A Guess-and-Determine Attack on Reduced-Round Khudra and Weak Keys of Full Cipher

Mehmet Özen[1], Mustafa Çoban[1,2], Ferhat Karakoç[2]

[1] Sakarya University, Faculty of Arts and Sciences, Department of Mathematics, Sakarya, Turkey
ozen@sakarya.edu.tr
[2] TÜBİTAK - BİLGEM - UEKAE, PK 74, 41470, Gebze, Kocaeli, Turkey
{mustafa.coban,ferhat.karakoc}@tubitak.gov.tr

Abstract. Khudra is a lightweight block cipher designed for Field Programmable Gate Array (FPGA) based platforms. The cipher has an 18-round generalized type-2 Feistel structure with 64-bit block size. The key schedule takes 80-bit master key and produces 32-bit round keys performing very simple operations.
In this work, we analyze the security of Khudra. We first show that the effective round key length is 16-bit. By the help of this observation, we improve the 14-round MITM attack proposed by Youssef et al. by reducing the memory complexity from $2^{64.8}$ to $2^{32.8}$. Also, we propose a new guess-and-determine type attack on 14 rounds where only 2 known plaintext-ciphertext pairs are required to mount the attack in a time complexity of 2^{64} encryption operations. To the best of our knowledge, this is the best attack in the single key model in terms of time, memory and data complexities where the data complexity is equal to the minimum theoretical data requirement. Moreover, we present two observations on differential probabilities of the round function and the symmetric structure of the cipher. We introduce 2^{40} weak keys for the full cipher by exploiting the symmetric structure of the cipher.

Key words: Cryptography, lightweight block cipher, guess-and-determine attack, meet-in-the-middle attack, Khudra cipher.

1 Introduction

Pervasive/ubiquitous computing and internet-of-things are growing computation concepts because of the increasing usage in daily life. In these computation models, addition to the big servers many resource constrained small devices are deployed. Lightweight cryptographic algorithms which can run in these constrained platforms have been required to provide the security of the traffic between the small devices. Since one type of the essential cryptographic algorithms is the block ciphers, a lot of lightweight block ciphers have been proposed. Most of the existing ciphers were designed especially for micro-controller or ASIC based

platforms such as ITUBEE [11], LED [10], PRESENT [5], PRINTCIPHER [12], PRIDE [1], Prince [6], RoadRunneR [2], SIMON and SPECK [3]. Khudra [13] is one of the lightweight block ciphers which was designed to meet the requirement of ciphers for FPGA based platforms.

Khudra was proposed at SPACE 2014 conference and is a competitive cipher according to the performance results given in a third party work [16]. The block size and key length of the cipher are 64 and 80 bits, respectively. The cipher is based on a generalized type-2 Feistel structure and consists of 18 rounds with whitening layers. In each round, a 32-bit round key is used and before the first round and after the last round half of the states are xored with the 32-bit pre- and post-whitening keys, respectively. The key schedule is very simple. 80-bit master key is divided into 5 16-bit parts and 32-bit round keys are created using the 5 parts iteratively and round constants are used to differentiate the round functions.

While Khudra was proposed in 2014, to the best of our knowledge, there have been 3 papers about its security [14, 18, 17]. In [14], a related-key rectangle attack on 16-round Khudra without whitening layers was introduced. The attack works under 4 related keys and the time and data complexities are $2^{78.68}$ encryptions and $2^{57.82}$ chosen plaintexts, respectively. In the attack proposed in [18], 2^{16} 14-round related-key impossible differentials were used to attack on the full Khudra. It requires 2^{63} related-key chosen plaintexts, 2^{64} memory and a time of $2^{68.46}$ encryption operations. To the best of our knowledge the only attack in the single key model was introduced in [17]. This attack is a meet-in-the-middle type attack [8] and works on 14 rounds with $2^{66.19}$, 2^{51} and $2^{64.8}$ time, data and memory complexities, respectively.

