On designing secure small-state stream ciphers against time-memory-data tradeoff attacks

Vahid Amin Ghafari, Honggang Hu, and Fujiang Lin
School of Information Science and Technology, University of Science and Technology of China
(USTC), Hefei, China, 230027
{vahidaming, hghu2005, linfj}@ustc.edu.cn

Abstract A new generation of stream ciphers, small-state stream ciphers (SSCs), was born in 2015 with the introduction of the Sprout cipher. The new generation is based on using key bits not only in the initialization but also continuously in the keystream generation phase. The new idea allowed designing stream ciphers with significantly smaller area size and low power consumption. A distinguishing time-memory-data tradeoff (TMDTO) attack was successfully applied against all SSCs in 2017 by Hamann et al. [1]. They suggested using not only key bits but also initial value (IV) bits continuously in the keystream generation phase to strengthen SSCs against TMDTO attacks.

Then, Hamann and Krause [2] proposed a construction based on using only IV bits continuously in packet mode. They suggested an instantiation of an SSC and claimed that it is resistant to TMDTO attacks. We point out that storing IV bits imposes an overhead on cryptosystems that is not acceptable in many applications. More importantly, we show that the proposed SSC remains vulnerable to TMDTO attacks.

To resolve security threat, the current paper proposes constructions, based on storing key or IV bits, that are the first to provide full security against TMDTO attacks. It is possible to obtain parameters for secure SSCs based on these suggested constructions. Our constructions are a fruitful research direction in stream ciphers.

Keywords: Stream cipher, Ultra-lightweight, Small-state, Sprout, Fruit, Plantlet, Distinguishing attack, Time-memory-data tradeoff attack.

1. Introduction

Small-state stream ciphers (SSC) research was started due to an important fact: key bits should be stored in cryptosystems after initialization in many applications [3]. The storing of key bits is necessary so they can be reused by different initial values (IV) in many applications, and also it is unavoidable to save key bits in a fixed memory in some applications (storing one fixed key is enough forever in some applications such as RFID systems and the SIM cards of mobile phones) [4,5]. The internal state size of stream ciphers must be at least twice the size of the security level in order to be resistant to TMDTO attacks [6]. It is a good idea to exploit key bits as a part of the internal state and design stream ciphers with smaller internal states [3]. This idea allows the design of ultra-lightweight stream ciphers [3,7,4,8].

Before the introduction of the first SSC (i.e., Sprout [3]) in 2015, all of the internal states were volatile memory items. In SSCs, the internal states consist of volatile memory items and also fixed memory items. The fixed items can be keys, IVs or both, and they continuously participate in the internal state updating and keystream generation.

Unfortunately, SSCs were not as strong as expected against a type of TMDTO attacks, i.e., a TMDTO distinguishing attack [1]. A construction was proposed by Hamann et al. [1] to strengthen SSCs against TMDTO distinguishing attack: continuously using key and IV bits after initialization in the keystream generation phase. Then, two papers were published by Hamann et al., who claimed that the best construction is continuously using only IV bits after initialization in the keystream generation phase [2,9]. They proposed a construction to guarantee security lower bounds of SSCs against all types of TMDTO attacks [2]. They specified the parameters of an SSC and claimed that it is resistant to TMDTO attacks. We show that the corresponding cipher is vulnerable to TMDTO attacks.

Furthermore, the construction requires storing IV bits in cryptosystems. Storing IV bits (unlike key bits) imposes overhead in many cryptosystem applications. In [1], Hamann et al. stated that storing IV bits provides a notable benefit for cryptosystems. The benefit can be employed to avoid using the same IV twice under the same key, which is a problem that could happen in old cryptosystems with small IV spaces, for example in A5/1 with 22 IV bits. Note that the size of the IV should be the same as the size of the key theoretically for providing key length security against TMDTO attacks (the space size of key plus IV should be more than twice the space size of the key) [10], and modern cryptosystem algorithms produce IV bits such that the same IV is never produced under the same key. Also, IV bits are usually produced elsewhere in the system (from some parameters of systems, for example from packet numbers) and transferred to the encryption section. Thus, storing IV bits requires extra memory in some cryptosystems, and that contradicts the design philosophy of SSCs. Note that continuously using secret key can provide more security (rather than continuously using public known IV) against other attacks.

