[ Robert Hiromoto, Albert Carlson, Mandeep Singh. (Dec 05, 2024). Breaking the Counter (CTR) Mode. Quantum Security Alliance. ]

# Breaking the Counter (CTR) Mode

**Robert E. Hiromoto***, **Albert Carlson**†, **Mandeep Singh** ‡.
*The University of Idaho, Moscow, ID, USA
†Chair for Entropy and Encryption, Quantum Security Alliance
‡Cryptography Researcher, Xsette Technology
*Email: *hiromoto@cs.uidaho.edu †LtZap1@gmail.com, ‡msingh9001@gmail.com*

***Abstract*—A block cipher mode is a cryptographic algorithm that employs a symmetric key block cipher algorithm to defend against language and frequency-based attacks. The goal of modes is to randomize the cipher so that for any two identical input blocks the cipher text output is likely to be different. This apparent morphing of the mapping between input and output makes the plain text block appear to be mapped into multiple cipher text blocks, thus confounding cryptologic analysis. Unfortunately, the structure of cryptographic modes introduces opportunities for side-channel attacks that can reveal enough information to uncover the original message without the need to first break the cipher. Of the six generally accepted cryptographic modes (ECB, CBC, PCBC, CTR, OFB, and CFB), the first three modes (ECB, CBC, PCBC) have been shown to be vulnerable to side-channel attacks through the detection of cipher collisions. In this paper, an attack on the CTR mode is presented and demonstrated to work in the example presented.**

**Keywords: Modes, Counter Modes, Side-channel attack, Set Theoretic Estimation, Keyless Decryption, 1:1 Principle**

## I. Introduction

Encryption algorithms are methods that obscure and protect the information content either in storage or transit. The ideal cryptographic algorithm is one whose algorithm strength requires the so-called Brute Force Attack, which applies every possible mapping for the key to recover the original unencrypted content of the information. Shannon noted that the most mathematically perfect cipher is the “one time pad" (OTP) [1], [2] in which every character is encrypted with a different cipher/key pair using a perfectly random choice of the ciphers and keys. Unfortunately, this method is proven to be so costly as to be impractical [3].

In response to maximizing the cryptographic algorithm strength, cryptographers developed the concept of randomizing functions called cryptographic modes [4]. These modes are designed to provide sufficient randomized encrypting mutations to the plain-text (PT) data. As inputs to the encryption function, these cryptographic modes typically apply some form of an offset and XOR operations to various combinations of the previous or current PT block with the previous cyber text (CT) block to form the current PT block. There are six major designated cryptographic modes. They are Electronic Code Book (ECB), Cipher Block Chaining (CBC), Propagating CBC (PCBC), Counter (CTR), Output Feedback (OFB), and Cipher Feedback (CFB) modes [5], [6]. The role of the randomization function, which includes XOR operations, is often viewed as mapping a particular PT to multiple CTs. This view is incorrect. Encryption is based on the 1:1 assumption - the principle that a particular plain text can only map to a single cipher text variable and vice versa. The 1:1 assumption ensures that an encrypted PT is reversable during the decryption process, thus preserving the 1:1 mapping.

Recent analysis demonstrated that the cryptographic modes ECB, CBC, and PCBC can be broken. In one study, ECB is broken using standard attacks on ciphers, while CBC and PCBC, with the same general architecture [7], are shown to be vulnerable to side-channel attacks [8], [9], [10] through the detection of cipher collisions. Although the remaining modes (CTR, OFB, and CFB) have different architectures, similar attacks may be possible. In particular, the Counter (CTR) mode has an architecture [11] that lends itself to side-channel attacks, detection of cipher collisions, and the analysis due to the Birthday Paradox arising from the finite set of the PT alphabet.

### A. Paper Organization

The remainder of this paper is organized into five sections. Section two describes the Counter (CTR) mode algorithm. Section three outlines the relationship between previous and current PT block in the encryption process. Section four explains the attacks employed to break the CT encryption. Section five demonstrates the proposed attack. Section six provides a summary and conclusion regarding the significance of the attack method.

## II. Background

Before addressing the attack methodology, an introduction to the constituent technologies is important. The underlying mathematics and approach to the solution are based on set theory and the constraints of the functions used in the mode, itself. Additionally, an introduction to modes is also beneficial.