In this paper, we analyze the security of Khudra in the single key scenario. We first show that the effective round key length is 16-bit while the length of round keys is 32-bit and give the reason behind this behavior of the cipher. With this observation, we are able to improve the previous MITM attack on 14 rounds [17] by reducing the memory usage from $2^{64.8}$ to $2^{32.8}$. Also, we propose a guess-and-determine [7] type attack on 14 rounds where only 2 known plaintext-ciphertext pairs are enough to recover the key. The time complexity of the attack is 2^{64} encryption operations and the memory usage is negligible. The need of only 2 known plaintexts makes this attack very attractive because the required data can be practically collected in a lightweight application. In addition to the new attacks, we show that the differential probabilities of round function given by the designers are not realistic. While it was given that maximum differential probability of F function is 2^{-12}, we have found that the probability is $2^{-9,49}$. This observation enables us to show that more than the number of rounds claimed by the designers can be attacked using the differential attack [4]. Our final observation is on the weak keys. We introduce 2^{40} weak keys for the whole cipher and show the importance of the choice of round constants in the design of a cipher having a symmetric structure like Khudra.

We organized the paper as follows. In Section 2, we introduce the notations used in the paper and give the original and an alternative definition of Khudra.

The improvement of the previous MITM attack is presented in Section 3. We propose a guess-and-determine type attack on 14-round cipher in Section 4. In Section 5, we give our observations on differential probabilities and weak keys. Section 6 concludes the paper.

2 Definitions of Khudra

2.1 Notations

Throughout the paper, we use the following notations.

$\|$	: Concatenation operation
K	: 80-bit master key
k_i	: 16 bits of K where $K = k_0\|k_1\|k_2\|k_3\|k_4$
$(x)_i$	: Binary representation of the integer x in i bits where the most significant bit is the left most bit
RC_i	: 16-bit constant used in rounds where $RC_i = (0)_1\|(i)_6\|(0)_2\|(i)_6\|(0)_1$
X_i	: Output of the i-th round where $18 \geq i \geq 1$ and X_0 is the input of the first round
$A[i]$	: Left most i-th 16 bits of the bit string A where $i \geq 0$ and $A[0]$ is the left most 16 bits
$A\{i\}$	: Left most i-th nibble of the bit string A where $i \geq 0$ and $A\{0\}$ is the left most nibble
$A[i, ..., j]$	: $A[i]\|...\|A[j]$
X_i^j	: Output of the i-th round for plaintext P^j.

2.2 Original definition of Khudra

Khudra [13] is a lightweight block cipher having a generalized type-2 Feistel structure (GFS). The block size and key length are 64 and 80 bits, respectively. The cipher consists of 18 rounds and key whitening operations. The input of the first round is calculated as $X_0 = (P[0] \oplus k_0)\|P[1]\|(P[2] \oplus k_1)\|P[3]$ using the pre-whitening key where P is the plaintext. The ciphertext C is produced performing the post whitening operation to the output of the last round as follows: $C = X_{18}[0]\|(X_{18}[1] \oplus k_4)\|X_{18}[2]\|(X_{18}[3] \oplus k_3)$. i-th round function depicted in Figure 1 generates the output performing the following operations:

- $X_i[0] = F(X_{i-1}[0]) \oplus X_{i-1}[1] \oplus RC_{2i-2} \oplus k_{(2i-2)mod5}$
- $X_i[1] = X_{i-1}[2]$
- $X_i[2] = F(X_{i-1}[2]) \oplus X_{i-1}[3] \oplus RC_{2i-1} \oplus k_{(2i-1)mod5}$
- $X_i[3] = X_{i-1}[0]$

where F is a permutation on 16 bits which has a iterated structure and based on a 6-round GFS. One iteration of F function seen in Figure 2 produces 16-bit output Y for a given 16-bit input X as follows: $Y\{0\} \leftarrow s(X\{0\}) \oplus X\{1\}$, $Y\{1\} \leftarrow X\{2\}$, $Y\{2\} \leftarrow s(X\{2\}) \oplus X\{3\}$, $Y\{3\} \leftarrow X\{0\}$ where s is the S-box used in PRESENT [5]. For a detailed definition one can refer to [13].

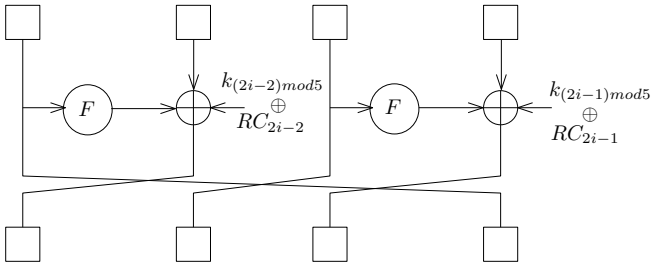


Fig. 1. i-th round of Khudra.

