Treatment of the Initial Value in Time-Memory-Data Tradeoff Attacks on Stream Ciphers*

Orr Dunkelman[1,**] and Nathan Keller[2,***]

[1] Département d'Informatique
École normale supérieure
45 rue d'Ulm, Paris 75005, France
orr.dunkelman@ens.fr
[2]Einstein Institute of Mathematics, Hebrew University.
Jerusalem 91904, Israel
nkeller@math.huji.ac.il

Abstract. Time-Memory Tradeoff (TMTO) attacks on stream ciphers are a serious security threat and the resistance to this class of attacks is an important criterion in the design of a modern stream cipher. TMTO attacks are especially effective against stream ciphers where a variant of the TMTO attack can make use of multiple data to reduce the off-line and the on-line time complexities of the attack (given a fixed amount of memory).

In this paper we present a new approach to TMTO attacks against stream ciphers using a publicly known initial value (IV): We suggest not to treat the IV as part of the secret key material (as done in current attacks), but rather to choose in advance some IVs and apply a TMTO attack to streams produced using these IVs. We show that while the obtained tradeoff curve is identical to the curve obtained by the current approach, the new technique allows to mount the TMTO attack in a larger variety of settings. For example, if both the secret key and the IV are of length n, it is possible to mount an attack with data, time, and memory complexities of $2^{4n/5}$, while in the current approach, either the time complexity or the memory complexity is not less than 2^n. We conclude that if the IV length of a stream cipher is less than 1.5 times the key length, there exists an attack on the cipher with data, time, and memory complexities less than the complexity of exhaustive key search.

Keywords: Time-Memory-Data Tradeoff attacks, Stream ciphers, IV initialization.

* This is the full version of a paper accepted to Information Processing Letters.

** The research described in this paper was done while the author was in the Department of Electrical Engineering ESAT/SCD-COSIC, Katholieke Universiteit Leuven. This work was supported in part by the Concerted Research Action (GOA) Ambiorics 2005/11 of the Flemish Government and by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy).

*** This author is supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

1 Introduction

The problem of inverting a one-way function is a basic problem in cryptanalysis. For example, the problem of deducing the encryption key from a plaintext/ciphertext pair can be modeled as such. Two extreme solutions of the problem are exhaustive key search and the table attack (also called the dictionary attack). In exhaustive key search, the time complexity of the attack is N encryptions (where N is the number of possible keys), the memory requirement is negligible, and there is no preprocessing phase. In the table attack, the time complexity of the on-line phase of the attack is negligible, while the memory requirement is N memory cells, and a one-time preprocessing phase with time complexity of N encryptions is required.

In [13] Hellman presented a trade-off between these two attacks. Hellman's attack uses pre-computed tables of total size M which allow to reduce the on-line time complexity T. The values of M and T satisfy the relation $N^2 = TM^2$ (up to logarithmic factors), and thus, a convenient choice of M and T is $M = T = N^{2/3}$. The attack also requires a pre-computation phase with time complexity of N encryptions. This phase is usually neglected in the treatment of TMTO attacks, since once it is performed, its results can be re-used for multiple attacks (each much faster than exhaustive key search). In [6] Biryukov and Shamir showed that if the attacker has access to multiple data points, the tradeoff curve obtained in Hellman's attack can be improved. If the number of available data points is D, the attacker can produce an attack which satisfies $N^2 = TM^2D^2$, and has a preprocessing step of N/D operations. However, this attack is applicable only for $T \geq D^2$.[1]

Hellman's ideas, originally proposed for attacking DES, were extended to various cryptosystems. Probably the most successful extension is the application of TMTO to stream ciphers. Several stream ciphers were broken using TMTO [7, 17], and resistance to TMTO attacks is considered an important criterion in stream cipher design [10].

TMTO attacks on stream ciphers can be divided into two classes, according to the one-way function the attacker tries to invert. The first class of attacks (denoted in the sequel *Scenario A* attacks) tries to invert the function (Internal State $\rightarrow$ Output Prefix) [1, 6, 12]. The second class of attacks (denoted in the sequel *Scenario B* attacks) tries to invert the function (Secret Key $\rightarrow$ Output Prefix) [14, 15].

