Differential Cryptanalysis of the BSPN Block Cipher Structure

Liam Keliher
Mathematics & Computer Science
Mount Allison University
Sackville, New Brunswick

NIST Lightweight Cryptography Workshop, July 21, 2015

CANADA

Mount Allison University

Sackville, NB

SAC 2015 + S3

This year, Mount Allison University is hosting:

- ***SAC 2015*** : 22nd International Conference on Selected Areas in Cryptography
  - August 12-14, 2015
- ***SAC Summer School (S3)***
  - August 10-12, 2015

Outline

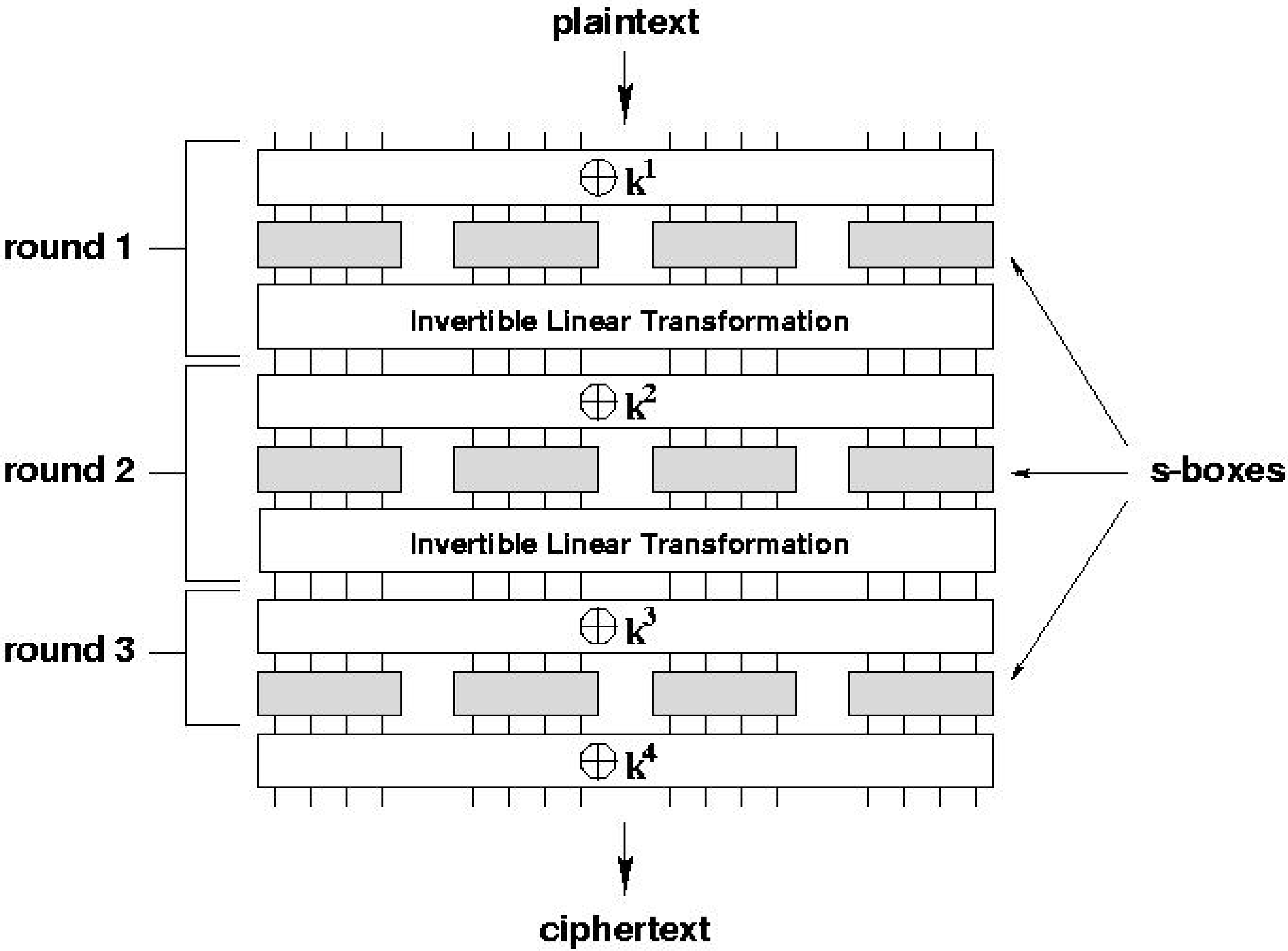
- Substitution-Permutation Networks (SPNs)
- BSPN
- BSPN Linear Transformation Properties
- High Probability Differentials for BSPN
- Conclusion