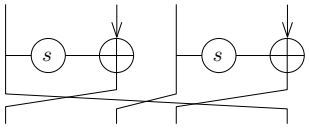


Fig. 2. One round of F function.

2.3 An Alternative Definition of Khudra

In the original definition of Khudra, it is given that the length of round keys is 32-bit. However, we observed that the effective key length for one round is 16-bit which can be seen from the following definition. The reason behind of this is the xoring of the same 16-bit key part on the same branch (e.g $X_i[1]$ and $X_{i+2}[3]$ is the same branch and the keys xoring to these 16 bits are $k_{(2i)mod5}$ and $k_{(2i+5)mod5}$ which are equal key parts).

In the new definition, the pre- and post-whitening keys are $((k_0 \oplus k_3)\|(0)_{16}\|k_1\|k_1)$ and $(k_4\|k_4\|(0)_{16}\|(k_2 \oplus k_3))$ respectively for a given 80-bit key $(k_0\|k_1\|k_2\|k_3\|k_4)$. First round key is the left most 16 bits of $(k_3\|k_0\|k_2\|k_4\|k_1)$ and the next 16-bit part is cyclically used in each successive rounds. We give the alternative definition of Khudra in Algorithm 1 and picture the i-th round in Figure 3 where F is the same F function in the original definition.

It is trivial to see that both definitions give the same encryption algorithm, Khudra.

3 An Improvement on the Youssef et al. Attack

3.1 Youssef et al. attack

In this section, we briefly summarize Youssef et al. attack [17]. They uses the following 6-round distinguisher to attack 14 rounds. The maximum number of possible values for the ordered sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$ is 2^{64} instead of $2^{7\times16} = 2^{112}$ when the set $\{X_3^0, X_3^1, ..., X_3^7\}$ is a 3-δ-set [9] in which each element only differs in the least significant 3 bits because the

Algorithm 1 An alternative definition of Khudra.

Input: 64-bit plaintext (P), 80-bit key ($K = (k_0 \| k_1 \| k_2 \| k_3 \| k_4)$)
Output: 64-bit ciphertext (C)
$X_0 \leftarrow (P \oplus ((k_0 \oplus k_3) \| (0)_{16} \| k_1 \| k_1))$
for $i = 1$ to 18 **do**
    $X_i[0] \leftarrow F(X_{i-1}[0] \oplus k_{(2i+1) mod 5}) \oplus X_{i-1}[1] \oplus RC_{2i-2}$
    $X_i[1] = X_{i-1}[2]$
    $X_i[2] = F(X_{i-1}[2]) \oplus X_{i-1}[3] \oplus RC_{2i-1}$
    $X_i[3] = X_{i-1}[0]$
end for
$C \leftarrow (X_{18} \oplus (k_4 \| k_4 \| (0)_{16} \| (k_2 \oplus k_3)))$

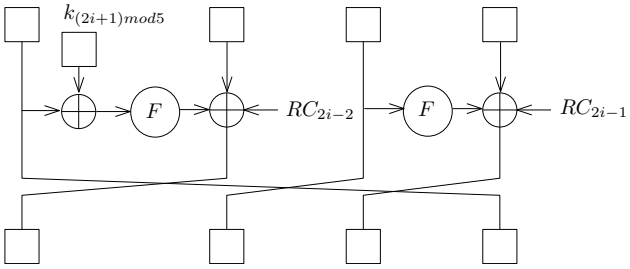


Fig. 3. i-th round of Khudra in the alternative definition.

4 16-bit parameters ($X_4^0[2], X_5^0[2], X_6^0[2], X_7^0[2]$) is enough to evaluate the value of the sequence. In the offline phase of the attack, the possible values for the sequence are computed and stored in a table indexed by the sequence value. In the online phase, for each guess of k_0, k_1 and k_2 8 plaintexts are found whose output of the 3-th round generate the 3-δ-set. Also k_4 is guessed to compute the difference sequence ($X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1]$) from the corresponding ciphertexts in the backward direction. If the computed sequence exists in the table then the guessed key value for k_0, k_1, k_3 and k_4 is given as a candidate key. For one guess of k_0, k_1, k_3 and k_4 the computed sequence value will be in the precomputed table with a probability of $2^{64} \times 2^{-112} = 2^{-48}$. Since the number of possible values of the keys is 2^{64}, there will remain about 2^{16} candidate keys. When k_3 is considered, the total number of possible keys will be 2^{32} which can be tested in a time about 2^{32} encryption operations.