### A. Modes

Modes are a randomization function that are added to encryption in order to make it appear that a particular plain text input from a message maps to different cipher text. However, encryption is based on the 1:1 assumption - the principle that a particular plain text can only map to a single cipher text variable and vice versa. As can be shown for each mode, the randomization actually changes the input to the encryption function and is reversed during the decryption process. The plain text input is changed to another input and then encrypted, preserving the 1:1 mapping.

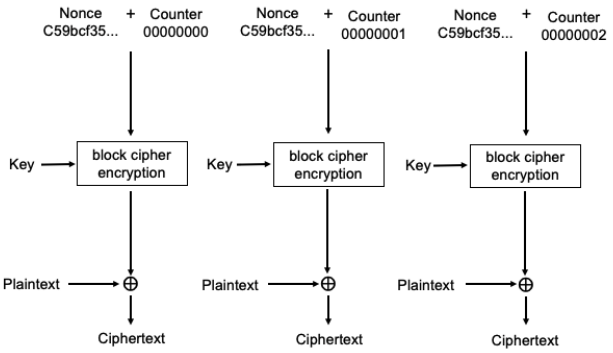


Fig. 1. CTR Mode Architecture [12]

## III. The CTR Mode

The CTR mode has an architecture as shown in Figure 1. Each encryption block depends solely on a randomizing value, known as a "nonce," to which a counter is added. Since each block consists of $b$ bits, the counter must run from $0 \leq ctr \leq 2^{|b|}$. For each new block the counter is incremented so that the result is $nonce + ctr$. When the counter reaches $2^{|b|}$ the next value in the sequence is $2^{|b|} + 1$, which overflows the register that holds the counter value. The effect is that $ctr$ appears to have a 0 value as the additional high order bit is dropped. Therefore, the input to the encryption function cycles through the same incremented values every $\lambda = 2^{|b|}$ blocks. Following the 1:1 principle - the principle that a particular plain text can only map to a single cipher text variable and vice versa - the output of the encryption block will also cycle in the same $\lambda$ blocks. For the CTR mode, the cipher text output ($CT_i$) is given by

$$CT_i = E_k(ctr) \oplus PT_i \tag{1}$$

$$E_k(ctr) = CT_i \oplus PT_i \tag{2}$$

## IV. CTR Attack Mode

In this section, two attack modes are presented. In Attack Mode I, the attack relies on the occurrence of the overflow of the $nonce + ctr$ buffer. In the second attack mode, the counter is observed as it is incremented over each block.

### A. Attack Mode I

Consider the sequence

$$CT_{j(0)} \oplus PT_{j(0)}, CT_{j(1)} \oplus PT_{j(1)}, CT_{j(2)} \oplus PT_{j(2)}, \cdots, CT_{j(m)} \oplus PT_{j(m)} \tag{3}$$

for which the counter at location $j(0) = i$ repeats at locations $j(m) = i + m \cdot |2^n|$, where m counts the number of blocks where the location in $i$ repeats. The significance of the sequence (1) is that it represents the collision of blocks using the same encryption key $E_k(ctr)$ and allows for the possibility of a side-channel attack. The side-channel attack progresses in the following way:

First, it is assumed that the ciphertext $CT_{j(0)}$ at location $i$ is available to the hacker and can be used to compute possible keys to the CTR problem, which satisfies Eqn. 2.

Second, since the same counter key, $E_k$, is reused ever period j, the above sequence must obey the equality

$$\begin{aligned} E_k &= CT_{j(0)} \oplus PT_{j(0)} = CT_{j(1)} \oplus PT_{j(1)} \\ &= CT_{j(2)} \oplus PT_{j(2)} = \cdots = CT_{j(m)} \oplus PT_{j(m)} \end{aligned} \tag{4}$$

for all $j(0) = i$ periods of $j$ with $i$ fixed. If the plain text $PT_i$ is known, the problem of determining the counter key is trivial. However, In general, the CTR key is not known, but Eqn. 4 can be used to generate a set $\{E_k\}$ that can be analyzed using the Set Estimation Technique (STE) developed by Carlson [13]. One possible set, $\{E_k\}$, is denoted by:

$$\{E_k(\mu)_{j(0)=i}\} = CT_{j(0)} \oplus \{\mu\} \tag{5}$$

Because Eqn. (3) represents collisions of the XORed terms with the same $E_k$, the desired $E_k$ represents the only key for which all associated plaintext $PT_j$ are characters allowed in A. It may happen that a collision between $CT_i$ and $CT_j$ may occur, such that,

$$CT_i = CT_j \Rightarrow CT_i \oplus CT_j = 0 \tag{6}$$

and $PT_i = PT_j$. In such a case, no new information is gained, but it does not eliminate the encryption key under evaluation.

