Cryptanalysis of XXTEA

Elias Yarrkov*

May 4, 2010

Abstract

XXTEA, or Corrected Block TEA, is a simple block cipher in Roger Needham and David Wheeler's TEA series of algorithms. We describe a chosen plaintext attack for XXTEA using about 2^{59} queries and negligible work.

1 Introduction

XXTEA, or Corrected Block TEA,[3] is a block cipher proposed by Roger Needham and David Wheeler after an attack on the original Block TEA.[2] Some more recent algorithms, notably EnRUPT,[1] use a structure very similar to that of XXTEA. Although flexible and fast in software, XXTEA has seen little cryptanalysis.

2 XXTEA description

XXTEA is a Feistel network operating on a block consisting of at least two 32-bit words, using a 128-bit key. The block can be viewed as a circular array. A single XXTEA full cycle consists of looping through the block words, adding to each word a function of its immediate neighbors, full cycle number and the key; a single XXTEA round for a fixed block length can be concisely described as $v_r \leftarrow v_r + F(v_{r-1}, v_{r+1}, r, k)$. A full cycle is n rounds, where n is the number of words in the block. The number of full cycles to perform over the block is given as $6 + 52/n$.

*yarrkov@gmail.com

Algorithm 1 XXTEA encryption implementation

```
void xxtea_full_cycle(uint32_t *v, int n, uint32_t *key, int cycle)
{
    uint32_t sum, z, y, e;
    int r;
    sum = cycle*0x9e3779b9;
    e = sum>>2;
    for(r=0; r<n; r++)
    {
        z = v[(r+n-1)%n]; // the left neighboring word
        y = v[(r+1)%n];   // the right neighboring word
        v[r] += ((z>>5^y<<2)+(y>>3^z<<4))^((sum^y)+(key[(r^e)%4]^z));
    }
}
void xxtea(uint32_t *v, unsigned int n, uint32_t *k)
{
    int i, cycles = 6+52/n;
    for(i=1; i<=cycles; i++)
        xxtea_full_cycle(v, n, k, i);
}
```

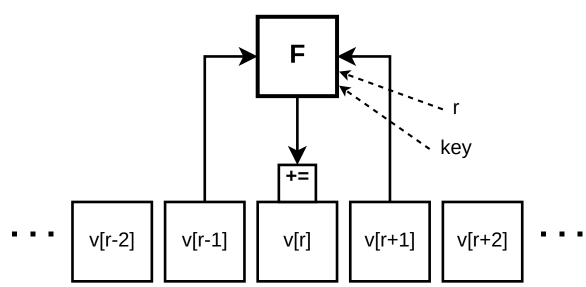

Figure 1: XXTEA structure

3 Attacks

The number of full cycles XXTEA performs over the block is reduced to only 6 when the block consists of at least 53 words. This property presents several attractive possible approaches for a cryptanalyst. Differential cryptanalysis will be used, and difference is considered subtraction per word.

3.1 Approach 1

We try to find a difference Δ so that $F(v_{r-1}, v_{r+1}+\Delta, ...) = F(v_{r-1}, v_{r+1}, ...)$, and two rounds later $F(v_{r-1} + \Delta, v_{r+1}, ...) = F(v_{r-1}, v_{r+1}, ...)$, with some probability. When we encrypt two blocks with such a difference Δ in a single word, the difference will remain in just that word with some probability.

XXTEA's F is not bijective for either neighbor, so such collisions are possible. The probability of $\Delta = 13$ passing each of the 5 first full cycles was experimentally measured, giving a combined 5-cycle probability between 2^{-109} and 2^{-100} for most keys. Passing 5 of 6 full cycles is considered a right pair, and detecting it is easy, as blocks will collide in most words when Δ is placed near the end of the block. The attack, including key recovery, was implemented and verified for XXTEA reduced to 3 full cycles. The precise probability of the characteristic is key-dependent.

3.2 Approach 2

We try to find a difference Δ so that $F(v_{r-1}, v_{r+1}+\Delta, ...)-F(v_{r-1}, v_{r+1}, ...) = \Delta$, and on the next round $F(v_{r-1}+\Delta, v_{r+1}, ...)-F(v_{r-1}, v_{r+1}, ...) = -\Delta$, with some probability. When we encrypt two blocks with such a difference in a single word, then with some probability, the difference will stay in a single word, but move to its left neighboring word on each full cycle. We place Δ in the second last word of a 53-word block to allow it enough room, though the exact location isn't important.