The time complexity of the attack is about $2^{66.19}$ encryption operations because for each of the 64-bit guesses in the offline and online phases partial encryption operations are performed for 8 plaintexts. Since the plaintexts used in the attack can take any values except $P[2]$ which is an element of a 3-δ-set, the data requirement is about 2^{51}. In the offline phase, $2^{64.8}$ memory is required to store the 2^{64} possible values of the ordered sequence. One can refer to [17] for details.

3.2 Improvement of the attack

We reduce the memory usage of Youssef et al. attack to $2^{32.8}$ 64-bit blocks by giving a condition to the input of the 6-round distinguisher and guessing 3 16-bit variables in the offline phase (lets call it phase 1) where one of the 16-bit guess is the same guess required in the online phase (lets call it phase 2).

The input of the distinguisher is a 3-δ-set $\{X_3^0, X_3^1, ..., X_3^7\}$ where each element only differs in the least significant 3 bits and the left most 16 bits of each element is a constant value (we choose this constant as $(0)_{16}$ in the attack for simplicity). While this constant condition gives an 16-bit advantage in phase 1, it can be satisfied freely without any extra guess in phase 2.

We use our definition to explain the new attack given in Algorithm 2. The computations in the attack are pictured in Figure 4.

In phase 1, as seen in Figure 4 the guesses of k_1, $X_3^0[2]$ and $x_4^0[0]$ is enough to compute the sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$ which is performed in Step 6 in Algorithm 2 when $X_3^i[0]$ and $X_3^i[3]$ is known for $i \in \{0, 1, ..., 7\}$ and $X_3^i[3]$ differs only at the least significant 3 bits. The 2^{48} possible values for the sequence are computed and stored in a table called V in Step 7 in Algorithm 2.

In phase 2, k_1, k_2 and k_0 are guessed to find 8 plaintexts where $X_3[0] = 0$ for all plaintexts, $X_3[3] = i$ for i-th plaintext and $X_3[1]$ and $X_3[2]$ are same for all plaintexts in Step 12 as pictured in Figure 4.

By performing an extra guess for k_4, the difference sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$ is computed from the corresponding ciphertexts as depicted in Figure 4 and the values of the sequence is checked in Table V in Step 15 and 16. If the value is in the table, k_3 is guessed to reach the whole key and the 80-bit key is tested using a plaintext-ciphertext pair in Step 18. It is expected that only one key candidate passes the test. Note that k_1 is guessed for forward and backward computations so we collect the computations in the same loop in Step 1 to reduce the memory to store the possible values for the sequence.

The time and data complexities of the attack are equal to the time and data complexities of the Youssef et al. attack. The memory usage of our attack is about $2^{32.8}$ while the previous attack requires $2^{64.8}$. The main difference of our attack is on the guessed values in the computation of the possible values of the sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$.

4 A Guess-and-Determine Attack on 14 Rounds

We propose a guess-and-determine type attack on 14 rounds of Khudra according to the our new definition. The 14-round cipher is the first 14 rounds and includes the post-whitening layer. The main advantage of this attack over the 14-round MITM attack [17] is the data complexity. Our attack requires only 2 known plaintext-ciphertext pairs which can be reached for practical attacks.

The attack procedure is introduced in Algorithm 3. The known, guessed and determined 16 bits in the attack algorithm are pictured in Figure 5.

Algorithm 2 Improved MITM Attack on 14-round Khudra.

for all possible values of k_1 **do**
  Initialize Table V
  $X_3^0[0] \leftarrow 0, X_3^1[0] \leftarrow 0, ..., X_3^7[0] \leftarrow 0$
  $X_3^0[3] \leftarrow 0, X_3^1[3] \leftarrow 1, ..., X_3^7[3] \leftarrow 7$
  for all possible values of $(X_3^0[2], X_4^0[0])$ **do**
    Compute the sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$ in forward direction
    Store the sequence in Table V indexed by the sequence value
  end for
  $X_1^0[0] \leftarrow 0, X_1^1[0] \leftarrow 0, ..., X_1^7[0] \leftarrow 0$
  $X_2^0[2] \oplus k_3 \leftarrow 0, X_2^1[2] \oplus k_3 \leftarrow 0, ..., X_2^7[2] \oplus k_3 \leftarrow 0$
  for all possible values of (k_0, k_2) **do**
    Compute $X_2^i[0]$, $X_2^i[1]$ and P^i for $i \in \{0, 1, ..., 7\}$
    Get the corresponding ciphertexts
    for all possible values of k_4 **do**
      Compute the sequence $(X_9^0[1] \oplus X_9^1[1], X_9^0[1] \oplus X_9^2[1], ..., X_9^0[1] \oplus X_9^7[1])$ in backward direction
      if The computed sequence exists in Table V **then**
        for all possible values of k_3 **do**
          Test the key using a plaintext-ciphertext pair. If test is OK then output k_0, k_1, k_2, k_3, k_4 as the correct key.
        end for
      end if
    end for
  end for
end for