Summarizing the Attack Mode I:

1) Observe the $CT_j$ at each periodic position is located at position $j(0) = i, j(m) = i + m \cdot |2^n|$ where $m = 0, 1, 2, \cdots$
2) Form the set of all possible encryption keys $E_k(\mu)_{j(0)}$ for each character in the alphabet A, where $\mu \in A$, and using the $j(0)$ ciphertext $CT_{j(0)}$
$$\{E_k(\mu)_{j(0)}\} = CT_{j(0)} \oplus \{\mu\} \tag{7}$$
for $\forall \mu \in A$.
3) Then for each character $\mu \in A$, use the corresponding encryption key to validate if the associated plaintext $PT_{j(m)}$ is or is not an allowed character in A.
4) The telescoping analysis applied to Eqn. 4 is given by
$$\begin{aligned} E_k(\mu)_{j(0)} &= CT_{j(0)} \oplus \mu \\ PT_{j(1)} &= CT_{j(1)} \oplus E_k(\mu)_{j(0)} \\ PT_{j(2)} &= CT_{j(2)} \oplus E_k(\mu)_{j(0)} \end{aligned}$$
etc.
5) In step 4, the first occurrence of $PT_{j(m)} \notin A$, eliminates the corresponding key $E_k(\mu)_{j(0)}$ from further consideration as a possible encryption key. Otherwise, $m$ is incremented by 1 and the next evaluate $PT_{j(m+1)}$ is evaluated.
6) At the end of this telescoping procedure, the only key in step 4 that correctly identifies all $PT_{j(m)} \in A$ is the encryption key.

The flowchart for the Attack Mode I algorithm is shown in Figure 2.

A second method assembles the set of possible outputs for the same $CTR$ values. Using the 1:1 principle, the same value

