Attacks on Blowfish Block Cipher: An Overview

Hesham T. Banafa
Electrical and Computer Engineering Department, King Abdulaziz University

Nov 10, 2022

Abstract

Modern networking and use of online services is a critically important to most people. Conducting this form of communication requires a great deal of confidentiality, integrity, and authentication. Blowfish is a component of modern cryptography suites that enables confidentiality as a 64-bit block cipher, initially released as public domain in 1994. Security strength tests and cryptanalysis is applied to all ciphers to ensure confidentiality of messages. Most 64-bit block ciphers are now obsolete. However, Blowfish still holds as cryptographically sound. In this paper, we explore the possibility of attacks being mounted on current implementations of Blowfish, and the feasibility of using Blowfish in new deployments is examined. We conclude that Blowfish still holds to be secure for general application use, when limitations of the use of CBC mode using a single encryption context, and inserting key strength tests during the key generation phase.

Keywords: Blowfish, Reflectively Weak Key, Linear cryptanalysis, Block Cipher, Birthday Attack

1 Introduction

The prevalence of substitution boxes S-boxes in almost all cryptographic systems shows the important role of injecting confusion [1]. Moreover, S-Boxes one of the most important determining factors of the strength of a cryptographic system, and are non-linear [2]. Blowfish derives it's S-Boxes from the secret key. This by design is quite secure, and unfortunately may add compute time when compared to the Advanced Encryption Standard (AES), and Data Encryption Standard (DES). Moreover, the key schedule when compared to AES and DES in Blowfish is more complex, and requires the Blowfish cipher to be executed 521 times to generate all the subkeys, processing 4KB of data [3]. In theory, the security of a system/cipher depends on how much compute, and time is available to the cryptanalyst [1]. Usually an assumption of unlimited resources is used to evaluate the security of a cipher.

In contrast to most other crypto systems, Blowfish is public domain specification, without any legal remafications for using, extending or selling the original, or enhanced version of Blowfish. RC2, RC4 are approved for small key to be exported outside of the United States. Khufu REDOC II and IDEA are under patent. GOST is a block cipher from the Soviet era, without S-boxes disclosed for security. SkipJack is cipher developed by United States National Security Agency, and was classified information for a long time before being declassified in 1998 [4, 5].

In absence of hardware acceleration, the work in [6] shows advantageous performance when compared to AES and DES in a software environment. In [7], have shown a Verilog HDL implementation of BF with a constant delay adder with improved timing constraints, therefore a higher operating frequency, in hardware acceleration context. Furthermore, in energy constrained environments, which become more important to the conclusion to this paper, the authors in [8] have show a better power-throughput ratio using FPGA as the hardware platform.

VLSI implementaions gained a factor of 9 times improved performance [9], by operator-reschedualing to reduce critical path delay. k_1 and k_18 array in registers, and others are stored in static random access memory. This implementation is also cascadeable for more performance when required.

For the reasons above, Blowfish is a valid candidate for consideration. However, security concerns are important, and should be analyzed. Generally, in security analysis and crytoanalysis, an "attack" becomes significant and worthy of note when the methodology reduces the search space and complexity to a value lower than a basic exhaustive search or brute-force. In this paper, we

denote multiple methodologies that exploit conditional weaknesses in different phases of the Blowfish algorithm.

Conducting this strength analysis is critical, since Blowfish is used commonly in PHP based web applications for password hashing. BF is also used in many encryption suites, with an example of GnuPG[1]. It follows that if an attack with the results if extracting the secret key, in some cases may lead to recovery of the original text, being the password. We note that this is the case for naive implementations of Blowfish, where the text or password fits into one block. In this paper, we explore the possibility of successful attack and address concerns regarding existing systems using Blowfish.