In fact, the construction is unrealistic for many applications, and we show that it cannot provide security against TMDTO attacks as discussed in [2,9]. Continuously using all bits of the key and IV together (as described in [1,4]) is also unrealistic and imposes an unsustainable overhead on cryptosystems in many cases.

We discuss various application scenarios for full security against TMDTO attacks, and we propose five constructions for obtaining secure design parameters in different conditions. In every construction, based on the cryptosystem conditions, it is possible to use IV bits, key bits, or a combination of these as a part of the internal state continuously.

Designers should choose one of the five constructions based on two factors. The first factor is related to how many times key (or IV) bits are used in the encryption, and the second factor is related to the limitation on the number of the produced keystreams per key (or IV). Our proposed constructions cover various applications, and it is possible to extract desirable parameters for secure design in every construction. Our results show that SSCs are a hopeful research direction in the future.

The paper is organized as follows. Notation and preliminaries are presented in Section 2, and TMDTO attacks on a previously proposed Continuous-IV-Use Construction are described in Section 3. Then, in Section 4, we present constructions for designing secure SSCs in different applications. Finally, we conclude the paper in Section 5.

2. Notation and preliminaries

We use the following notation:

- IV Length (IVL): Length of IV in bits
- Continuously used IV (CIV): A section of the IV which is used in the initialization and also continuously in the keystream generation phase
- Continuously used IV Length ($CIVL$): Length of CIV in bits
- Key Length (KL): Length of key in bits
- Continuously used Key (CK): A section of the key which is used in the initialization and also continuously in the keystream generation phase
- Continuously used Key Length (CKL): Length of CK in bits
- State Length (SL): Internal state length in bits
- Volatile State (VS): A section of the internal state which can be changed in every clock (i.e., it is volatile)
- Volatile State Length (VSL): Length of VS in bits
- Packet length per IV (PIV): The maximum number of keystream bits that can be produced per IV
- Packet length per Key (PK): The maximum number of keystream bits that can be produced per key
- Packet length per Key/IV pair (PKI): The maximum number of keystream bits that can be produced per key/IV pair

Every keystream has a length of 2^{SL} bits and it is obvious that two keystreams may differ only in one bit. We suppose that the state transition functions are bijective and the period of transition functions are very big (these assumptions are conceivable for any good stream ciphers). As mentioned, the length of the key and IV should be theoretically equal for providing key length security against TMDTO attacks (i.e., $IVL = KL$) [10]. This is expected when we consider a one-way function from key and IV bits to keystreams (the domain space size of the one-way function should be twice the size of the security level to resist against generic TMDTO attacks).

Until now, three type constructions for SSCs have been proposed:

- Continuous-Key/IV-Use construction: A stream cipher which uses key and IV bits not only in the initializations but also continuously in the keystream generations (as a part of the internal state) [4,1].
- Continuous-IV-Use construction: A stream cipher which uses IV bits not only in the initializations but also continuously in the keystream generations, and key bits are used only in the initializations [2,9].

- Continuous-Key-Use construction: A stream cipher which uses key bits not only in the initializations but also continuously in the keystream generations, and IV bits are used only in the initializations [5,3,7].

This paper uses Continuous-Key/IV-Use construction as the general construction, and designers can obtain parameters for designing secure SSCs against TMDTO attacks.

3. TMDTO attacks on a previously proposed Continuous-IV-Use Construction

Hamann and Krause claimed that the security of Continuous-IV-Use construction is $VSL - log_2(PKI)$ bits (where $log_2(PKI) = IVL - CIVL$) against TMDTO attacks [2]. As the security level against all types of attacks is considered to be KL bits, the volatile state length (VSL) will be $KL + IVL - CIVL$. They discussed the following parameters for an SSC.

