On the Key Schedule of Blowfish

Dieter Schmidt*

August 9, 2006

Abstract

In this article the author shows that for the block cipher Blowfish, the subkeys for the third and fourth round do not depend on the first 64 bits of the userkey, if the userkey is of maximal length.

1 Introduction

Key schedules are nearly as numerous as the underlying block ciphers, since virtually every cipher comes with its own schedule. All have in common, that the key schedule is usually finetuned to the properties of the underlying cipher. It shall facilitate a good diffusion of the bits of the userkey into the ciphertext, e.g. the cipher should use as few rounds as possible until it is complete with regard to the key bits. Moreover, the derivation of one of the subkeys by a cryptanalyst should give as little information as possible about the other subkeys or the userkey. The block cipher Blowfish [1] comes with a very complex key schedule that employs the underlying block cipher to derive the roundkeys from the userkey. While at first glance the key schedule looks perfect in the sense that every bit of every subkey depends on all the bits of the userkey, the author shows in his paper that the subkeys of the third and fourth rounds do not depend on the first 64 bits of the userkey, if the userkey is of maximal length.

The rest of the article is organized as follows: In section 2 we give a brief overview over the block cipher Blowfish and its key schedule. In section 3 we verify our claim and we conclude in section 4.

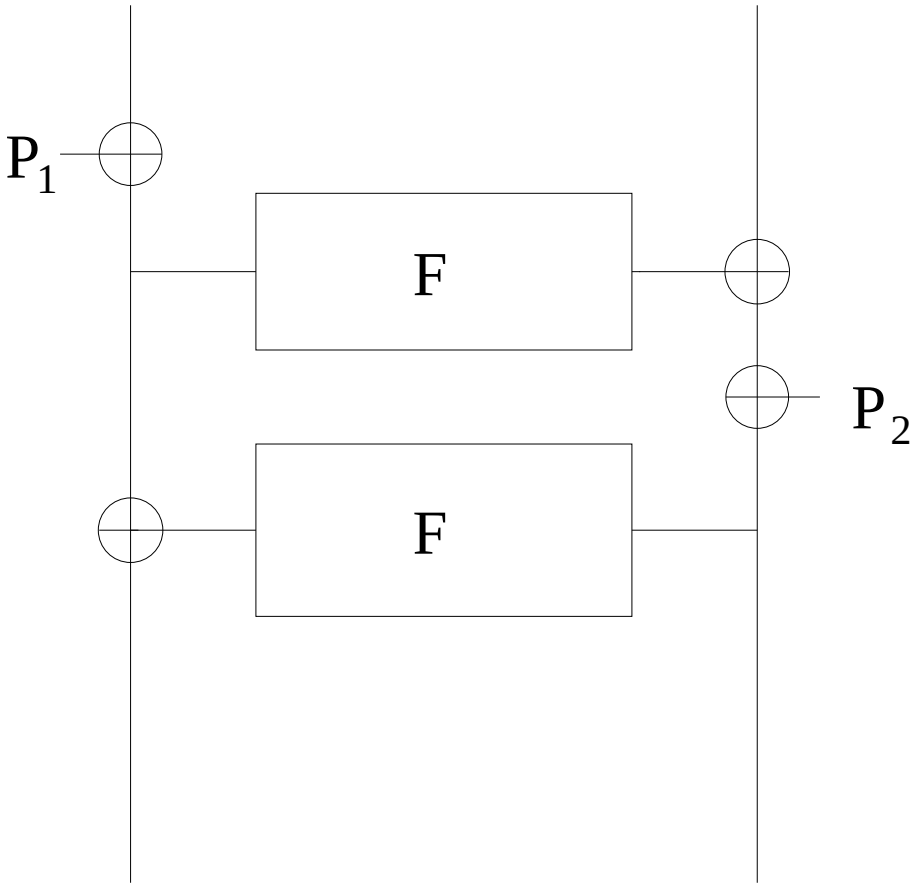

Figure 1: The first two rounds of Blowfish

2 Description of Blowfish and its Key Schedule

Blowfish is a block cipher with a block size of 64 bit and a key length of up to 448 bit. It is a generalized Feistel cipher, where the subkeys are not fed into the round function F, but rather are added modulo 2 (XORed) to the left and right halves of the data blocks alternately. The details of the round function are not necessary to understand our claim. Fig. 1 depicts the first two rounds of the 16 total rounds of Blowfish. The rounds are followed by a final transformation, details of which are not necessary in this article.