In this paper we consider Scenario B attacks on stream ciphers with IV. We propose a new approach to attacks of this class: Instead of inverting the function ((Key,IV) $\rightarrow$ Output Prefix), we suggest to choose in advance several IVs, compute the TMTO tables for the function (Key $\rightarrow$ Output Prefix) for each of the chosen IVs, and apply Hellman's original attack on encryptions using the chosen IVs.

In the new approach, the state space of the function to be inverted is reduced significantly, compared to the former approach. On the other hand, the attacker

[1] In some cases, the restriction $T \geq D^2$ can be removed, as discussed in Section 2.

loses the ability to exploit any set of multiple data points, which could be obtained in the former approach from encryptions under the same secret key with different IVs.

The new approach is similar to the BSW-sampling technique, proposed in [7, 6] for Scenario A attacks. Whereas the BSW-sampling depends crucially on the specific structure of the cipher, our attack is general and is applicable to any stream cipher with a publicly known IV.

We show that the tradeoff curve obtained for the new approach is $N^2 = TM^2D^2$, where N is the number of possible keys multiplied by the number of possible IVs, and D is the amount of data available to the attacker. Hence, the tradeoff curve is the same as the tradeoff curve obtained by the current approach.

However, we show that the new approach has several advantages. The main advantage is that while the current approach allows to obtain the tradeoff curve $N^2 = TM^2D^2$ only for $T \geq D^2$, the new approach allows to obtain this curve for all values of D. For example, if the key size is 64 bits, the IV size is 40 bits, and the amount of available data is $D = 2^{32}$, it is possible to mount a TMTO attack with $T = M = 2^{48}$, while in the current approach, the parameters of the attack must satisfy the inequality $TM \geq 2^{104}$.

Moreover, we show that if the length of the secret key is n bits and the IV length is less than $1.5n$ bits, the new approach allows to mount an attack with data, memory, and time complexities less than 2^n.[2] In particular, setting the IV length to be equal to the key length does not ensure an n-bit security against TMTO attacks, contrary to [5, 9, 14, 15]. For example, if both the secret key and the IV are of length n, the new approach allows to mount an attack with data, time, and memory complexities of $2^{4n/5}$.

Finally, we discuss cases where the IV update scheme is (even partially) deterministic. We show that this common approach is less secure with respect to TMTO attacks than picking the IV at random each and every time.

The rest of the paper is organized as follows: In Section 2 we briefly describe the basic ideas of time-memory-data tradeoffs. In Section 3 we present our new approach and its advantages. We summarize the paper in Section 4.

2 A Brief Overview of Time-Memory-Data Tradeoffs

In this section we briefly overview some of the previously known TMTO attacks. A more extensive summary of TMTO attacks can be found in [4].

2.1 Hellman's and Oechslin's TMTO Attacks

We start with the basic TMTO attack of Hellman [13]. Let $f : \{0, 1, \ldots, N-1\} \rightarrow \{0, 1, \ldots, N-1\}$ be the function that the attacker tries to invert.

[2] We note that a similar claim regarding Scenario A attacks is presented in [2]: In order to ensure an n-bit security against Scenario A attacks using BSW-sampling, the state size has to be at least 2.5 times larger than the key size. In our case, the overall size of the key and the IV has to be at least 2.5 times larger than the key size.

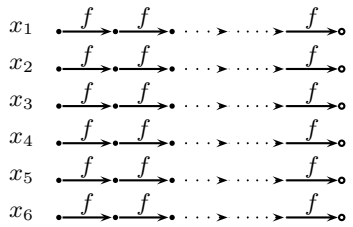

Fig. 1. Constructing Hellman's Table

The pre-processing phase of Hellman's attack consists of constructing several tables. To construct each table, the attacker chooses m random starting points, and from each starting point x she computes the chain $SP = x, f(x), f^2(x) = f(f(x)), \ldots, f^t(x) = EP$, as shown in Figure 1. The pairs (EP, SP) are stored in a table. The attacker constructs t such tables $T_0, \ldots T_{t-1}$, each for a different function f_i that is usually a slight modification of the original f (e.g., a permutation of the bits). In the on-line phase of the attack, the attacker is given $z = f(y)$ and has to find y. For all $0 \leq i \leq t-1$, she applies f_i repeatedly to $f_i(y)$ (that can be easily computed given $f(y)$) to get the sequence $f_i(y), f_i^2(y), \ldots, f_i^t(y)$, for each new value the attacker checks whether the obtained value is an end point in the table T_i. If a value appears as an end point in the table, the attacker takes the corresponding starting point and applies f_i sequentially, until $f_i(y)$ is reached. The point encountered just before $f_i(y)$ is supposedly y.