$$VSL\ =\ 100\ ,KL\ = IVL\ =\ 80\ ,PKI\ = 2^{20}, CIVL = 60, CKL\ = 0$$

They claimed that the corresponding stream cipher is resistant to TMDTO attacks. We show that the security of the corresponding stream cipher against TMDTO distinguishing attacks is not as promised. It is obvious that there are 2^{20} IVs that produce the same CIV (from 80 bits of IVs, only 60 bits are CIV). In other words, as every IV can produce at most 2^{20} internal states (and keystreams), there are 2^{40} internal states that have the same CIV. An attacker saves half of the keystreams under the same CIV in a searchable table. Then, the attacker searches for a collision between the remaining keystreams and the keystreams in the searchable table. Note that the attacker saves half of the keystreams in the online phase of attack and searches with another half of the keystreams to find a collision. As there are 100 unknown bits (i.e., $VSL = 100$) for the attacker in the internal state, the probability of failure (i.e., the attacker cannot find any collision) is:

$$\left(1-\frac{2^{39}}{2^{100}}\right)^{2^{39}} = \left(1-\frac{1}{2^{61}}\right)^{2^{39}}$$

Note that there are 2^{160} different internal states, but the attacker only considers the states with the same CIV. The CIV is known to the attacker, who can receive keystreams corresponding to different VSs and the same CIV. Now, the attacker repeats this process 2^{22} times (i.e., the attacker saves 2^{22} times 2^{39} keystreams in a table and searches for a collision). The probability of failure is:

$$\left(\left(1-\frac{1}{2^{61}}\right)^{2^{39}}\right)^{2^{22}} = \left(1-\frac{1}{2^{61}}\right)^{2^{61}} \approx 0.36$$

Thus, the probability of success is $1 - 0.36 = 0.64$. This shows that with a probability of more than 0.5, the attacker can find two equal internal states that produce the same keystream, which means the attacker can distinguish between the random sequences and keystream sequences (similar to the attack in [1]). The data and memory complexity of the attack are $2^{40}\ .\ 2^{22} = 2^{62}$ and 2^{39}, respectively. The attack shows that the security of the suggested parameters in [2] is not as promised (i.e. $2^{KL} = 2^{80}$). Hamann and Krause did not consider TMDTO distinguishing attacks carefully enough.

Some may think that if all IV bits are used continuously in keystream production (i.e., $CIVL = 80$ bits while $IVL = 80$ bits), then the proposed attack is not applicable. This logic is right (because, in this situation, different IVs produce different CIV and the state transition function is bijective), but another type of attack is still applicable. Suppose an attacker has access to the keystreams under an arbitrarily fixed IV and different keys in an SSC with the following parameters.

$$VSL = 100, KL = IVL = 80, PKI = 2^{20}, CIVL = 80, CKL = 0$$

The attacker can apply a TMDTO distinguishing attack similar to the previous attack. The attacker needs all keystreams of 2^{31} keys under an arbitrarily fixed IV (i.e. $2^{31} . 2^{20}$ keystream bits). In this case, the attacker receives 2^{51} keystreams under the same CIV, saves half of the keystreams in a table, and searches for a collision between the remaining keystreams and the keystreams in the table. The probability of failure is:

$$\left(1 - \frac{2^{19} .\ 2^{31}}{2^{100}}\right)^{2^{19} .2^{31}} = \left(1 - \frac{1}{2^{50}}\right)^{2^{50}} = 0.36$$

The probability of success is 0.64. This shows that the attacker can apply a TMDTO distinguishing attack successfully with 2^{51} data complexity.

In addition to these distinguishing attacks, a TMDTO attack for recovering internal state bits of the proposed stream cipher in [2] is surprisingly applicable. The parameters of the corresponding SSC are as follows.

$$VSL = 100 , KL = IVL = 80 , PKI = 2^{20}, CIVL = 60, CKL = 0$$

The goal is to recover VS bits in this attack (CIV bits are known to attackers). In the offline phase of the attack, an attacker produces 2^{40} keystreams under a fixed known CIV (e.g., CIV_1) and random VS bits and saves VS bits and the corresponding keystream bits in a searchable table. The attacker chooses 2^{20} IVs corresponding to CIV_1 in the online phase of the attack, and receives 2^{20} keystreams for every IV. The probability of failure is:

$$\left(1 - \frac{2^{40}}{2^{100}}\right)^{2^{40}} = \left(1 - \frac{1}{2^{60}}\right)^{2^{40}}$$

The attacker can repeat this process 2^{20} times with different CIVs to achieve success. The probability of failure will be:

$$\left(1 - \frac{1}{2^{60}}\right)^{2^{40} .\ 2^{20}} = \left(1 - \frac{1}{2^{60}}\right)^{2^{60}} = 0.36$$

So, the probability of success will be 0.64. This shows that the attacker can apply a TMDTO attack successfully with 2^{60} data complexity.