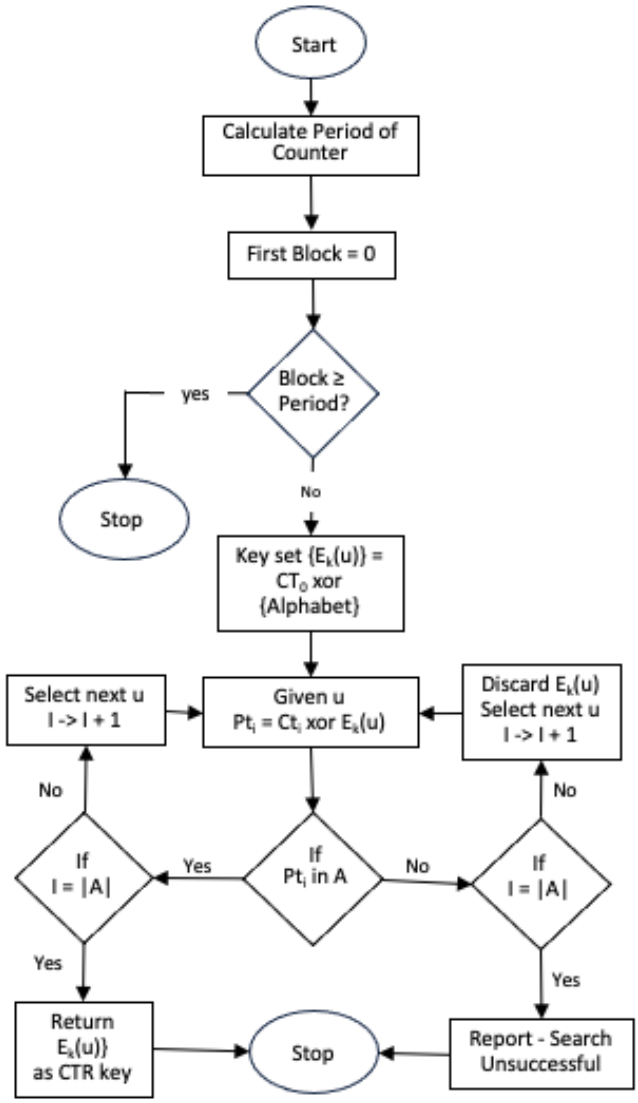


Fig. 2. Method 1: Input Reduction

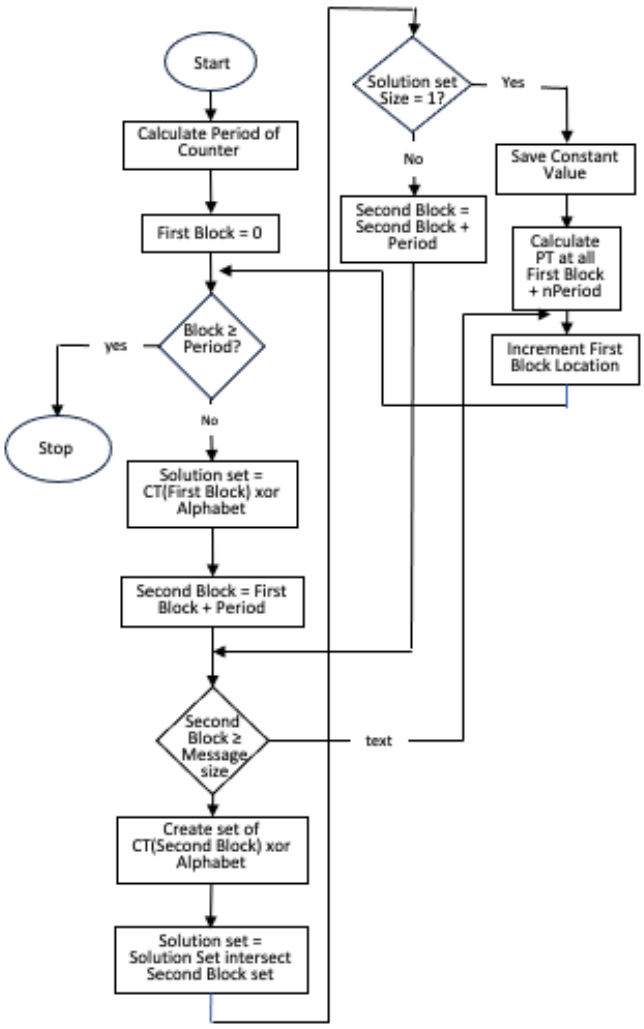


Fig. 3. Method 2: Input Reduction

for $E_k(CTR)$ must occur at each location $i + n|CTR|$. This value is given by

$$E_k(CTR) = CT \oplus PT \tag{8}$$

where all possible values for $PT$ consist of all of the members of the set $\{A\}$ of the encrypted language.

Upon each application to the solution set of keys, the possible set of values for the encrypted $CTR$ should be reduced until only one value is left. Once identified, that value can be used to return the plaintext value for each location for $i + n|CTR|$ in the message, where $n \in \mathbb{I}$ defining the cycle number of the counter in the message. That same value can be eliminated from consideration for each of the other positions in the message, since each encryption mapping must be unique. The flowchart for the algorithm is shown in Figure 3.

Therefore, when considering values where $CTR$ collides, the key $E_k(CTR)$ corresponding to location $i$ is evaluated by intersecting the result for each location $j$ located at the next location $|CTR|$ away. The reduction of spurious keys is accomplished through Eqn 9.

$$E_k(CTR)|_i \in (CT_i \oplus \{A\}) \oplus \bigcap_{j=2}^{n} (CT_{j|CTR|} \oplus \{A\}) \tag{9}$$

### *B. Attack Mode II*

A second side-channel attack relies on the behavior of the counter key. Rather than relying on the periodicity of the counter, the key $(nonce + counter)$ is incremented by one for each subsequent encrypted text. Attack Mode II has the advantage of breaking the key using fewer blocks of encrypted text. On the first pass a fixed location $i$ is assumed for which