The time complexity of the attack is t^2 applications of f and t^2 database accesses (or $T = t^2$), and the memory required for the attack is $M = mt$. Each of the tables "covers" mt points (m chains of length t each), and thus all the t tables cover about mt^2 states. Since the tables should cover most of the N possible states, we have $N \approx (mt)t = mt^2$, and hence $N^2 = TM^2$ is the tradeoff curve obtained for this attack. The time complexity of the pre-processing phase is N, but as we noted before, this phase is usually neglected in the analysis of TMTO attacks.

In 2003, Oechslin [16] presented a different method to construct the tables in the TMTO attack. In the new method, a single table (called the *Rainbow table*) is constructed. This time, mt starting points are chosen, and the constructed chains are of the form $x, f_0(x), f_1(f_0(x)), f_2(f_1(f_0(x))), \ldots, f_{t-1}(\ldots(f_0(x)))$, where the functions f_i are the same as used in Hellman's scheme. This technique reduces the effect of collisions in the table, and hence allows to cover most of the state space by a single table.

On the other hand, the on-line phase of the attack is more complicated. First, the attacker checks whether $f_{t-1}(y)$ (computed from the given $f(y)$) appears as an end point in the table. If not, she computes $f_{t-1}(f_{t-2}(y))$ and checks whether this value appears in the table. If not, she computes $f_{t-1}(f_{t-2}(f_{t-3}(y)))$, and so on. Once an end point is encountered, the attack proceeds as Hellman's attack.

The time complexity of the attack is $1+2+\ldots+(t-1) \approx t^2/2$ applications of f and t database accesses, and the memory requirement is mt. Since this time most of the states are covered by the single table, we have $N \approx mt^2$, and hence the obtained tradeoff curve is $N^2 = 2TM^2$.

2.2 Time-Memory-Data Tradeoff Attacks

In 2000, Biryukov and Shamir [6] showed that if the attacker has access to multiple data points, the tradeoff curve obtained in Hellman's attack can be improved.[3] It is important to note that the multiple data points stand for pairs of (unknown input, known output) to the function the attacker tries to invert, and the attacker is satisfied with finding any one of the unknown inputs. If the number of available data points is D, the attacker can construct tables that cover only N/D of the N states, and by the birthday paradox one of the data points is supposed to fall into the covered states. In the on-line phase of the attack, the attacker repeats the attack for all the data points, and once one of the points is covered by the table, the secret key can be retrieved.

In this case there are only t/D tables, and thus the time complexity of the attack remains t^2 while the memory requirement is reduced to mt/D. Since this time we have $N/D \approx mt^2$, the obtained tradeoff curve is $N^2 = TM^2D^2$. The time complexity of the pre-processing phase is reduced to N/D.

It is noted in [6] that the tradeoff curve $N^2 = TM^2D^2$ can be obtained only if $T \geq D^2$, since only in this case at least one "full" table consisting of m chains of length t each, where $mt^2 \approx N$, can be constructed. If $T < D^2$ and a single "smaller" table is used in the attack, the resulting time complexity is Dt and the memory requirement is m. However, since in this case we have $N/D \approx mt$, the obtained tradeoff curve is $TM^2D^2 = D^3tm^2 > D^2t^2m^2 = D^2(N/D)^2 = N^2$ (where we used the inequality $D > t$ following from the assumption $D^2 > T$).

In [5] the technique presented in [6] is extended to the case $T < D^2$ and various tradeoff curves are suggested. However, all the suggested curves are inferior to the curve $N^2 = TM^2D^2$.