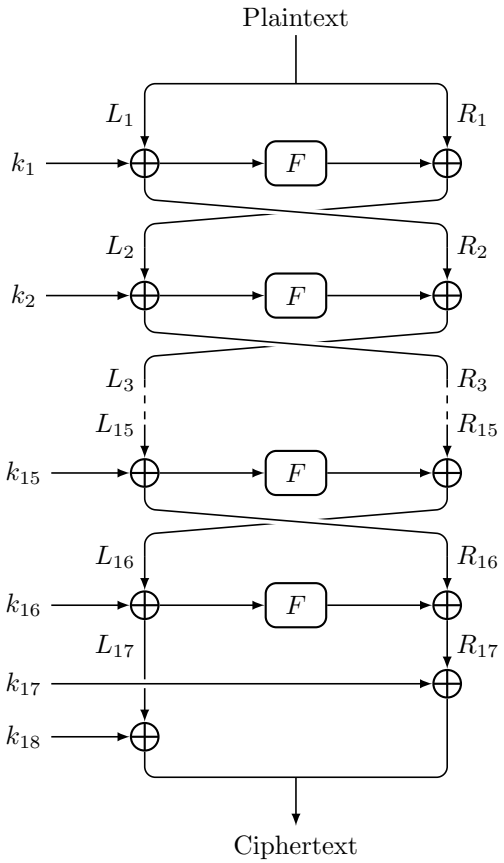

Figure 1: Blowfish Fiestel Network

2 The Blowfish Block Cipher

Blowfish is a symmetric block cipher with 64-bit blocks, and a variable length key up to 448-bits. Initially designed and released in 1993 by Bruce Schneier as a public-domain, unpatented block cipher [4, 3]. The cipher makes use of a modified fiestel network as the base, as shown in Fig. 1, with the "bypass" half being L, as opposed to R in a traditional Fiestel Cipher.

[1]https://gnupg.org

Note from Fig.1 that the modification of the the original Fiestel Network can be summrized as

$$R_i = L_{i-1} \oplus k_{i-1} \tag{1}$$

Where in the traditional fiestel network

$$R_i = L_{i-1} \tag{2}$$

And it follows that from Fig.1 that

$$L_i = (F(L_{i-1} \oplus k_{i-1}) \oplus R_{i-1}) \tag{3}$$

The cipher is completed after 16 rounds.

2.1 Initialization

The initialization steps in Blowfish are required to populate the S-Boxes, and expand the provided variable key.

The key is expanded into an array of 18 32-bit subkeys, denoted heron after as P

$$k_1, k_2, \ldots, k_{18}$$

The array is initialized by using the hexadecimal digits of π, and the key k as the inputs to the key schedule. P and S arrays are initialized by the digits of π.

Algorithm 1 k Array Inilization

```
for i = 0 to 18 do
    k[i] = k[i] ⊕ K[i mod KeyLength]
end for
datal = 0
datar = 0
for i = 0 to 18; i + 2 do
    BFEnc(datal, datar, k)
    k[i] = datal
    k[i + 1] = datar
end for
```

The S-Box inilization algorithm is denoted in Alg. 2.

Algorithm 2 S-Boxes Array Inilization

```
for i = 0 to 4 do
    for j = 0 to 255; i + 2 do
        BFEnc(datal, datar, k)
        S[i][j] = datal
        S[i][j + 1] = datar
    end for
end for
```

From algorithms 1 and 2, the encryption context is ready to encrypt plain text, using the secret key dervied S-Boxes.

2.2 Blowfish Round Function

The Blowfish function F introduces a new concept, by containing a set of 4 S-boxes, with the content being derived from the secret key k, as denoted in 2. In each Encryption and decryption round, the value L_i depends on the function F. In the following we describe the function and a diagram is presented.