XXTEA is vulnerable to this as well. $\Delta = 1$ gives a 5-cycle passing probability greater than 2^{-69}, estimated by experimentally testing the probability for each full cycle. However, there is a large number of other possible valid differential trails we may end up hitting; a single-word difference Δ_0 will turn into the single-word difference Δ_1 when $F(v_{r-1}, v_{r+1}+\Delta_0, ...)-F(v_{r-1}, v_{r+1}, ...) = \Delta_1$, and on the next round $F(v_{r-1} + \Delta_1, v_{r+1}, ...) - F(v_{r-1}, v_{r+1}, ...) = -\Delta_0$. We can use a somewhat more complex method to get a better estimate of the probability of a right pair. The most effective single-word differences seem to be near zero; we take into account differences no further than d steps from zero.

We first experimentally measure, with 2^{20} tests, the probabilities of $F(a, b+\Delta_0, r, k) - F(a, b, r, k) = \Delta_1$ and $F(a + \Delta_1, b, r, k) - F(a, b, r, k) = -\Delta_0$ for each Δ_0 and Δ_1 in $[-d, d]$, each r that will encounter the differences, fixed k, and random a and b. From these, we calculate the probability for each cycle that both of the conditions hold, for given Δ_0 and Δ_1 in $[-d, d]$. Now, we can quickly calculate the probability of any one appropriately behaving differential trail, when the difference of the only differing word stays in $[-d, d]$. For example, the trail $1 \to 1 \to 1 \to 1 \to 1$ has a probability above 2^{-69} for most keys. Starting from an initial difference 1, we calculate the sum of all possible subsequent trails. Several optimizations to this method come from noticing that the differences x and $-x$ are equivalent. Running the method for several thousand keys, with $d = 32$, yields a median right pair probability near $2^{-56.6}$, with the majority of or all keys above 2^{-58}. A higher d doesn't seem to increase the expected probability significantly. While the method doesn't give an exact probability, it is reasonable and provides consistent output. Thus, about 2^{59} chosen plaintext queries is expected to be enough.

Key recovery is quite straightforward. When we get a right pair, we know all input in the last use of F, except the used key word, and we also know the output difference. With this information, it's easy to find the possible candidate subkeys for the last round and proceed backwards to recover more. The additional effort is negligible.

The attack was succesfully used against XXTEA reduced to 4 of 6 full cycles. A right pair was found after about 2^{35} chosen plaintext query pairs, about $2^{34.7}$ being the expected number.

4 Conclusions

Attacks for XXTEA were presented that show it does not provide the intended 128-bit security. The used approaches may be useful against similar current and future designs.

References

[1] Sean O'Neil. Enrupt: First all-in-one symmetric cryptographic primitive. The State of the Art of Stream Ciphers (SASC), 2008.

[2] Markku-Juhani Saarinen. Cryptanalysis of block tea, unpublished manuscript, 1998.

[3] D J Wheeler and R J Needham. Correction of xtea. unpublished manuscript. In *Computer Laboratory, Cambridge University*, 1998.