$$\{E_k(\mu)_{j(0)}\} = CT_{j(0)} \oplus \{\mu\} \tag{10}$$

for $\forall \mu \in A$ On the second pass

$$\{\mu\} = CT_{j(1)} \oplus \{E_k(\mu)_{j(0)} + 1\} \tag{11}$$

| $j(r,s)$ | $E_k(0) \oplus \{\mu\}$ | $CT$ | ASCII |
|---|---|---|---|
| $j(0,0)$ | $E_k(0) \oplus a$ | U | 0101 0101 |
| $j(1.0)$ | $E_k(0) \oplus l$ | X | 0101 1000 |
| $j(2,0)$ | $E_k(0) \oplus b$ | V | 0101 0110 |
| $j(0,1)$ | $E_k(0) \oplus e$ | Q | 0101 0001 |
| $j(1,1)$ | $E_k(0) \oplus r$ | F | 0100 0110 |
| $j(2,1)$ | $E_k(0) \oplus o$ | [ | 0101 1011 |
| $j(0,2)$ | $E_k(0) \oplus g$ | S | 0101 0011 |
| $j(1,2)$ | $E_k(0) \oplus w$ | C | 0100 0011 |
| $j(2,2)$ | $E_k(0) \oplus n$ | Z | 0101 1010 |
| $j(0,3)$ | $E_k(0) \oplus m$ | Y | 0101 1001 |
| $j(1,3)$ | $E_k(0) \oplus z$ | N | 0100 1110 |
| $j(2,3)$ | $E_k(0) \oplus c$ | W | 0101 0111 |

TABLE I
CIPHERTEXT FOR TEXT KEY $E_k(0)$

and on subsequent passes

$$\{\mu\} = CT_{j(q)} \oplus \{E_k(\mu)_{j(0)} + q\} \tag{12}$$

where $1 \leq q \leq$ BufferOverFlow.

Finally, the one key $E_k$ whose corresponding plaintext are all contained in $A$, is the actual encryption key.

### C. Example - Demonstrating Attack Mode I

For purposes of illustration, assume a CTR mode is added to an encryption on a single byte block. That is, $|CTR| =$ 8-bits. Therefore, the CTR will cycle every $2^8 = 256$ blocks/characters. Further, assume that the language of the message is English, with only lower case character letters (so $|A| = 26$), and the message uses ASCII encoding. Therefore, the PT values range from 0110 0001 to 0111 1010 depicted in the first column of Table II.

Next, assume that the encryption key for this example is given by $E_k(0) = 0011\,0100$. Based on this key, the location within the various cipher blocks can be defined by

$$j(i,m) = i + m \cdot |CTR| = i + m \cdot 2^8 = i + m \cdot 256$$

It is assumed that the attacker has access to all ciphertext in sequential order as well as the ability to detect the buffer overflow of the counter. Therefore, a hacker that observes three consecutive CT values within the first four blocks of $j(i,m)$ would have knowledge of the CTs as displayed in Table I. From this set of observed CTs, the hacker can now compute a set of possible CTR keys from which the correct $E_k(0)$ key can be inferred.

As a side note, Table I illustrates the derivation of the observed CTs as it relates to the encryption key $E_k(0)$ chosen for this example.

Computing $\{E_k(\mu)\}_{j(0)} = CT_{j(0)} \oplus \{\mu\} = U \oplus \{\mu\}$

| $\{\mu\}$ | $\{E_k(\mu)\} = U \oplus \{\mu\}$ |
|---|---|
| (a) 0111 0000 | 0010 0101 |
| (b) 0111 0001 | 0010 0100 |
| (c) 0111 0010 | 0010 0111 |
| (d) 0111 0011 | 0010 0110 |
| (e) 0111 0100 | 0010 0001 |
| (f) 0111 0101 | 0010 0000 |
| (g) 0111 0110 | 0010 0011 |
| (h) 0111 0111 | 0010 0010 |
| (i) 0111 1000 | 0010 1101 |
| (j) 0111 1001 | 0010 1100 |
| (k) 0111 1010 | 0010 1111 |
| (l) 0110 0001 | 0011 0100 |
| (m) 0110 0010 | 0011 0111 |
| (n) 0110 0011. | 0011 0110 |
| (o) 0110 0100 | 0011 0001 |
| (p) 0110 0101 | 0011 0000 |
| (q) 0110 0110 | 0011 0011 |
| (r) 0110 0111 | 0011 0010 |
| (s) 0110 1000 | 0011 1101 |
| (t) 0110 1001 | 0011 1100 |
| (u) 0110 1010 | 0011 1111 |
| (v) 0110 1011 | 0011 0110 |
| (w) 0110 1100 | 0011 1001 |
| (x) 0110 1101 | 0011 1000 |
| (y) 0110 1110 | 0011 1011 |
| (z) 0110 1111 | 0011 1010 |