Another technique allowing to extend the tradeoff curve $N^2 = TM^2D^2$ to the range $T < D^2$ is the *BSW-sampling* technique, introduced in [7] and examined in [6]. This technique can be applied if for some easily distinguishable subset of the output values of f, the inputs leading to these outputs can be efficiently enumerated.

Assume, for example, that $N = 2^n$, and that the set

$$A = \{x : \text{ The k least significant bits of f(x) are zeros } \}$$

can be enumerated efficiently. In this case, the attacker considers in the pre-computation phase only the values of $x \in A$. For each such value, she associates two $(n-k)$-bit indices: The *short name* that is used by the enumeration procedure

[3] We note that the idea of exploiting multiple data points in TMTO attacks was first presented in [1, 12]. Biryukov and Shamir were the first to combine this idea with Hellman's attack.

to define x, and the *output name*, that corresponds to the $n-k$ most significant bits of $f(x)$. Then, the attacker considers the function $g : \{0,1\}^{n-k} \to \{0,1\}^{n-k}$, defined by $g($ short name $) =$ corresponding output name . For this function, the attacker constructs the Hellman tables. In the on-line phase of the attack, the attacker considers only data points of the form $y \in f(A)$, and uses the inversion of g to find the pre-image of one of them.

As a result, the data available to the attacker is reduced from D to $D/2^k$, while the inverted function is reduced from an n-bit to n-bit function to an $(n-k)$-bit to $(n-k)$-bit function. It is shown in [6] that the tradeoff curve obtained by the BSW-sampling is $N^2 = TM^2D^2$ (like the original curve), but the range of applicability of the curve is increased to $T \geq (2^{-k}D)^2$. It is also shown that BSW-sampling can be used in Scenario A attacks on stream ciphers, and the effectiveness of the technique (i.e., the maximal possible k for which the efficient enumeration is possible) depends heavily on the structure of the cipher.

The usage of multiple data points in the Rainbow scheme is examined in [3, 5]. In [5] it is shown that a direct application of the Rainbow scheme in the case of multiple data points leads to the inferior tradeoff curve $N^2 = TM^2D$. In [3] two more complicated algorithms are presented. The first leads to the tradeoff curve $N^2 = TM^2D^2$, thus losing the factor two advantage of the Rainbow scheme. The second yields the curve $N^2 + ND^2M = 2TM^2D^2$. This allows to save the factor two advantage if $T \gg D^2$, but if T is close to D^2, most of the gain of the Rainbow scheme is lost.

2.3 TMTO Attacks on Stream Ciphers

TMDTO attacks on stream ciphers can be divided into two classes, according to the one-way function the attacker tries to invert: Inverting the internal state to output transformation, and inverting the (key, IV) to output transformation.

The first class of attacks (i.e., Scenario A attacks) is presented in [1, 7, 6, 12]. These attacks try to invert the function (Internal State $\to$ Output Prefix). In attacks of this class the trade-off curve provided by Hellman's basic attack can be improved using multiple data points.[4] Each additional output bit provides the attacker with an additional data point, and if the attacker uses D such points, the tradeoff curve can be improved to $N^2 = TM^2D^2$, for $T \geq D^2$. Attacks of this class were used to break several stream ciphers, including A5/1 and LILI-128 [7, 17]. A countermeasure against Scenario A attacks, deployed in most modern stream ciphers, is using a large internal state that makes this class of attacks less favorable than exhaustive key search.

The second class of attacks (i.e., Scenario B attacks) is presented in [14, 15]. These attacks try to invert the function (Secret Key $\to$ Output Prefix).

Scenario B attacks cannot use multiple data points from the same output stream, but the on-line time complexity of the attacks is less than that of exhaustive key search, independently of the size of the internal state. The security of a stream cipher against Scenario B attacks is increased if the cipher uses a

[4] We remark that the attacks presented in [1, 12] are not based on Hellman's technique.

publicly known Initial Value (IV). If the attacker does not know the IV in advance, she cannot use it in the pre-processing phase of the TMTO attack. The current approach in this situation is to treat the IV as part of the secret key [5, 14, 15]. As a result, the amount of secret key material is increased, but at the same time, the attack can use multiple data points. The data points are obtained from encryptions using the same secret key and different IVs.[5] As a result, the tradeoff curve $N^2 = TM^2D^2$, for $T \geq D^2$, is again possible, where N is the number of possible keys multiplied by the number of possible IVs. In order to prevent Scenario B attacks, as well as other attacks, it is suggested in [5, 9, 14, 15] to require stream ciphers to have IV at least as long as the secret key.