Although it is possible to obtain the correct security level against TMDTO attacks through Continuous-IV-Use construction (and choosing correct parameters), Continuous-IV-Use construction is unrealistic in many applications. We now turn to discuss general and secure constructions, considering all aspects.

4. Constructions for designing secure SSCs against TMDTO attacks

Selecting a suitable construction for an SSC depends on the application scenario. In some applications, a TMDTO distinguishing attack with high data complexity might be tolerable because it is possible that a cipher never produces enough keystream bits to succeed in distinguishing attacks). In this case, $VSL = CKL = KL$ and $CIVL = 0$ are the best design parameters, and these are used in the Fruit-80 and Fruit-128 SSCs [4,8].

We discuss different application scenarios for full security against TMDTO attacks and propose five constructions for different applications to obtain desirable secure parameters in every construction. These five constructions are the topics of the next five subsections.

In the first case, we propose a construction with the assumption that each IV can be used at most once for encryption. This case seems unrealistic, but when we want to design ultra-lightweight cipher, we should consider all conditions of cryptosystems. This condition has been exploited as mentioned in [9].

"*We would like to point out that for scenarios where different (e.g., session) keys are used, it is important to deprive an attacker of the possibility to collect more data based on a situation where the same IVs are used in different sessions*"[9].

Nevertheless, our third case provides parameters for designing secure SSCs without any limitation on the IVs and keys. We propose two constructions based on the limitation on the number of keystreams per key or IV (i.e., packet mode). For example, the packet length per key (PK) and packet length per key/IV pair (PKI) in the instantiation of the fourth case are 2^{35} and 2^{20}, respectively. In this case, it is possible to use every key ${2^{35}}/{2^{20}} = 32768$ times with different IVs without reducing the security against TMDTO attacks.

Note that in many applications (for example, A5/1 in GSM, E0 in Bluetooth, CCMP in WLAN, SSL/TLS and TLS in computer networks), stream ciphers are employed in packet mode (in contrast with one-stream mode), and they need less than 2^{18} keystream bits for encryption [11]. In another example, the LIZARD stream cipher [11,12], it is forbidden to produce more than 2^18 keystreams per IV. Thus, the limitation on the number of produced keystreams is acceptable in many cases.

4.1. First case: Every IV can be used at most once in the initializations

In this case, it is not possible to initialize the cipher with different keys and the same IV. Suppose that an attacker wants to apply a TMDTO distinguishing attack to a stream cipher in Continuous-Key/IV-Use construction. The construction can produce PIV keystreams for every initialization[1], and there are at most $2^{IVL-CIVL}$ different IVs with the same CIV. For example, if we consider a fixed key (e.g., K_1); the internal state transition can be the same as follows.

$$\begin{array}{lllll}
K_1, IV_1 & \xrightarrow{initialization} VS_1^1, CK_1, CIV_1 & \xrightarrow{Clock} VS_1^2, CK_1, CIV_1 & \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS_1^{PIV}, CK_1, CIV_1 \\
K_1, IV_2 & \xrightarrow{initialization} VS_2^1, CK_1, CIV_1 & \xrightarrow{Clock} VS_2^2, CK_1, CIV_1 & \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS_2^{PIV}, CK_1, CIV_1 \\
\vdots & \qquad\vdots & \qquad\vdots & \qquad\vdots & \qquad\vdots \\
K_1, IV_{2^{IVL-CIVL}} & \xrightarrow{initialization} VS^1_{2^{IVL-CIVL}}, CK_1, CIV_1 & \xrightarrow{Clock} VSL^2_{2^{IVL-CIVL}}, CK_1, CIV_1 & \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS^{PIV}_{2^{IVL-CIVL}}, CK_1, CIV_1
\end{array}$$