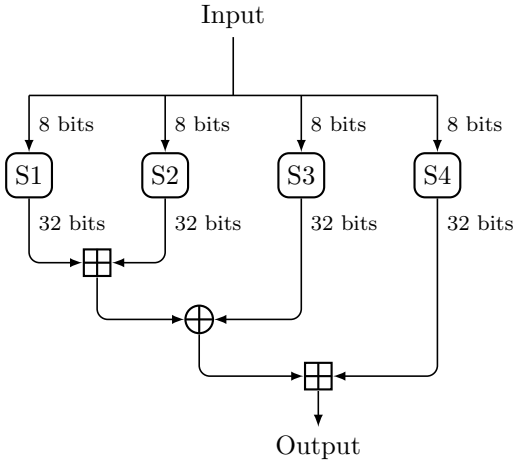

Figure 2: Blowfish Fiestel Function F

From Fig. 2, the input half block $L_i \oplus k_i$ is split into 8-bit blocks, which values are used to index into the S-Boxes as lookup tables. This *substitution* step expands each block and operates on each substitution in intermediate steps as follows.

$$out = S4 + [S3 \oplus (S2 + S1)] \tag{4}$$

Note, all addition operations in Eq. 4 are modulo 2^{32}. With this in hand, implementors are able to use this block cipher in any mode of encipherment, such as ECB, CBC, CTR, or any others that apply to asymmetric block ciphers.

3 Reflectively Weak Keys

Kara and Manap [10] have shown that there exists a set of keys referred to as *Reflectively Weak Keys* for Blowfish. The work builds on a more general approach described in [11], which describes a new term *Degree of Similarity* in round functions, of Fiestel Network ciphers. The work shows two new model descriptions of Blowfish are equivalent to the original description. However, the models described help deduce reflection properties, that are considered a weakness and leads to reduction in time complexity of key search. The original model is

$$
\begin{aligned}
(L_2, R_2) &= (F(k_1 \oplus L_1) \oplus R_1, k_1 \oplus L_1) \\
(L_3, R_3) &= (F(k_2 \oplus L_2) \oplus R_2, k_2 \oplus L_2) \\
&\dots \\
(L_{17}, R_{17}) &= (F(k_{16} \oplus L_{16}) \oplus R_{16}, k_{16} \oplus L_{16}) \\
(L_{18}, R_{18}) &= (F(k_{17} \oplus L_{17}) \oplus R_{17}, k_{17} \oplus L_{17})
\end{aligned}
$$

Kara *et al.* in [10] move the XOR operator, since it is communicative. Hence, it can be moved through other XOR operators with the limiting factor being any non-communicative operations. In this case the round function F, is the non-communicative operation. Many forms can be reached. However, the writer denote two types, namely *Type II Description of Blowfish* and *Type III Description of Blowfish.*

$$
\begin{aligned}
(L_1, R_1) &= (k_1 \oplus L, k_2 \oplus R) \\
(L_2, R_2) &= (F(L_1) \oplus R_1, L_1) \\
(L_3, R_3) &= (F(R_2) \oplus L_2, R_2 \oplus k_4) \qquad (5) \\
(L_4, R_4) &= (F(R_3 \oplus k_3) \oplus L_3, R_3 \oplus k_3 \oplus k_5) \qquad (6) \\
(L_5, R_5) &= (L_4 \oplus F(R_4), R_4) \\
(L_6, R_6) &= (L_5 \oplus F(R_5), R_5) \\
&\dots \\
(L_{18}, R_{18}) &= (k_{18} \oplus R_{17}, k_{17} \oplus L_{17})
\end{aligned}
$$

Kara and Manap from formulations (5) and (6) prove that if $k_1 = k_4$ and $k_2 = k_3$, then the rounds (5) and (6) have 2^{32} *fixed points.* In other words, there exists A count of 2^{32} Plaintexts that remain unchanged in the intermediate steps 3 and 4. Hence, any keys that satisfy this are *Reflectively Weak Keys.* From [10], an approximation of the number of weak keys is $2^{k+32-16r}$. 2^{32} Plaintexts are required and used to identify weak keys. The authors also mention the use of pre-calculated key schedules for many keys, which shortens repeated exhaustive search time complexity, in a trade of with space complexity.