Algorithm 3 Guess-and-determine attack on 14-round cipher.

Input: 2 plaintext-ciphertext pairs (P_1, C_1) and (P_2, C_2)
Output: 80-bit key $(K = (k_0 \| k_1 \| k_2 \| k_3 \| k_4))$
for all possible values of (k_0, k_2, k_3, k_4) **do**
  Compute $X_1, X_2, X_3, X_4, X_5[1, 2, 3], X_6[1, 2], X_7[1]$ using P_1.
  Compute $X_{14}, X_{13}, X_{12}, X_{11}, X_{10}, X_9[0, 2, 3], X_8[0, 1], X_7[2]$ using C_1.
  $X_5[0] = X_6[3] \leftarrow F(X_6[2]) \oplus X_7[2] \oplus RC_{13}$
  $k_1 \leftarrow F^{-1}(X_5[0] \oplus X_4[1] \oplus RC_8) \oplus X_4[0]$
  $X_7[3] = X_6[0] \leftarrow F(X_5[0] \oplus k_3) \oplus X_5[1] \oplus RC_{10}$
  $X_8[3] = X_7[0] \leftarrow F(X_6[0] \oplus k_0) \oplus X_6[1] \oplus RC_{12}$
  if $X_8[0] = F(X_7[0] \oplus k_2) \oplus X_7[1] \oplus RC_{14}$ **then**
    $X_9[1] = X_8[2] \leftarrow F(X_7[2]) \oplus X_7[3] \oplus RC_{15}$
    if $(X_9[2] = F(X_8[2]) \oplus X_8[3] \oplus RC_{17})$ and $(X_{10}[0] = F(X_9[0] \oplus k_1) \oplus X_9[1] \oplus RC_{18})$ **then**
      if 80-bit key $(k_0, k_1, k_2, k_3, k_4)$ satisfies the (P_2, C_2) pair **then**
        Output the key
      end if
    end if
  end if
end for