Thus, the attacker can access at most $PIV\,.\,2^{IVL-CIVL}$ keystreams under an arbitrary unknown fixed key with different IVs but the same CIV. The attacker saves half of the keystreams (with the same CIV) in a table and searches for a collision between the remaining keystreams and the keystreams in the table. If the attacker finds a collision, he will be able to distinguish keystreams of the cipher from truly random sequences. The probability of failure (i.e., the attacker cannot find any collision) is:

$$\left(1-\frac{PIV\,.\,2^{IVL-CIVL}/2}{2^{vsl}}\right)^{PIV\,.\,2^{IVL-CIVL}/2} = \left(1-\frac{1}{2^{vsl-log_2 PIV-IVL+CIVL+1}}\right)^{2^{log_2 PIV+IVL-CIVL-1}}$$

The attacker can repeat this process 2^x times to achieve success. The probability of failure is:

$$\left(1-\frac{1}{2^{vsl-log_2 PIV-IVL+CIVL+1}}\right)^{2^{log_2 PIV+IVL-CIVL-1+x}}$$

If $2^{vsl-log_2 PIV-IVL+CIVL+1} \leq 2^{log_2 PIV+IVL-CIVL-1+x}$, the attacker can apply this TMDTO distinguishing attack successfully by choosing the maximum possible value of x to succeed in the attack. The x value is limited by the data complexity of the attack. The data complexity of the attack is $2^x\,.\,PIV\,.\,2^{IVL-CIVL}$ (that should be less than 2^{KL}). The maximum possible value of x is $KL-IVL+CIVL-log_2 PIV$ (as 2^x is the number of repeats, x should be at least zero). Thus, VSL should be less than $KL+IVL+log_2 PIV-CIVL-2$ to succeed in the attack.

We conclude that $VSL \geq 2KL+log_2 PIV-CIVL-2$ guarantees the KL-bit security against TMDTO distinguishing attacks[2]. Note that the results of choosing bigger values for $CIVL-log_2 PIV$ are a smaller VSL and a lighter cipher.

[1] In this case PIV is equal to PKI.

[2] As we mentioned previously, the length of keys and IVs should theoretically be equal for providing key length security against TMDTO attacks, thus, it is supposed that the length of keys and IVs are equal throughout the paper.

Now, if we consider TMDTO attacks for recovering internal state bits of a Continuous-Key/IV-Use construction, we obtain $VSL \geq 2KL + log_2 PIV - CIVL - CKL$ for KL-bit security against TMDTO attacks.

Therefore, designers should consider at least VSL bits for optimal parameters when every IV is used at most once in the initialization, where VSL is defined as follows[3].

$$VSL = max\{2KL + log_2 PIV - CIVL - 2,\ 2KL + log_2 PIV - CIVL - CKL\}$$

For instantiation, a suitable secure choice for the parameters would be as follows.

$$KL = IVL = 80, PIV = PKI = 2^{20}, CKL = 0, CIVL = 80, VSL = 100$$

4.2. Second case: Every key can be used at most once in the initializations

This case's condition means that it is not possible to initialize the cipher with one key and different IVs[4]. Suppose that an attacker wants to apply a TMDTO distinguishing attack to a stream cipher in Continuous-Key/IV-Use construction. The construction can produce PK keystreams for every initialization[5], and there are at most 2^{KL-CKL} different keys with the same CK. For example, if we consider a fixed IV (e.g., IV_1); the internal state transition can be as follows.

$$\begin{array}{lllllllll}
K_1, IV_1 & \xrightarrow{initialization} & VS_1^1 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} & VS_1^2 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS_1^{PK}, CK_1 \ ,CIV_1 \\
K_2, IV_1 & \xrightarrow{initialization} & VS_2^1 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} & VS_2^2 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS_2^{PK}, CK_1 \ ,CIV_1 \\
\vdots & & \vdots & & & \vdots & & \vdots & \vdots \\
K_{2^{KL-CKL}}, IV_1 & \xrightarrow{initialization} & VS_{2^{KL-CKL}}^1 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} & VSL_{2^{KL-CKL}}^2 & ,CK_1 & ,CIV_1 \xrightarrow{Clock} \ldots. \xrightarrow{Clock} & VS_{2^{KL-CKL}}^{PK} \ ,CK_1 \ ,CIV_1
\end{array}$$

Thus, the attacker can access at most $PK.2^{kL-CKL}$ keystreams under an arbitrary known fixed IV and different keys (but the same CK). The attacker saves half of the keystreams in a table and searches for a collision between the remaining keystreams and the keystreams in the table. Similar to the last section, it is obtained that VSL should be as follows for the construction to guarantee KL-bit security against TMDTO distinguishing attacks. Note that the results of choosing bigger values for $CKL - log_2 PK$ are a smaller VSL and a lighter cipher.

$$VSL \geq 2KL + log_2 PK - CKL - 2$$

If we consider TMDTO attacks for recovering internal state bits of a Continuous-Key/IV-Use construction, we obtain $VSL \geq 2KL - CKL$ for KL-bit security against TMDTO attacks.