TABLE II
ASCII ENCODED PT AND CT CHARACTERS

In this example, the set $\{E_k(\mu)\}$, in Table II, are the possible keys for the example CTR problem. These keys are derived from the first ciphertext, in Table I, as observed by the hacker. Since the side-channel attack relies on the repeated use of the same key $E_k$ for each position, one and only one $E_k$ key in the set $\{E_k(\mu)\}$ can satisfy this requirement. Therefore, there are no restrictions on the derivation of the initial set of possible keys, which in this case is based on the ciphertext U.

The last column of Table III shows how many possible mappings for the encrypted key are eliminated as calculating the sets of possible keys for each CT block. The first block considered and the initial set of possible There are no other sets assembled for comparison and therefore, no keys from that first set can be eliminated.

Among all the possible keys found in Table II, the isolation of the correct CTR key is demonstrated using the telescoping procedure defined by Eqn. (6). As an example, consider the key $E_k(c)$ as it is applied to all remaining CT in the observed sequence.

$$\begin{aligned} PT_{j(1)} &= CT_{j(1)} \oplus E_k(c) \\ &= X \oplus 0010\,0111 \\ &= 0101\,1000 \oplus 0011\,0110 \\ &= 0110\,1110 \\ PT_{j(1)} &= y. \end{aligned}$$

Since $y \in A$, the key $E_k(c)_{j(0)}$ remains as a possible solution key. This procedure continues to the next $j(2)$ CT in the

sequence with

$$\begin{aligned} PT_{j(2)} &= CT_{j(2)} \oplus E_k(c)_{j(0)} \\ &= V \oplus 0011\,0110 \\ &= 0101\,0110 \oplus 0011\,0110 \\ &= 0110\,0000 \end{aligned}$$

Unfortunately, $PT_{j(2)} \notin A$ and therefore, the $E_k(c)_{j(0)}$ key is eliminated from the set of possible keys.

As a second example, consider the key associated with the CT Y and the PT m.

$$E_k(m) = Y \oplus m = 0011\,0110,$$

then for the next $j(1)$ cycle

$$\begin{aligned} PT_{j(1)} &= CT_{j(1)} \oplus E_k(m) \\ &= X \oplus 0111\,1010 \\ &= 0101\,1000 \oplus 0011\,0110 \\ &= 0110\,1110 \\ PT_{j(1)} &= o. \end{aligned}$$

$$\begin{aligned} PT_{j(2)} &= CT_{j(2)} \oplus E_k(m) \\ &= V \oplus 0011\,0011 \\ &= 0101\,0110 \oplus 0011\,0011 \\ &= 0110\,0001 \\ PT_{j(2)} &= l. \end{aligned}$$

$$\begin{aligned} PT_{j(3)} &= CT_{j(3)} \oplus E_k(m) \\ &= Q \oplus 0011\,0011 \\ &= 0101\,0001 \oplus 0011\,0011 \\ &= 0110\,0010 \\ PT_{j(3)} &= m. \end{aligned}$$

$$\begin{aligned} PT_{j(4)} &= CT_{j(4)} \oplus E_k(m) \\ &= F \oplus 0011\,0011 \\ &= 0100\,0110 \oplus 0011\,0011 \\ &= 0111\,0101 \\ PT_{j(4)} &= g. \end{aligned}$$

$$\begin{aligned} PT_{j(5)} &= CT_{j(5)} \oplus E_k(m) \\ &= [ \oplus 0011\,0011 \\ &= 0101\,1011 \oplus 0011\,0011 \\ &= 0111\,0101 \\ PT_{j(5)} &= s. \end{aligned}$$

$$\begin{aligned} PT_{j(6)} &= CT_{j(6)} \oplus E_k(m) \\ &= S \oplus 0011\,0011 \\ &= 0101\,0011 \oplus 0011\,0011 \\ &= 0110\,0000 \\ PT_{j(6)} &= \notin A.. \end{aligned}$$

| PT | CT | Eliminated |
|---|---|---|
| (a) 0101 0101 | U | - |
| (l) 0101 1000 | X | 7 |
| (b) 0101 0110 | V | 3 |
| (e) 0101 0001 | Q | 3 |
| (r) 0100 0110 | F | 4 |
| (o) 0101 1011 | [ | 0 |
| (g) 0101 0011 | S | 1 |
| (w) 0100 0011 | C | 2 |
| (n) 0101 1010 | Z | 0 |
| (m) 0101 1001 | Y | 1 |
| (z) 0100 1110 | N | 2 |
| (c) 0101 0111 | W | 1 |