A Right pair example

Approach 2 right pair example for 4 of 6 full cycles.

```
Key: 1234BABE 56756756 55555555 DEADBEEF

Input:

B[0] =
D31BC730 909DEF3A 493A0BF8 ADAC5BB7 B141376D B97239ED A6B8BB9D 2484FA40
7A2E2EB7 87542F1C 8A057153 B72D5A1B 8F01DBD8 5104CBA7 036BE40E 24FCFCE9
9BA5DF15 50879A78 F05697F4 C190B59F A1FBBA85 5CA0084D F1F01E15 A2208492
1F7C5C5B C24033B7 01FAFFEC 4E76A72E 7085CB18 CD543B48 DD846F0E FFF7E8C2
499EC7AF 3CBB8582 46743280 D1FF26C0 21A0BD62 B5309C3A 52032679 98AC9ECA
7B74BD9A 27C2FFA4 B89D22F2 3CF2ACDE 8803A892 B6DB90B4 B39E4040 081CF23A
B536C14D 157AECD5 1C82835D 2814F84D 861A04ED

B[1]-B[0] =
```

```
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000001 00000000
```

```
After cycle 1:

B[0] =
2B1E4F6D A353E5C3 7A6F54AB D9D1ED76 904038B8 E1BAFECC C0D535B2 AEF215D5
7D27C16B DFD96AAB BCBEC7EE A719F7FE 0C6D6F56 F59AB5FA 575656EA F9877936
3F608191 4912A479 28461663 6A9B8E15 B8634B2E E938568F 1299BE17 040E5D46
EB9938C8 31A63946 C1CB8F64 9F26CD1B 712CB436 5974C4D2 064C298F B3D19BDC
028D6A31 CA746C08 1D959D17 3AA3C5E0 2CF087D8 65534C18 2BBD1431 5AEF63B0
09BA151D BAF15777 8DC0C290 3D7A237D 92E62F36 31E730E7 AC30D4BA 8A3E6E34
4EF1EF7F 639952B3 14BA3938 56E746A5 2758E6E7

B[1]-B[0] =
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000007 00000000 00000000
```

```
After cycle 2:

B[0] =
660E5F76 CF9DC778 3F156ACB 83835829 59B18827 69D6C84A 4EEE5AA8 B896F21B
009489CA A6468E16 4429DCD6 633CA213 447E6508 C75F9B25 745A84AD E7E7F99C
30FD62EB 1AE11A52 139BDF69 C92319EF 3A33E988 5F5120A0 5CCC5803 286EF8F5
EC2F0FF0 9E8D69AD DE4D6959 82376D6D F031AD54 1C58A40A 81A46DB5 698F08FE
E01B05A3 75A4B361 7F943918 576F8262 B9C0EC28 A88E1182 C2A73B6C 788C6D90
807C5ECA 89BB0E4D E1889FF2 F841F309 4863DF12 1A3008B2 8C0B5910 5693FE63
A57236BA 1A45109D C4FF21C3 2ABD1E64 F9EF78EE

B[1]-B[0] =
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 FFFFFFFF 00000000 00000000 00000000
```

```
After cycle 3:

B[0] =
```

```
6757A3AE BD485043 1793F96F 9FFE3C37 42136DA8 BD68D1B2 07CBB920 0D7BAF68
8A805AE9 EE101EB0 1181D193 280170E8 72F3E444 7749817F 1A8EC7F3 2D2B5B01
3510A612 329CF2B0 75CA541D 0C34A07E D4CFF349 8B827E88 4C39D5AD E88FB242
06C89F96 CBA179B5 69425F80 A6B5F970 58266376 9387429C 64DD2CF1 2208249E
03F441DC 232E6598 A9EC2337 6628482C 76FCCB43 E91FA636 DBD7626C D561C1DF
349FA34D 81EBC8C8 3202F5CD B6C1CAD5 C15FBD0F 666D7613 511A47DD 719D3E24
060A39B9 954E92F4 3C8823CD 5B1C4A77 8A7DCB45

B[1]-B[0] =
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
FFFFFFFD 00000000 00000000 00000000 00000000
```

```
After cycle 4:

B[0] =
67CDB2C2 FC264373 57AE6481 E05B9240 C16562A4 3B25ECD9 BAD389A1 1B1FBF9F
0B6AAB68 665CF534 1F4D1214 B9848F91 03019F21 662CFA14 2C0D9F07 89E42D2B
84AD0C70 E7619F0D 7707EDC0 81407B6F BABF3B02 964B0D8C 454E8D92 CB983981
9F81BFAC D41F1510 81C81F85 303BCA70 88E6D65E 4080B59D D44965EC D4221949
120C2132 2F7949DC 2FD2FD6F ACFF27CA 76A33665 C2AFB179 262EFEDF 5DD3A747
0B33C90B A0155A23 EB4301DF 909A31CB 086018AF FABE1909 9B4206E7 0ED85139
18441B11 CDC39681 B2DDE794 0DF7ACA9 92CD9E55

B[1]-B[0] =
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 0000001C
00001920 FFFDA2B9 004B4BA4 D67B1949 2AACDD45
```