[3] For the construction to resist against TMDTO attacks, VSL should be equal or bigger than VSL in both distinguishing attack and internal state recovery attack.

[4] It is obvious that using one IV twice under one key is forbidden in any stream cipher.

[5] In this case PK is equal to PKI

Therefore, designers should consider at least VSL bits for optimal parameters when every key is used at most once in the initialization, with VSL is defined as follows[6].

$$VSL = max\{2KL + log_2 PK - CKL - 2,\ 2KL - CKL\}$$

For instantiation, a suitable secure choice for the parameters would be as follows.

$$KL = IVL = 80, PK = PKI = 2^{20}, CKL = 80, CIVL = 0, VSL = 98$$

4.3. Third case: Every key or IV can be used many times in the different initializations

In This case, there is no limitation on how many times one key or IV is used for initializations. Suppose that an attacker wants to apply a TMDTO distinguishing attack to a stream cipher in Continuous-Key/IV-Use construction similar to the other cases. The construction can produce at most $PKI\ .\ 2^{IVL-CIVL}\ .\ 2^{kL-CKL}$ keystreams under a fixed CIV and CK. The attacker saves half of the keystreams in a table and searches for a collision between the remaining keystreams and the keystreams in the table. Similar to the last section, it is obtained that VSL should be as follows until the construction guarantees KL-bit security against the TMDTO distinguishing attack. Note that the results of choosing bigger values for $CKL + CIVL - log_2 PKI$ are a smaller VSL and a lighter cipher.

$$VSL \geq 3KL + log_2 PKI - CIVL - CKL - 2$$

If we consider TMDTO attacks for recovering internal state bits of a Continuous-Key/IV-Use construction, we obtain $VSL \geq 2KL - CKL$ for KL-bit security against TMDTO attacks, where VSL is defined as follows.

$$VSL = max\{3KL + log_2 PKI - CIVL - CKL - 2,\ 2KL - CKL\}$$

For instantiation, a suitable secure choice for the parameters would be as follows.

$$KL = IVL = 80, CKL = CIVL = 60, PKI = 2^{20}, VSL = 138$$

4.4. Fourth case: Limitation on the number of keystreams per key

If we suppose that every key can produce at most PK keystream bits, there will be at most $PK\ .\ 2^{KL-CKL}$ internal states with the same CK and CIV. For example, if we consider a fixed key (e.g., K_1); the internal state transition will be as follows[7].

$$\begin{array}{lllll}
K_1, IV_1 \xrightarrow{initialization} & VS_1^1\ , CK_1\ , CIV_1 \xrightarrow{Clock} & VS_1^2\ , CK_1\ , CIV_1 \xrightarrow{Clock} & \dots . \xrightarrow{Clock} & VS_1^{PKI}\ , CK_1\ , CIV_1 \\
K_1, IV_2 \xrightarrow{initialization} & VS_2^1\ , CK_1\ , CIV_1 \xrightarrow{Clock} & VS_2^2\ , CK_1\ , CIV_1 \xrightarrow{Clock} & \dots . \xrightarrow{Clock} & VS_2^{PKI}\ , CK_1\ , CIV_1 \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
K_1, IV_{PK/_{PKI}} \xrightarrow{initialization} & VS_{PK/_{PKI}}^1, CK_1, CIV_1 \xrightarrow{Clock} & VSL_{PK/_{PKI}}^2, CK_1, CIV_1 \xrightarrow{Clock} & \dots . \xrightarrow{Clock} & VS_{PK/_{PKI}}^{PKI}\ , CK_1\ , CIV_1
\end{array}$$

[6] To resist the construction against TMDTO attacks, VSL should be equal to or bigger than VSL in both the distinguishing attack and internal state recovery attack.