TABLE III
NUMER OF STEPS TO ELIMINATION

Therefore, after six steps, the key $E_k(m)$ is also eliminated from the set of possible keys to the example CTR problem. It should be clear that the key $E_k(l)$ computed in Table II will satisfy $PT = CT \oplus E_k(l)$ for $\forall PT \in A$.

For the second method using a set-reduction scheme, Eqn. 9, the results for the number of elimination steps are displayed in Figure III. Notice that the number of elimination steps differ from the evaluations presented above for the telescoping method. The important point to understand is that the set-reduction method created and reduced different collections of key sets that still led to the identification of the correct key for the test problem.

### *D. Example - Outline of Attack Mode II*

A second side-channel attack relies on the behavior of the counter key. Rather than relying on the periodicity of the counter, the key (nonce + counter) is incremented by one for each subsequent encrypted text. Attack Mode II has the advantage of breaking the key using fewer blocks of encrypted text. On the first pass a fixed location i is assumed for which

$$\{E_k(\mu)_{j(0)}\} = CT_{j(0)} \oplus \{\mu\}$$

for $\forall \mu \in A$. On the second pass

$$\{\mu\} = CT_{j(1)} \oplus \{E_k(\mu)_{j(0)} + 1\},$$

and on subsequent passes

$$\{\mu\} = CT_{j(q)} \oplus \{E_k(\mu)_{j(0)} + q\},$$

where $1 \leq q \leq$ BufferOverFlow. Finally, the one key $E_k$ whose corresponding plaintext is contained the $A$, is the actual encrypted key.

This example took 11 periods of the counter, but settled on the correct key value. While 11 periods may appear long, most applications of the technique result in reducing the solutions set. For each value of $E_k(CTR)$ eliminated more is known about the randomization function and its final value. The methodology is an excellent representation of using STE in a side-channel attack to arrive at an answer.

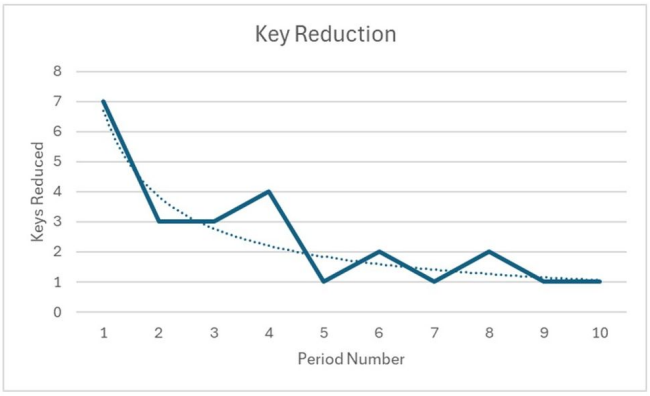


Fig. 4. Input Reduction for Example

## V. Math Analysis

Language is not random, but it is often treated as being random. Even in random environments, there are collisions. There are periodic and predictable collisions in the input to the randomizing encryption function, but collisions of cipher text do not play a significant role in attacking the CTR mode. However, knowing that one of the inputs to the CT output is a constant constitutes a vulnerability in the mode architecture. The periodicity is determined by the size of the block feeding the encryption block. As with all ciphers, Shannon showed that there must be a sufficient number of input symbols (the unicity distance) to allow for unambiguous decryption [1]. This number may be relatively large and as long as the message is shorter than the unicity distance, the message is likely to be secure. The minimum number of characters (theoretical unicity distance) can be estimated as used as the lower limit for message size. Assuming randomness, the minimum number of periods is

$$n_{min} \approx lg_2|A| \tag{13}$$

the log of the size of the alphabet of the message. Practically, the number of periods is increased if the character found at the position of $i\lambda$ in the message collides with any of the characters for $1 \leq i < n$ for any of the $n$ periods processed. In practice, when there was no collision of data, the reduction of possible values for the constant input follows a predictable drop (see Figure 4). Whether this drop is completed depends on the size of the input message. However, the reduction in possible values for the constant input $E_k(CTR)$. Each reduction makes the encrypted message less secure [14].