SPN Round Structure

- SPN: standard block cipher structure (e.g., AES)

  Let n = block size

- Round stages:
  1. XOR n-bit subkey
  2. apply $m \times m$ s-boxes (substitution boxes)
     - invertible mappings from $\{0,1\}^m$ to $\{0,1\}^m$
  3. apply linear transformation (traditionally a bitwise permutation)


plaintext
round 1
round 2
round 3
⊕ k1
Invertible Linear Transformation
⊕ k2
Invertible Linear Transformation
⊕ k3
⊕ k4
s-boxes
ciphertext

Independent Subkeys

- We assume the most general situation for the subkeys, namely: k^1, k^2, ... are chosen independently and uniformly from $\{0,1\}^n$

- This is a standard assumption that facilitates analysis
  - Expected values over cipher keys often approximated by expected values over independent subkeys

BSPN

- *BSPN* (byte-oriented SPN) is an SPN structure presented at SAC 1996 by Youssef, Tavares, and Heys
- It was designed as a more efficient version of the *bit-oriented* SPN structure published earlier in 1996 in J. Cryptology by Heys and Tavares
- BSPN is meant to be *involutional* (self-inverting)
  - has influenced other involutional ciphers such as Khazad and CURUPIRA

BSPN Structure

- Many BSPN parameters/components are left unspecified
  - only the linear transformation is given exactly
- A BSPN block consists of B bytes (so $n = 8B$), where B is <u>*even*</u> (e.g., $B = 8$, $B = 16$)
- Key schedule not proposed
  - we assume independent subkeys anyway
- S-boxes not given (involutional recommended)

BSPN-*n*

- Let BSPN-*n* denote BSPN with block size *n*
- We focus on:
  - *BSPN-128* (*B* = 16) (AES-like block size)
  - *BSPN-64* (*B* = 8) (lightweight cipher block size)

BSPN Linear Transformation

- Let $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_B]$ be an input to the BSPN linear transformation, and let $\mathbf{y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_B]$ be the corresponding output

- Then for each $j \in \{1, 2, \ldots, \boldsymbol{B}\}$

$$\mathbf{y}_j = \bigoplus_{1 \le i \le \boldsymbol{B},\, i \ne j} \mathbf{x}_i$$

- This is involutional

BSPN Linear Transformation

- Alternatively, **y = xM**

$$\mathbf{x} = [x_1, x_2, \ldots, x_B]$$
$$\mathbf{y} = [y_1, y_2, \ldots, y_B]$$

$$\mathbf{M} = \begin{bmatrix} 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 0 & 1 & \cdots & 1 \\ 1 & 1 & 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & 1 & \cdots & 0 \end{bmatrix}$$

only invertible if *m* is even

BSPN Linear Transformation

- Efficient computation of BSPN LT:

$$x = [x_1, x_2, \ldots, x_B]$$
$$y = [y_1, y_2, \ldots, y_B]$$

If $Q = \bigoplus_{1 \le i \le B} \mathbf{x}_i$

then $y_i = Q \Leftarrow x_i$ for each i

- BSPN-64 has been considered as a lightweight block cipher (see, e.g., Zhang et al.)