[7] It is supposed that $^{PK}/_{PKI}$ is a positive integer for simplicity of calculations.

As there are 2^{KL-CKL} different keys with the same CK, there are at most $PK\ .\ 2^{KL-CKL}$ internal states with CK_1 and CIV_1. An attacker can save half of the keystreams in a table and search for a collision between the remaining keystreams and the keystreams in the table. The probability of failure (i.e., the attacker cannot find any collision) is:

$$\left(1-\frac{PK\ .\ 2^{KL-CKL}/2}{2^{vsl}}\right)^{PK.2^{KL-CKL}/2} = \left(1-\frac{1}{2^{vsl-log_2 PK-KL+CKL+1}}\right)^{2^{log_2 PK+KL-CKL-1}}$$

The attacker can repeat this process 2^x times to achieve success. The probability of failure is:

$$\left(1-\frac{1}{2^{vsl-log_2 PK-KL+CKL+1}}\right)^{2^{log_2 PK+KL-CKL-1+x}}$$

If $2^{vsl-log_2 PK-KL+CKL+1} \le 2^{log_2 PK+KL-CKL-1+x}$, the attacker can apply this TMDTO distinguishing attack successfully by choosing the maximum possible value of x to succeed in the attack. The x value choice is limited by the data complexity of the attack, just as for the first case. The data complexity of the attack is $2^x\ .\ PK\ .\ 2^{KL-CKL}$ (that should be less than 2^{KL}). The maximum possible value of x is $CK - log_2 PK$ (as 2^x is the number of repeats, x should be at least zero). Thus, VSL should be less than $2KL + log_2 PK - CKL - 2$ to succeed in the attack.

We conclude that $VSL \ge 2KL + log_2 PK - CKL - 2$ guarantees the KL -bit security against TMDTO distinguishing attacks. Note that the results of choosing bigger values for $CKL - log_2 PK$ are a smaller VSL and a lighter cipher.

Now, let us consider a TMDTO attack for recovering the internal state. We obtain $VSL \ge 2KL - CKL$ for KL-bit security against TMDTO attacks. This shows that designers should use at least VSL bits for optimal parameters, with VSL as follows.

$$VSL = max\{2KL + log_2 PK - CKL - 2,\ 2KL - CKL\}$$

For instantiation, a suitable secure choice for the parameters would be as follows.

$$KL = IVL = 80, PK = 2^{35}, CKL = 80, CIVL = 0, PKI = 2^{20}, VSL = 113$$

If $CIVL = 0$ and $CKL = KL$ are selected, then $VSL = KL + log_2 PK - 2$ and the parameters introduce an SSC with KL-bit security against all types of TMDTO attacks.

4.5. Fifth case: Limitation on the number of keystreams per IV

If we suppose that every IV can produce at most PIV keystream bits, there will be at most $PIV\ .\ 2^{IVL-CIVL}$ internal states with the same CK and CIV[8]. It is simple to show that designers should consider at least VSL bits for optimal parameters, with VSL as follows.

$$VSL = max\{2KL + log_2 PIV - CIVL - 2,\ 2KL + log_2 PIV - CIVL - CKL\}$$

For instantiation, a suitable secure choice for the parameters would be as follows.

$$KL = IVL = 80, PIV = 2^{35}, CKL = 0, CIVL = 80, PKI = 2^{20}, VSL = 115$$

[8] Similar to the fourth case, it is supposed that ${}^{PIV}/_{PKI}$ is a positive integer.

This construction requires saving IV bits after initialization and, as discussed previously, storing IV bits contradicts the design philosophy of SSCs in many applications.

5. Conclusion

Small-state stream ciphers (SSC) are considered because of their small area in hardware, low power consumption, and low cost. SSCs such as Sprout, Fruit-80, and Plantlet have been designed, but Hamann et al. [1] proposed a distinguishing TMDTO attack against all SSCs. They proposed to save and use IV bits (as well as key bits) continuously to strengthen SSCs against TMDTO attacks. Then, Hamann and Krause [2] claimed that only continuously using IV bits in packet mode is sufficient to strengthen SSCs against TMDTO attacks, and they proposed parameters of an SSC resistant to TMDTO attacks.