An attacker, hence, tests to see if a weak key is used, and can extract information about the k array. This however requires 2^{32} plaintext- ciphertext pairs to make the attack feasible. This evaluates to around 512GB of data to be held in memory, which is in reach in terms of space complexity. The attacker can recover about half the key, after guessing the first half.

Kara *et al.*, argue that this weakness is due to the degree of similarity in the functions that produce the round keys, despite being one-way functions. Furthermore, it is suggested that the length of secret key should not be larger than the block size.

4 Linear Cryptanalysis

Linear Cryptanalysis is one of the most important test of strength to applied to any new or proposed crypto system. Linear Cryptanalysis is a Known-Plaintext attack (KP). Nakahara in [12] defines LC as "A linear distingusiher which consists of a linear relationship between bits of plaintext, ciphertext, and key, holding with non-uniform porobability." This association between porobability in a cipher and of random behaviour is referred to as the bias, and is denoted by p'. It follows that the number of required known plaintext block required is invresly proportional to the bias, as given in [13, 12], $N = 8p'^{-2}$. Hence, a higher bias leads to a weaker cipher. This is a fundamental test of strength for most symmetric block ciphers. Similar to previously mentioned attacks, using working memory with per-calculated values significantly decreases the time complexity or compute time. Therefore, the authors in [13, 12] describe an exhaustive list of all inputs and outputs of a substitution box S is called Linear Approximation Table, or (LAT) of S. With LAT, one can identify linear relations with the highest p', thus, reducing the needed plaintext requirement.

Moreover, separate linear relations are then combined to form a relation on the round level, then multi-round. This lead, however, to a reduced bias p'. Since the design of Blowfish S-boxes are key dependent, it is not feasible to compte LAT from the point of view of an attacker or cryptanalyst. However, it is possible to calculate a LATs for a subset of S-Boxes derived from random keys. The S-Boxes in BF are non-surjective since they of the shape 8×32. A linear distingusiher can be made such as that the input $(00000000)_{18} \xrightarrow{F} (00000001)_{18}$ shown in 3. In other words, the relation propagates through the BF function F.

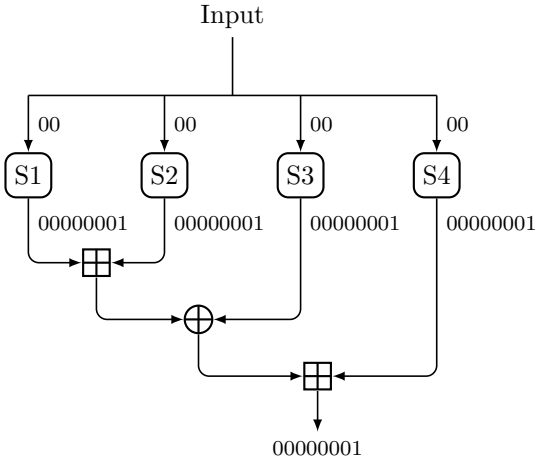

Figure 3: Hex representation of propagation of linear relation in Blowfish round Function F

The effect exhibited in Fig. 3, the work in [12] arrives at 2-round itirative linear relation. That is for input at round n is $(00000001, 00000000)$ is the same output in round $n+1$. Based on this, two bits from the k array of subkeys.

The conclusion and results to this research for a reduced number of rounds of BF, 6, only a small number of keys, about 0.0000013% of the keys tested have p' higher than $2^{-10.83}$. Nakahara in [12], therefor suggests a simple test for the susobtability of a linear attack on a particual set of S-Boxes. Fetch the entry $(00, 00000001)_{18}$ from the LAT for each S-Box, and verify that the bias p' is small or prefreably zero, using the Piling-Up Lemma.