3 A New Approach to TMTO Attacks on Stream Ciphers Using Initial Values

In this section we present our new approach to Scenario B TMTO attacks on stream ciphers with IV.

We observe that the current approach, i.e., treating the IV as part of the secret key and trying to invert the function ((Key,IV) $\rightarrow$ Output Prefix) is not optimal, since it does not exploit the fact that the IV is publicly known. We propose to choose in advance several IVs and try to invert the function (Key $\rightarrow$ Output Prefix) for the chosen IVs.[6]

Denote the number of possible secret keys by $K = 2^k$ and the number of possible IVs by $V = 2^v$. Assume that the amount of data (i.e., output streams generated under the secret key with different IVs) available in the on-line phase of the attack is $D = 2^d \leq V$.

In the pre-processing phase of the attack, the attacker chooses V/D IVs. For each chosen IV_i, the attacker prepares Hellman's tables (or Rainbow tables) to invert the function $f:$ Key $\rightarrow$ Output Prefix for IV_i. In the on-line phase of the attack, the attacker waits until IV_i is used for some i and applies the TMTO attack with the tables prepared for that IV_i.

The new approach is similar to applying the BSW-sampling technique [7] to the former approach. In the former approach, the inverted function, $f : (Key, IV) \rightarrow$ Output Prefix , is defined on a space of 2^{k+v} values. However, since the IV is public, the set of outputs corresponding to inputs of the form:

$$B = \{x = (Key, IV) : \text{ The } d \text{ least significant bits of IV are equal to a fixed value } (x_1, \ldots, x_d)\}$$

is easy to enumerate. Indeed, these are the output streams produced using IVs whose d least significant bits are $(x_1, \ldots, x_d)$. Hence, the attacker can reduce the

[5] At this stage we consider only attacks mounted on a single key. The scenario in which the attack is mounted on several secret keys simultaneously and the attacker is satisfied with retrieving only one out of the several secret keys is discussed in Section 3.1.

[6] We note that possibly this idea is implicitly mentioned in Section 8 of [8]. However, this is the first paper which analyzes this approach.

space of the function she tries to invert to only 2^{k+v-d} values, while using only a single data point instead of $D = 2^d$ data points.[7]

Using the notations introduced in Section 2, the time complexity of the attack is t^2 and the memory requirement is mtV/D. The tables cover most of the key space, and thus $mt^2 \approx K$. Hence, the resulting tradeoff curve is $TM^2 = t^4m^2V^2/D^2 = K^2V^2/D^2 = (KV)^2/D^2$, or equivalently $TM^2D^2 = (KV)^2$.

Using the previous approach, i.e., trying to invert the function ((Key,IV) $\rightarrow$ Output Prefix), results in the same tradeoff curve $TM^2D^2 = (KV)^2$. However, it appears that the new approach has several important advantages over the former one:

1. **Attacking Ciphers with Long IVs:** Since in the new approach the underlying TMTO attack does not use multiple data points, the tradeoff curve $N^2 = TM^2D^2$ can be obtained without restrictions on the parameters, i.e., even for $T < D^2$. Hence, assuming $T = M = D$, the complexity of the attack is less than that of exhaustive key search as long as $N^2 < K^5$, or equivalently, $V^2 < K^3$. Therefore, if the key length is n bits, the cipher offers n-bit security w.r.t. TMTO attacks only if the IV length is at least $1.5n$ bits. In the current approach, the parameters of the attack must satisfy the inequality $MT \geq N$ (see [5, 6]), and hence if the IV length is at least n bits, either the time or the memory complexity of the attack is not less than 2^n, regardless of the amount of available data. Moreover, since in the former approach the optimal curve can be obtained only for $T \geq D^2$ (see [5, 6]), the new approach leads to a better tradeoff curve whenever the amount of available data is greater than $2^{n/2}$, for an n-bit secret key. This advantage is particularly significant in modern stream ciphers, where the IV size is large, where the new approach is able to use more available data then the previous one.[8]
2. **Optimal tradeoff for multiple data points in the Rainbow scheme:** The new approach allows to use Rainbow tables in the attack instead of the standard Hellman's tables, without losing the advantage of the Rainbow's tradeoff curve. This is possible since the underlying TMTO attack does not use multiple data points at all (which prevents using the full power of the rainbow tables [3, 5]). Therefore, the new approach allows to use the Rainbow tables also in attacks on stream ciphers with IVs, gaining an improved time complexity and reduction in the rate of false alarms [16].[9]