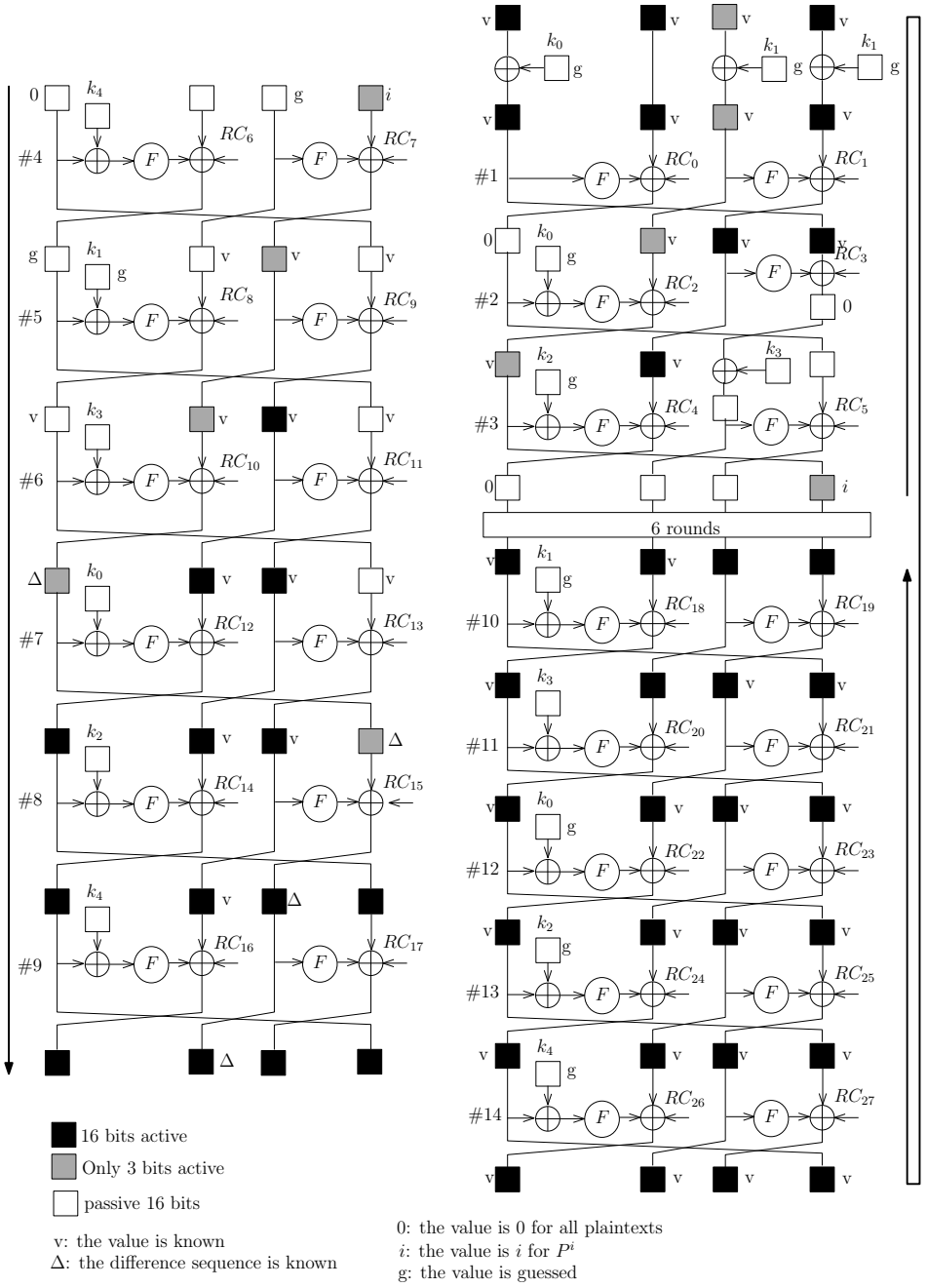
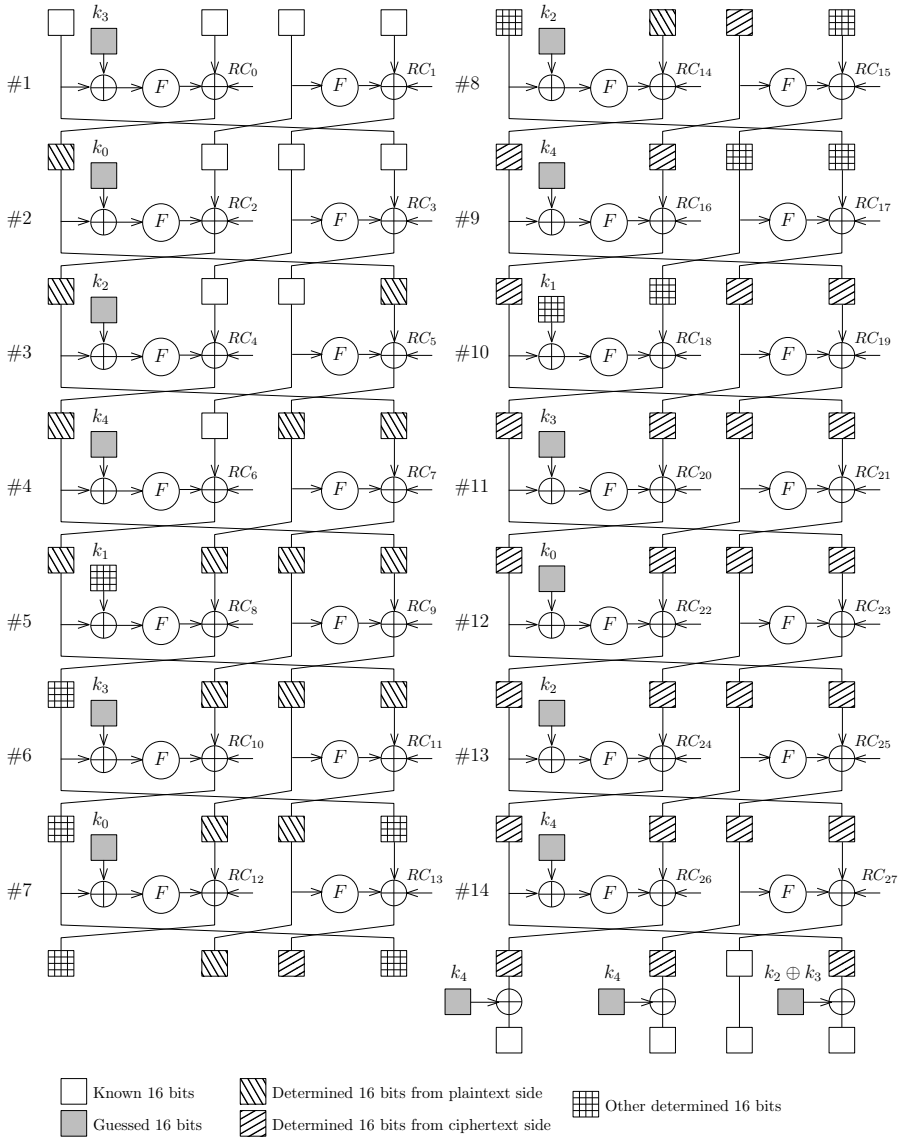