The example used to show the reduction of PT values until the correct constant value is identified was applied to a single byte. This is not a limitation of the algorithm. This algorithm is extensible for any size block used as an input to the mode. There is no limitation as to the size of the block or input symbols and constitutes a general solution.

## VI. Conclusions and Future Work

CTR mode is one of the family of modes that are generally accepted as being effective for adding randomness to encryption routines. Previous work has demonstrated that CBC and PCBC modes are easily broken by side-channel attacks.

Since the actual counter value and nonce also do not appear in the mathematics used to break the encryption, knowledge of the two values is not needed and does not constitute a secret value requiring security during sharing. In practice, neither value needs to be shared, rendering them irrelevant to security.

The majort conclusion is that the CTR mode is broken via a side-channel attack and is not effective for security. This is the fourth major mode that has been shown to be vulnerable to side-channel attacks. With each successive mode break, it appears that modes using the traditional mode architectures are not safe and should be abandoned. Bolt on modes appear to carry enough information to constitute cyberfragility and actually make encryption less secure. Therefore, modes such as CTR should be avoided. Any randomization must be integral to the encryption algorithm in order to be useful.

There are two major modes that remain to be shown as vulnerable to side-channel attacks. These are the OFB and CFB modes. Both have similar architectures that must be analyzed for susceptibility to attack. Investigating these modes is the next step in verifying and characterizing the approach to randomization through incrementally adding mode functions.

## References


[1] Claude Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28:656 – 715, 1949.

[2] Uli Maurer. A universal test for random bit generators. *Journal of Cryptography*, 5(2):89–105, 1992.

[3] John Earl Haynes and Harvey Klehr. *Venona: Decoding Soviet Espionage in the United States (Yale Nota Bene)*. Yale University Press, 1999.

[4] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley and Sons Inc., New York, 2nd edition, 1996.

[5] ISO/IEC. Iso/iec 10116:2017 – information technology – security techniques – modes of operation for an n-bit block cipher, https://www.iso.org/obp/ui/en/iso:std:iso-iec:10116:ed-4:v1:en. Technical report, ISO/IEC, 2017.

[6] von Oorschot Paul Menezes, Alfred. and Scott Vanstone. *Handbook of Applied Cryptography*. CRC Press, New York, 1996.

[7] NIST. Fips 81, des modes of operation, https://csrc.nist.gov/files/pubs/fips/81/final/docs/fips81.pdf. Technical report, NIST, 1980.

[8] David McGrew. Impossible plaintext cryptanalysis and probable-plaintext collision attacks of 64-bit block cipher modes. In *Proceedings of the Fast Software Encryption Workshop*, 2013.

[9] Albert Carlson, Bhaskar Ghosh, and India K. Dutta. Using the collision attack for breaking cryptographic modes. *13th International Congress on Computing, Communication, and Networking Technologies*, Oct. 3 - 5, 2022.

[10] Bhaskar Ghosh, Albert Carlson, and Indira Dutta. A demonstrable break of pcbc mode. *International Symposium on Networks, Computers and Communications (ISNCC): Trust, Security and Privacy (ISNCC 2023), Dohar, Qatar*.

[11] Stack Exchange. stream cipher - does an iv need to be used in aes ctr mode? - cryptography stack exchange, https://crypto.stackexchange.com/questions/62029/does-an-iv-need-to-be-used-in-aes-ctr-mode, 2024.

[12] Wikipedia. Block cipher mode of operation - wikipedia, https://en.wikipedia.org/wiki/block_cipher_mode_of_operation.

[13] Albert Carlson. *Set Theoretic Estimation Applied to the Information Content of Ciphers and Decryption*. PhD thesis, University of Idaho, 2012.

[14] Albert Carlson, Torsten Gang, Garrett Gang, Bhaskar Ghosh, and Indira Dutta. Evaluating true cryptographic key spacesize. *Ubiquitous Computing, Electronics, & Mobile Communication Conference (UEMCON 2021)*, 1 - 4 December 2021.