incompressible S-boxes are random permutations. This is in contrast with the previous SPACE and WhiteBlock designs, in which the secret S-boxes are expanding, and the cipher is either a generalized Feistel construction (SPACE) or interleaving of Feistel layers with SPN layers (WhiteBlock). We showed that using an SP network allows reducing the number of rounds in the scheme (for the same space-hardness level), and thus, making the scheme faster if application of S-boxes in parallel is possible. In particular, we present a specific scheme called WEM(128,16,12,AES-128,5) with space hardness of $(2^{14.86}, 112)$ bytes and encryption speed of less than 100 cycles per byte.

References

1. Adams, C.M., Miri, A., Wiener, M.J. (eds.): Selected Areas in Cryptography, 14th International Workshop, SAC 2007, Ottawa, Canada, August 16-17, 2007, Revised Selected Papers, Lecture Notes in Computer Science, vol. 4876. Springer (2007)
2. Billet, O., Gilbert, H., Ech-Chatbi, C.: Cryptanalysis of a White Box AES Implementation. In: Handschuh, H., Hasan, M.A. (eds.) Selected Areas in Cryptography, 11th International Workshop, SAC 2004, Waterloo, Canada, August 9-10, 2004, Revised Selected Papers. Lecture Notes in Computer Science, vol. 3357, pp. 227–240. Springer (2004)
3. Biryukov, A.: The Boomerang Attack on 5 and 6-Round Reduced AES. In: Dobbertin, H., Rijmen, V., Sowa, A. (eds.) Advanced Encryption Standard - AES, 4th International Conference, AES 2004, Bonn, Germany, May 10-12, 2004, Revised Selected and Invited Papers. Lecture Notes in Computer Science, vol. 3373, pp. 11–15. Springer (2004)
4. Biryukov, A., Bouillaguet, C., Khovratovich, D.: Cryptographic Schemes Based on the ASASA Structure: Black-Box, White-Box, and Public-Key (Extended Abstract). In: Sarkar, P., Iwata, T. (eds.) Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8873, pp. 63–84. Springer (2014)
5. Biryukov, A., Wagner, D.: Advanced Slide Attacks. In: Preneel, B. (ed.) Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding. Lecture Notes in Computer Science, vol. 1807, pp. 589–606. Springer (2000)

6. Bogdanov, A., Isobe, T.: White-Box Cryptography Revisited: Space-Hard Ciphers. In: Ray, I., Li, N., Kruegel, C. (eds.) Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015. pp. 1058–1069. ACM (2015), http://doi.acm.org/10.1145/2810103.2813699
7. Chow, S., Eisen, P.A., Johnson, H., van Oorschot, P.C.: White-Box Cryptography and an AES Implementation. In: Nyberg, K., Heys, H.M. (eds.) Selected Areas in Cryptography, 9th Annual International Workshop, SAC 2002, St. John's, Newfoundland, Canada, August 15-16, 2002. Revised Papers. Lecture Notes in Computer Science, vol. 2595, pp. 250–270. Springer (2002)
8. Daemen, J.: Limitations of the Even-Mansour Construction. In: Imai, H., Rivest, R.L., Matsumoto, T. (eds.) Advances in Cryptology - ASIACRYPT '91, International Conference on the Theory and Applications of Cryptology, Fujiyoshida, Japan, November 11-14, 1991, Proceedings. Lecture Notes in Computer Science, vol. 739, pp. 495–498. Springer (1991)
9. Delerablée, C., Lepoint, T., Paillier, P., Rivain, M.: White-Box Security Notions for Symmetric Encryption Schemes. In: Lange et al. [16], pp. 247–264
10. Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Key Recovery Attacks on Iterated Even-Mansour Encryption Schemes. J. Cryptology 29(4), 697–728 (2016)
11. Dunkelman, O., Keller, N., Shamir, A.: Slidex Attacks on the Even-Mansour Encryption Scheme. J. Cryptology 28(1), 1–28 (2015)
12. Even, S., Mansour, Y.: A Construction of a Cipher from a Single Pseudorandom Permutation. J. Cryptology 10(3), 151–162 (1997)
13. Fisher, R.A., Yates, F.: Statistical tables for biological, agricultural and medical research. Oliver and Boyd (1938)
14. Fouque, P., Karpman, P., Kirchner, P., Minaud, B.: Efficient and Provable White-Box Primitives. IACR Cryptology ePrint Archive 2016, 642 (2016), http://eprint.iacr.org/2016/642
15. Gilbert, H., Plût, J., Treger, J.: Key-Recovery Attack on the ASASA Cryptosystem with Expanding S-Boxes. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 475–490. Springer (2015)
16. Lange, T., Lauter, K.E., Lisonek, P. (eds.): Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers, Lecture Notes in Computer Science, vol. 8282. Springer (2014)
17. Lepoint, T., Rivain, M., Mulder, Y.D., Roelse, P., Preneel, B.: Two Attacks on a White-Box AES Implementation. In: Lange et al. [16], pp. 265–285
18. Minaud, B., Derbez, P., Fouque, P., Karpman, P.: Key-Recovery Attacks on ASASA. In: Iwata, T., Cheon, J.H. (eds.) Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9453, pp. 3–27. Springer (2015)
19. Tiessen, T., Knudsen, L.R., Kölbl, S., Lauridsen, M.M.: Security of the AES with a Secret S-Box. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 175–189. Springer (2015)
20. Wyseur, B., Michiels, W., Gorissen, P., Preneel, B.: Cryptanalysis of White-Box DES Implementations with Arbitrary External Encodings. In: Adams et al. [1], pp. 264–277