Fig. 4. Improved MITM attack. Computations in phase 1 and phase 2 are given on the left and right, respectively.



RC

Fig. 5. Known, guessed and determined 16 bits in Algorithm 3.

In the attack we use only 2 known pliantext-ciphertext pairs and the memory requirement is negligible. For 2^{64} different guesses of (k_0, k_2, k_3, k_4) by performing partial encryptions for one plaintext-ciphertext pair we determine a value for k_1 and check some conditions given in Step 10 and 12. If the 48-bit condition satisfied the candidate key is checked using the other plaintext-ciphertext pair in Step 13. This step is executed nearly $2^{64} \times 2^{-48} = 2^{16}$ times in total. The condition in Step 13 is satisfied only for the correct key with a probability near to 1. As a result the time complexity of the attack is 2^{64} partial encryption operations.

5 Other observations

5.1 Differential properties

It is stated in the cipher proposal that the maximum probability of a differential for F function is $(2^{-2})^6 = 2^{-12}$, because the s-box used in the function is PRESENT's S-box and the maximum probability is 2^{-2} and there are at least 6 active S-boxes in the case of the function is active. However, we show that the probability is not realistic due to the more than one differential characteristics and the data dependencies of the input of the active S-boxes. Since F function operates on 16 bits and the key is not used in the function, we can consider the function as a 16-bit S-box. We find that the maximum probability for this big S-box is $2^{-9.48}$ by an exhaustive search. Under this knowledge with the minimum number of active F functions for 7 round given by the designers, it can be said that the maximum probability of a differential for 7-rounds is $(2^{-9,48})^6 = 2^{-56.88}$. It can be concluded that a 7-round differential characteristic can be used in a differential attack while the designers' claim is that there is no differential characteristic for at least 6 rounds usable in an attack. Similar observation can be found for the linear cryptanalysis [15].

5.2 Weak keys

In this section, we introduce 2^{40} weak keys for full Khudra. We show that the encryption algorithm under a weak key can be distinguished easily because of the usage of symmetric round constants in addition to the symmetric structure of the algorithm.

Let A, B and C be sets of 16-bit, 64-bit and 80-bit words defined as $A = \{X : X\{0\} = X\{2\} \text{ and } X\{1\} = X\{3\}\}$, $B = \{X : X[0], X[1], X[2], X[3] \in A\}$, and $C = \{X : X[0], X[1], X[2], X[3], X[4] \in A\}$, respectively. When the input of F comes from the set A then the output will also in A because the output of first round is $((s(a) \oplus b \| a \| (s(a) \oplus b \| a)$ for a given input $(a \| b \| a \| b)$ where a and b 4-bit values and F consists of 6 same rounds. Each round constant $RC_i = (0)_1 \| (i)_6 \| (0)_2 \| (i)_6 \| (0)_1$ is also an element of A and A is closed under the xor operation. Thus, the addition of round constants does not destroy this property. If a key from the set C is used, encryption of a plaintext in the set

B will result a ciphertext which is also in B. This structural behaviour of the algorithm gives a distinguisher for full round. When we enumerate the number of elements in the set C, we see that the number of weak keys is 2^{40}.