[7] Note that while in the BSW-sampling technique, the attacker enumerates inputs that lead to outputs of a prescribed form, in our attack the attacker enumerates outputs that result from inputs of a prescribed form. It is easy to see that in our case, the attacker can "identify" the instance much faster.

[8] This advantage holds also for the original BSW-sampling technique, as noted in [6].

[9] As noted in [3], the advantage of the Rainbow scheme over the standard Hellman's scheme can be questioned. The comparison between the two schemes is out of the scope of this paper. Our claim is that using our technique, both schemes can be used also in attacks on stream ciphers with IVs.

3. **Reduction in the memory requirements:** The new approach reduces the memory requirements of the attack. Since the function we try to invert is (Key → Output Prefix), the number of memory bits needed for each chain in the table is $2\lceil \log K \rceil$. When inverting ((Key,IV) → Output Prefix) each chain requires $2\lceil \log K + \log V \rceil$ bits. If the IV length is equal to the key length (as recommended in [9]), this difference reduces the memory requirements of the attack by factor 2.[10]
4. **Reduction in the duration of active eavesdropping:** In some cases the new approach allows to reduce the duration of active eavesdropping performed in the attack. Assume that only the first IV is randomly chosen, and then the IV is incremented sequentially (as recommended in [9]). In this case, the attacker can examine the first used IV and find out when one of the chosen IVs will be used. Then, the attacker has to eavesdrop only when the required IV is used. In the standard approach, such an attack requires eavesdropping to all the data.
5. **Exploiting uneven distribution of the IVs:** If the IVs are not chosen properly, i.e., the distribution of the IVs is not uniform, the new approach allows to exploit this weakness easily. The attacker prepares the tables for the IVs that are most frequently used, and thus reduces the memory and data complexities of the attack.[11] For example, if the first used IV is known to the attacker in advance (e.g., fixed to zero), the attacker prepares the tables only for this IV, and thus reduces the effect of the IV on the cipher's security. It seems that the standard approach also can exploit improper usage of IVs, but the algorithm becomes more complicated.

On the other hand, the new approach has one drawback: The success probability of the attack might be lower than the success probability in the former approach. In our new approach, the attack succeeds if the following two conditions are satisfied:

1. Amongst the D encryptions available to the attacker, there is an encryption using one of the V/D chosen IVs.
2. The secret key used with the chosen IV is covered by the tables prepared for this IV.

Assuming the attacker chooses the parameters such that $N = TM^2$ (or $N = TM^2D^2$), the probability that the obtained sample is covered by one of the tables is about 63%. If all the IVs are chosen randomly and independently, then the probability that one of the IVs that are covered is encountered in the D samples is 63%, and hence, the total success probability of our approach in this case is 40%. For comparison, the standard approach yields a success rate of 63% while using $\lceil \log K + \log V \rceil / \lceil \log K \rceil$ times more memory. When the size of the key and the IV is the same, both approaches yield 40% success rate for the same amount of memory.

[10] This advantage holds also for the original BSW-sampling technique.
[11] A similar approach was used in [7], under the name *biased birthday attack*.

However, in the majority of protocols where stream ciphers are used with IVs, the first IV is chosen randomly (or even in a deterministic manner), and the IV is incremented in a deterministic manner between the different packets. In this case, by choosing the IVs appropriately, the attacker can assure that one of the chosen IVs is used in the available encrypted streams, thus, addressing the first of the two issues. Hence, in most of the applications, it is expected that the success rate of the new approach is equal to that of the former approach while the memory requirements are reduced.