5 Birthday Attack

Birthday attacks exploit the mathematics behind the birthday problem, that is based on probability theory. Usually this arises as a concern when the smallest unit of encryption is relatively small, in relation to the compute power available to cryptanalysts. The attacker depends on likelihood of collision in random attack attempts and a fixed degree of permutations (64-bit block cipher codebook). While in most cases, the birthday attack has similar complexity to exhaustive search, and therefore not feasible. However, the work in [14] demonstrates the possibility of recovering plaintext in long-living HTTP sessions over OpenVPN, that contain HTTP BasicAuth information[2]. OpenVPN uses Blowfish, and other 64-bit block ciphers. An attacker is required to collect 785GB of encrypted data related to *one* HTTP session that uses a 64-bit block cipher in CBC mode. The attack is executed in around 19-38 hours in a lab setting done in [14]. We note that the recovery of BasicAuth information is transmitted in every HTTP request.

5.1 Birthday Bound

The weakness herein lies in the combination of block size and mode of operation, CBC in this case. Block ciphers are meant to secure data with up to 2^n complexity for space and 2^k complexity for time, where n is the block size, and k is the key size. However, it is inevitable for common use of information transfer for data to exceed the block size of ciphers. Hence, modes of operation such as CBC are used to chain blocks of data. However, they are only proven to protect $2^{n/2}$ [14], where this limit is called *birthday bound*. After this bound, the probability of an n-bit block collision becomes of concern.

5.2 Mitigation

It is easy to see that one possible mitigation to this attack is to implement a re-initlizaiton of Blowfish after a certain amount of time, or blocks, on one Blowfish context. Another mitigation is to stop using *any* 64-bit block cipher in favor of 128-bit or higher block ciphers.

6 Conclusion

In this paper, the Linear attacks and other new forms such as reflectivity attacks were explored. The goal of cryptology is to ensure confidentiality in the public network, that is the internet. In most cases the information within transmitted, encrypted messages become of no use over time. Therefore, even with the attacks show in this paper, it is safe to conclude that the security of Blowfish is still standing, and in many cases where hardware acceleration of the Advanced Encryption Standard is not available, software implementations of Blowfish surpass AES in throughput [6], which is always better for energy constrained applications such as IoT. Barring tests of generated key strength for reflectivity and linear distinguishers. At the core of the discussed attacks, the

[2]HTTP BasicAuth contains Username and Password in clear text (not hashed version) (RFC7617)

main target of weakness is the key generation. This fact is similar to issues exposed in RC4 in the implementation of WPA for wireless communication.

The linear distingushier attacks are concluded to be not affecting the security of Blowfish by the researchers conducting the attacks. The attack is of low significance in a reduced number of rounds of BF. For the Reflectively Weak Keys, the Authors in publication did not make clear the feasibility of the attack, and how the "first half" of the k array is recovered, in our view, and in [15] this conclusion is reached. On the other hand, the birthday attack is of significance with vary low probability of occurrence. It is required for the attack to be mounted for *single* HTTP session to be transmitting ciphertexts for 19-38 Hours, *without stopping*. We see that the chances of a single long-lived HTTP session, being active for multiple days with practical data rate are miniscule. Furthermore, Transport Layer Security (TLS) implements re-keying for long-lived sessions for all block, which mitigates.

Attempts were done to enhance the initialization phase of Blowfish using the hash function SHA-256, however, the strength of this extension is not tested. The overhead of this extension did not increase significantly.

Concluding that the attacks that can be mounted against Blowfish are impractical outside of controlled environments. However, given the trend of block cipher obsolescence, Blowfish should not be used in newly developed systems, and should be limited to energy constrained environments and embedded systems. Moreover, Blowfish should not be used to encrypt data at rest larger than 4GB with a single key.

References

[1] C. E. Shannon. "Communication Theory of Secrecy Systems*". In: *Bell System Technical Journal 1949-oct vol. 28 iss. 4* 28 (4 Oct. 1949). DOI: 10.1002/j.1538-7305.1949.tb00928.x.