The analysis of the current paper shows that the corresponding cipher (using the proposed parameters) is vulnerable to TMDTO attacks and storing IV bits in many applications imposes an unacceptable overhead on cryptosystems.

This paper proposed five different constructions based on different applications that can provide full security against TMDTO attacks. With these constructions, the design of a new generation of stream ciphers (i.e., ultra-lightweight) is achievable with full security against TMDTO attacks. We present instantiations of parameters from the proposed constructions as follows.

	CKL	$CIVL$	PK	PIV	PKI	VSL
First case: Every IV can be used at most once in the initializations	0	80	$\geq 2^{20}$	2^{20}	2^{20}	100
Second case: Every key can be used at most once in the initializations	80	0	2^{20}	$\geq 2^{20}$	2^{20}	98
Third case: Every key/IV can be used many times in the different initializations	60	60	$\geq 2^{40}$	$\geq 2^{40}$	2^{20}	138
Fourth case: Limitation on the number of keystreams per key	80	0	2^{35}	$\geq 2^{20}$	2^{20}	113
Fifth case: Limitation on the number of keystreams per IV	0	80	$\geq 2^{20}$	2^{35}	2^{20}	115

Volatile state length, VSL, of instantiations for different cases under $KL = IVL = 80$

Acknowledgement

This work has been supported by CAS President's International Fellowship Initiative, PIFI, program.

References

1. Hamann, M., Krause, M., Meier, W., Zhang, B.: Design and analysis of small-state grain-like stream ciphers. Cryptography and Communications 10(5), 803-834 (2017).

2. Hamann, M., Krause, M.: Tight Security Bounds for Generic Stream Cipher Constructions. Cryptology ePrint Archive 2019, 007 (2019). https://eprint.iacr.org/2019/007
3. Armknecht, F., Mikhalev, V.: On lightweight stream ciphers with shorter internal states. In: Leander, G. (ed.) Fast Software Encryption: 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers, pp. 451-470. Springer, Berlin (2015).
4. Amin Ghafari, V., Hu, H.: Fruit-80: A Secure Ultra-Lightweight Stream Cipher for Constrained Environments. Entropy 20(3), 180 (2018). https://www.mdpi.com/1099-4300/20/3/180
5. Ghafari, V.A., Hu, H., Chen, Y.: Fruit-v2: ultra-lightweight stream cipher with shorter internal state. Cryptology ePrint Archive 2016, 355 (2016). https://eprint.iacr.org/2016/355
6. Biryukov, A., Shamir, A.: Cryptanalytic time/memory/data tradeoffs for stream ciphers. In: In: Okamoto, T. (ed.) Advances in Cryptology — ASIACRYPT 2000: 6th International Conference on the Theory and Application of Cryptology and Information Security Kyoto, Japan, December 3–7, 2000 Proceedings, pp. 1-13. Springer, Berlin (2000).
7. Mikhalev, V., Armknecht, F., Müller, C.: On ciphers that continuously access the non-volatile key. IACR Transactions on Symmetric Cryptology 2016(2), 52-79 (2017). https://tosc.iacr.org/index.php/ToSC/article/view/565
8. Ghafari, V.A., Hu, H., Alizadeh, M.: Necessary conditions for designing secure stream ciphers with the minimal internal states. Cryptology ePrint Archive 2017, 765 (2017). https://eprint.iacr.org/2017/765
9. Hamann, M., Krause, M., Meier, W.: A Note on Stream Ciphers that Continuously Use the IV. Cryptology ePrint Archive 2017, 1172 (2017). https://eprint.iacr.org/2017/1172
10. Hong, J., Sarkar, P.: New applications of time memory data tradeoffs. In: International Conference on the Theory and Application of Cryptology and Information Security 2005, pp. 353-372. Springer, Berlin (2005).
11. Hamann, M., Krause, M.: On stream ciphers with provable beyond-the-birthday-bound security against time-memory-data tradeoff attacks. Cryptography and Communications 10(5), 959-1012 (2018).
12. Hamann, M., Krause, M., Meier, W.: LIZARD–a lightweight stream cipher for power-constrained devices. IACR Transactions on Symmetric Cryptology 2017(1), 45-79 (2017). https://tosc.iacr.org/index.php/ToSC/article/view/584