3.1 Attacking Several Secret Keys Simultaneously

If the attacker mounts the attack on several secret keys simultaneously and is satisfied with retrieving only one of them, the TMTO attack benefits from the ability to use multiple data points. In the standard approach, where the inverted function is ((Key,IV) $\rightarrow$ Output prefix), encryptions under different keys can be naturally used as multiple data points in the attack. In the new approach, the attack strategy depends on the amount of available data.

Assume that the data available to the attacker consists of encryptions under D_K different keys, where each key is encrypted under D_V IVs. In this case, the tradeoff curve obtained by the standard approach is $TM^2(D_K D_V)^2 = K^2V^2$.

In the new approach, if $D_K D_V \leq V$, the attacker chooses $V/D_K D_V$ IVs and pre-computes the tables for inverting the function (Key $\rightarrow$ Output prefix) for each of the chosen IVs. The data is supposed to contain an encryption using one of the chosen IVs, and once such instance is encountered, the corresponding key can be retrieved.

If $D_K D_V \geq V$, the attacker chooses a single IV and prepares the tables for it. However, since it is expected that the data contains $D_K D_V/V$ encryptions using the chosen IV, the constructed tables have to cover only $K/(D_K D_V/V)$ of the possible keys, and hence the memory complexity of the attack is reduced. Note that since in this case the underlying attack is a TMDTO attack, the optimal tradeoff curve can be obtained only if $T \geq (D_K D_V/V)^2$.

In both cases, the obtained tradeoff curve is $TM^2(D_K D_V)^2 = K^2V^2$, as in the standard algorithm. In the case $D_K D_V \leq V$, all the advantages of the new approach are preserved. In the case $D_K D_V \geq V$, the first two advantages are weakened, while the other three are preserved.[12] Therefore, the new approach is better than the former one also in the multiple key case.

4 Summary and Conclusions

In this paper we presented a new approach to Time-Memory-Data tradeoff attacks on stream ciphers with initial value. Instead of treating the IV as part

[12] On the other hand, for $D_K D_V \gg V$, the success probability of the new approach is equal to that of the standard approach, since the chosen IV appears in the data with overwhelming probability.

of the secret key material and trying to invert the function ((Key,IV) → Output prefix), we propose to fix several IVs in advance and attempt to invert the function (Key → Output prefix). Our technique is similar to the BSW-sampling technique, originally applied to attacks trying to invert the function (State → Output prefix). We showed that while the resulting tradeoff curve in the new approach is the same as the curve obtained by the former approach, the new approach has several advantages over the former one, such as less restrictions on the parameters of the tradeoff.

Using our technique we showed that a stream cipher whose IV length is the same as the key length does not provide security equal to the key length. There are seemingly three ways to counter our results:

1. Require that the IV length will be at least 1.5 times larger than the key length.[13]
2. Allow the usage of a limited number of IVs with a single key. For an n-bit key, only $2^{n/2}$ IVs should be allowed.[14]
3. Accept the fact that n-bit security does not fully cover time-memory-data tradeoff attacks (like is practically the case in block cipher design these days).

We conclude that the new approach allows to exploit (to some extent) the fact that the IV is known to the attacker. It is tempting to find further ways to exploit this knowledge, and thus to reduce the influence of the IV on the cipher's security. We expect these results to affect the way IVs are used in various protocols, to counter our findings.

Acknowledgments

We would like to thank Eli Biham and Adi Shamir for the fruitful discussions and their suggestions. We would also like to thank the anonymous referees of Information Processing Letters for their constructive remarks.

References

1. Steve H. Babbage, *Improved "exhaustive search" attacks on stream ciphers*, IEE European Convention on Security and Detection, IEE Conference publication 408, pp. 161–165, IEE, 1995.
2. Steve H. Babbage, Matthew Dodd, *Specification of the Stream Cipher Mickey 2.0*, submitted to eStream, 2006. Available on-line at: http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey_p3.pdf.
3. Elad Barkan, Eli Biham, Adi Shamir, *Rigorous Bounds on Cryptanalytic Time/Memory Tradeoffs*, Advances in Cryptology, proceedings of Crypto 2006, Lecture Notes in Computer Science 4117, pp. 1–21, Springer-Verlag, 2006.