[2] Guoping Tang; Xiaofeng Liao; Yong Chen. "A novel method for designing S-boxes based on chaotic maps". In: *Chaos, Solitons & Fractals 2005-jan vol. 23 iss. 2* 23 (2 Jan. 2005). DOI: 10.1016/j.chaos.2004.04.023.

[3] Wikipedia. *Blowfish (cipher) — Wikipedia, The Free Encyclopedia*. [Online; accessed 2-November-2022]. 2022.

[4] Bruce Schneier. "Description of a new variable-length key, 64-bit block cipher (Blowfish)". In: *Fast Software Encryption*. Ed. by Ross Anderson. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 191–204. ISBN: 978-3-540-48456-1.

[5] Wikipedia contributors. *Skipjack (cipher) — Wikipedia, The Free Encyclopedia*. [Online; accessed 10-November-2022]. 2022.

[6] Kuntal Patel. "Performance analysis of AES, DES and Blowfish cryptographic algorithms on small and large data files". In: *International Journal of Information Technology 2019-jan 01 vol. 11 iss. 4* 11 (4 Jan. 2019). DOI: 10.1007/s41870-018-0271-4.

[7] Prabhu V Kumara Swamy; Benakop. "Predominance of Blowfish Over Triple Data Encryption Standard Symmetric Key Algorithm for Secure Integrated Circuits Using Verilog HDL". In: *International Journal of Network Security & Its Applications 2017-nov 30 vol. 9 iss. 6* 9 (6 Nov. 2017). DOI: 10.5121/ijnsa.2017.9603.

[8] Ahmad Rafidah, Manaf Asrulnizam Abd, and Ismail Widad. "[IEEE 2016 IEEE 12th International Colloquium on Signal Processing & Its Applications (CSPA) - Melaka, Malaysia (2016.3.4-2016.3.6)] 2016 IEEE 12th International Colloquium on Signal Processing & Its Applications (CSPA) - Development of an improved power-throughput Blowfish algorithm on FPGA". In: (Mar. 2016). DOI: 10.1109/cspa.2016.7515838.

[9] M.C.-J.; Youn-Long Lin Lin. "[IEEE ASP-DAC2000: Asia and South Pacific Design Automation Conference 2000 - Yokohama, Japan (25-28 Jan. 2000)] Proceedings 2000. Design Automation Conference. (IEEE Cat. No.00CH37106) - A VLSI implementation of the Blowfish encryption/decryption algorithm". In: (2000). DOI: 10.1109/aspdac.2000.835049.

[10] Orhun Kara and Cevat Manap. "A New Class of Weak Keys for Blowfish". In: *Fast Software Encryption*. Ed. by Alex Biryukov. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 167–180. ISBN: 978-3-540-74619-5.

[11] Orhun Kara. "Reflection Attacks on Product Ciphers". In: *IACR Cryptol. ePrint Arch.* 2007 (2007), p. 43.

[12] Jorge Nakahara. "A Linear Analysis of Blowfish and Khufu". In: *Information Security Practice and Experience*. Ed. by Ed Dawson and Duncan S. Wong. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 20–32. ISBN: 978-3-540-72163-5.

[13] Mitsuru Matsui. "Linear Cryptanalysis Method for DES Cipher". In: *Advances in Cryptology — EUROCRYPT '93*. Ed. by Tor Helleseth. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 386–397. ISBN: 978-3-540-48285-7.

[14] Karthikeyan Bhargavan and Gaëtan Leurent. "On the Practical (In-)Security of 64-Bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN". In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS '16. Vienna, Austria: Association for Computing Machinery, 2016, pp. 456–467. ISBN: 9781450341394. DOI: 10.1145/2976749.2978423.

[15] Tom Gonzalez. "A Reflection Attack on Blowfish". In: *Journal of LaTex Class Files* 6.1 (2007).