6 Conclusion

In this work, we analyzed the security of Khudra and showed that focusing more on performance in the design of a cipher can lead to unexpected behaviors of the cipher in terms of security. The designers chose the key schedule and the places where the round keys are added to the state in a way to optimize the performance of the algorithm in FPGA. However, we observed that these choices reduces the effective key length from 32-bit to 16-bit. We proposed the best attacks in the single key model by exploiting this observation. One attack reduces the memory requirement of the MITM attack from $2^{64.8}$ to $2^{32.8}$ proposed in [17]. The other attack is a type of guess-and-determine attack and requires only 2 known plaintext-ciphertex pairs. The time complexity of this attack is 2^{64} encryption operations and need a negligible memory. Also, we investigated the differential behavior of the round function concluding that the differential attack can be mount on more rounds than the number of rounds claimed by the designers. Finally, we showed the importance of the selection of round constants for a cipher having symmetric structure introducing 2^{40} weak keys for the full Khudra.

References

1. Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun, Gregor Leander, Christof Paar, and Tolga Yalçin. Block ciphers - focus on the linear layer (feat. PRIDE). In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 57–76. Springer, 2014.
2. Adnan Baysal and Suhap Sahin. Roadrunner: A small and fast bitslice block cipher for low cost 8-bit processors. *IACR Cryptology ePrint Archive*, 2015:906, 2015.
3. Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK lightweight block ciphers. In *Proceedings of the 52nd Annual Design Automation Conference, San Francisco, CA, USA, June 7-11, 2015*, pages 175:1–175:6. ACM, 2015.
4. Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *J. Cryptology*, 4(1):3–72, 1991.
5. Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.

6. Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.
7. Charles Bouillaguet, Patrick Derbez, and Pierre-Alain Fouque. Automatic search of attacks on round-reduced AES and applications. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 169–187. Springer, 2011.
8. Hüseyin Demirci and Ali Aydin Selçuk. A meet-in-the-middle attack on 8-round AES. In Kaisa Nyberg, editor, *Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers*, volume 5086 of *Lecture Notes in Computer Science*, pages 116–126. Springer, 2008.
9. Jian Guo, Jérémy Jean, Ivica Nikolic, and Yu Sasaki. Meet-in-the-middle attacks on generic feistel constructions. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 458–477. Springer, 2014.
10. Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer, 2011.
11. Ferhat Karakoç, Hüseyin Demirci, and A. Emre Harmanci. Itubee: A software oriented lightweight block cipher. In Gildas Avoine and Orhun Kara, editors, *Lightweight Cryptography for Security and Privacy - Second International Workshop, LightSec 2013, Gebze, Turkey, May 6-7, 2013, Revised Selected Papers*, volume 8162 of *Lecture Notes in Computer Science*, pages 16–27. Springer, 2013.
12. Lars R. Knudsen, Gregor Leander, Axel Poschmann, and Matthew J. B. Robshaw. Printcipher: A block cipher for ic-printing. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, volume 6225 of *Lecture Notes in Computer Science*, pages 16–32. Springer, 2010.
13. Souvik Kolay and Debdeep Mukhopadhyay. Khudra: A new lightweight block cipher for fpgas. In Rajat Subhra Chakraborty, Vashek Matyas, and Patrick Schaumont, editors, *Security, Privacy, and Applied Cryptography Engineering - 4th International Conference, SPACE 2014, Pune, India, October 18-22, 2014. Proceedings*, volume 8804 of *Lecture Notes in Computer Science*, pages 126–145. Springer, 2014.
14. Xiaoshuang Ma and Kexin Qiao. Related-key rectangle attack on round-reduced *Khudra* block cipher. *IACR Cryptology ePrint Archive*, 2015:533, 2015.
15. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993,*

Proceedings, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

16. Bassam J Mohd, Thaier Hayajneh, and Athanasios V Vasilakos. A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues. *Journal of Network and Computer Applications*, 2015.
17. Mohamed Tolba, Ahmed Abdelkhalek, and Amr M. Youssef. Meet-in-the-middle attacks on round-reduced khudra. In Rajat Subhra Chakraborty, Peter Schwabe, and Jon A. Solworth, editors, *Security, Privacy, and Applied Cryptography Engineering - 5th International Conference, SPACE 2015, Jaipur, India, October 3-7, 2015, Proceedings*, volume 9354 of *Lecture Notes in Computer Science*, pages 127–138. Springer, 2015.
18. Qianqian Yang, Lei Hu, Siwei Sun, and Ling Song. Related-key impossible differential analysis of full khudra. *IACR Cryptology ePrint Archive*, 2015:840, 2015.