[13] A similar design criteria is proposed in the specification of the stream cipher Mickey 2.0 [2]. The internal state of Mickey 2.0 was set to be 2.5 times the key size to prevent BSW-sampling attacks on the internal state being faster than exhaustive search.

[14] This design criteria is proposed in the specification of the stream cipher Mickey 2.0 [2].

4. Elad Barkan, *Cryptanalysis of Ciphers and Protocols*, Ph.D. Thesis, Technion, 2006.
5. Alex Biryukov, Sourav Mukhopadhyay, Palash Sarkar, *Improved Time-Memory Tradeoffs with Multiple Data*, proceedings of Selected Areas in Cryptography 2005, Lecture Notes in Computer Science 3897, pp. 245–260, Springer-Verlag, 2006.
6. Alex Biryukov, Adi Shamir, *Cryptanalytic Time/Memory/Data Tradeoffs for Stream Ciphers*, Advances in Cryptology, proceedings of ASIACRYPT 2000, Lecture Notes in Computer Science 1976, pp. 1–13, Springer-Verlag, 2000.
7. Alex Biryukov, Adi Shamir, David Wagner, *Real Time Cryptanalysis of A5/1 on a PC*, proceedings of FSE 2000, Lecture Notes in Computer Science 1978, pp. 1–18, Springer-Verlag, 2001.
8. Johan Borst, Bart Preneel, Joos Vandewalle, *On the Time-Memory Tradeoff Between Exhaustive Key Search and Table Precomputation*, proceedings of the 19th Symposium on Information Theory in the Benelux, Veldhoven (NL), pp. 111–118, 1998.
9. Christophe De Canniere, Joseph Lano, and Bart Preneel, *Comments on the Rediscovery of Time Memory Data Tradeoffs*, eSTREAM, ECRYPT Stream Cipher Project, Report 2005/040.
10. ECRYPT, *Call for Stream Cipher Primitives*, version 1.3, 12.4.2005. Available online at *http://www.ecrypt.eu.org/stream/call/*.
11. Amos Fiat, Moni Naor, *Rigorous Time/Space Trade-offs for Inverting Functions*, SIAM Journal of Computing, vol. 29, No. 3, pp. 790–803, 1999.
12. Jovan Dj. Golic, *Cryptanalysis of Alleged A5 Stream Cipher*, Advances in Cryptology, proceedings of EUROCRYPT '97, Lecture Notes in Computer Science 1233, pp. 239–255, Springer-Verlag, 1997.
13. Martin E. Hellman, *A Cryptanalytic Time-Memory Tradeoff*, IEEE Transactions on Information Theory, Vol. 26, No. 4, pp. 401–406, 1980.
14. Jin Hong, Palash Sarkar, *Rediscovery of Time Memory Tradeoffs*, IACR Eprint Report 2005/090, 2005. Available on-line at *http://eprint.iacr.org/2005/090*.
15. Jin Hong, Palash Sarkar, *New Applications of Time Memory Data Tradeoffs*, Advances in Cryptology, proceedings of Asiacrypt 2005, Lecture Notes in Computer Science 3788, pp. 353–372, Springer-Verlag, 2005.
16. Philippe Oechslin, *Making a Faster Cryptanalytic Time-Memory Trade-Off*, Advances in Cryptology, proceedings of CRYPTO 2003, Lecture Notes in Computer Science 2729, pp. 617–630, Springer-Verlag, 2003.
17. Markku-Juhani Olavi Saarinen, A Time-Memory Tradeoff Attack Against LILI-128, proceedings of FSE 2002, Lecture Notes in Computer Science 2365, pp. 231–236, Springer-Verlag, 2002.
18. Michael J. Wiener, *The Full Cost of Cryptanalytic Attacks*, Journal of Cryptology, Vol. 17, No. 2, pp. 105–124, 2004.