The subkeys which are added modulo 2 (XORed) before the round function is applied, are named P_1 to P_{16} (P_{17} and P_{18} are reserved for the final transformation). Initially the P_1 to P_{18} are assigned the hexadecimal digits of π in ascending order. If a userkey is not of maximal length, then it is appended to itself until a length of 448 bits is reached. For example, a user key A of 64 bits will result in a key of AAAAAAAA.The 448 bits of the user key are added modulo 2 (XORed) to P_1 to P_{14}. The derivation of the subkeys now works as follows: First encrypt the all-zero string. The output of that first encryption is assigned to P_1 and P_2. While it is not clear from description of Blowfish in [1, 2] which half of the ciphertext is assigned to P_1, the source code published in [2] makes it clear, that the left half of the ciphertext is assigned to P_1 and the right half to P_2. This property is crucial to the twist discovered by the author, since a reversal of that assignment would have made this paper obsolete. The fact that the order of assignment was not published in the original paper may have contributed to the fact that this property was long overlooked by the crypto community. The cipher is then employed in Output Feed Back Mode and the subsequent outputs are assigned to P_3 (right half of ciphertext) and P_4 (left half of ciphertext) and so on. This is repeated until all the elements up to P_{18} have been assigned new values.

3 Description of the Twist

Let $\mathbb{F}_2^{32}$ be the 32-dimensional vectorspace over $GF(2)$. Let $F : \mathbb{F}_2^{32} \mapsto \mathbb{F}_2^{32}$ be the round function of Blowfish. Let 0 be the null vector in $\mathbb{F}_2^{32}$. Consider the state after the first iteration: Let the output be X_L (left half) and X_R (right half). P_1 ist assigned X_L and P_2 is assigned X_R.

*Denkmalstrae 16,D-57567 Daaden, Germany, dieterschmidt@usa.com

When the second iteration starts, X_L is XORed with P_1, leaving the left half of the data:

$$\mathrm{X}_L \oplus P_1 = X_L \oplus X_L = 0 \tag{1}$$

When the round function is applied the first time, the right half becomes:

$$\mathrm{X}_R \oplus F(0) \tag{2}$$

When P_2 ist added to the right half modulo 2 (XORed), it becomes:

$$\mathrm{X}_R \oplus F(0) \oplus P_2 = X_R \oplus F(0) \oplus X_R = F(0) \tag{3}$$

When now the round function is applied the second time, the left half becomes:

$$0 \oplus F(F(0)) = F(F(0)) \tag{4}$$

Thus when the third round of the second iteration starts, the right and left halves of data are constants. Since P_1 and P_2, which initially contained the first 64 bits of the user key XORed with some digits of π, have been replaced after the first iteration, the output of the second iteration, which will form P_3 and P_4, does not depend on P_1 and P_2 and hence not on the first 64 bits of the userkey, if the userkey is of maximal length. If the original userkey is in length 384 bits or less, then the first 64 bits of the userkey reappear in the higher round keys, which means that P_3 and P_4 do depend on the first 64 bits of the

original userkey, since they are XORed with the higher initial round keys at a later stage.

4 Concluding Remarks

In this article the authors have shown that for the block cipher Blowfish, the subkeys of the third and fourth round do not depend on the first 64 bits of the userkey. A possible remedy would be to change the order of assignment of the $P_i, i = 1 \ldots 18$ in the key schedule. Despite the fact that the twist was obviously unintended, we do not believe that it will facilitate an attack on Blowfish, because it affects only two out of the 18 subkeys.

5 Acknowledgement

The author is grateful to Claus Grupen of Siegen University for encouragement and support and to Juha Erkkila for pointing to a flaw in the first version of this paper. Special thanks go to Johanna and Robert Schmidt.

References

[1] Schneier, Bruce: Description of a New Variable-Length Key, 64-bit Block Cipher (Blowfish), in: Anderson, Ross (Ed.): *Fast Software Encryption, Cambrigde Security Workshop, Proceedings,* Springer-Verlag, Berlin, 1994

[2] Schneier, Bruce: *Angewandte Kryptographie*, Addison-Wesley (Deutschland), Bonn, 1996, German Translation of *Applied Cryptography*