BSPN LT Weaknesses

- The BSPN LT has two main properties that make it vulnerable to attack:

1. large number of fixed points

2. low diffusion

Fixed Points

- A *fixed point* is an input **x** for which

$$LT(\mathbf{x}) = \mathbf{x}$$

- BSPN has a fixed point $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_B]$ whenever

$$\mathbf{Q} = \bigoplus_{1 \le i \le B} \mathbf{x}_i = \mathbf{0}$$

- So BSPN has $2^{8(B-1)} = 2^{n-8}$ fixed points

Fixed Points

- In particular, any input with two identical nonzero bytes is a fixed point, e.g.,

$$\mathbf{x} = [\mathbf{w}, \mathbf{w}, 0, 0, \ldots, 0] \qquad \mathbf{w} \neq 0$$

- We exploit fixed points of this form

Differential Probability (DP)

Let $F : \{0,1\}^N \rightarrow \{0,1\}^N$. Fix $\boldsymbol{a}, \boldsymbol{b} \in \{0,1\}^N$

$$\boxed{DP(\boldsymbol{a}, \boldsymbol{b}) = \mathrm{Prob}_X\{\, F(X) \Leftarrow F(X \Leftarrow \boldsymbol{a}) = \boldsymbol{b} \,\}}$$

- For our purposes, F may be:
  - an s-box
  - a single SPN round
  - multiple consecutive SPN rounds
- If F is parameterized by key material, the expected DP value is denoted $EDP(\boldsymbol{a}, \boldsymbol{b})$

Differential Cryptanalysis (DC)

- Chosen-plaintext attack that exploits differences *a* and *b* with relatively large EDP values over *T* core rounds (e.g., $T = R - 2$)

- Data complexity (# chosen plaintexts required) is given by

$$\frac{C}{EDP(a,b)}$$

where *C* is a small constant

Differential Characteristics

- A ***differential characteristic (trail)*** is a vector

$$\Omega = \left\langle a^1, a^2, \ldots, a^T, a^{T+1} \right\rangle$$

- a^t / a^{t+1} are input/output differences for round t
- gives input/output differences for each s-box
- product of resulting s-box DP values is the ***expected differential characteristic probability***, denoted $EDCP(\Omega)$

Common Approach

- *Usual approximation:* Find

$$\Omega = \left\langle a^1, a^2, \ldots, a^T, a^{T+1} \right\rangle$$

whose *EDCP* is maximal (***best characteristic***)
(there are efficient algorithms for this)

- Set $\boldsymbol{a} = \boldsymbol{a}^1$ and $\boldsymbol{b} = \boldsymbol{a}^{T+1}$ and assume

$$EDP(a, b) \approx EDCP(\Omega)$$

Differentials

- However, Lai et al. (1991) showed that the value *EDP* (*a*, *b*) is actually a sum of EDCP terms over a (large) set of characteristics

$$EDP(a,b) = \sum_{\Omega = \left\langle a, a^2, \ldots, a^T, b \right\rangle} EDCP(\Omega)$$

- This set is called a ***differential***
- To assess the vulnerability to DC, we need to compute differential EDP values

High Prob. BSPN Differentials

- For BSPN, the highest prob. characteristics consist entirely of differences of the form we considered earlier:

$$[\mathbf{w}, \mathbf{w}, 0, 0, \ldots, 0] \qquad \mathbf{w} \neq 0$$

(any two fixed byte positions can be used)

- **We designed a (simple) algorithm to add up the ELDP values of all characteristics of this form over any number of core BSPN rounds**

S-Box Choice

- In keeping with the BSPN designers' recommendation, we chose the strongest involutional s-boxes we could find
- Sometimes called *Nyberg s-boxes*, these are based on inversion in the finite field $GF(2^8)$

$$0 \leftarrow 0$$

$$x \leftarrow x^{-1} \qquad x \neq 0$$

- The AES s-box is derived from this formula

Best BSPN Characteristics

- For a Nyberg s-box in $GF(2^8)$, the maximum nontrivial LP value is 2^{-6}
- This means that the highest possible ELCP value over T rounds for our characteristics (2 active s-boxes per round) is

$$2^{-12T}$$

- Implies: DC of BSPN-64 impossible for $T > 5$

  DC of BSPN-128 impossible for $T > 10$

Results

- Our algorithm produced the following EDP values as a function of *T* (#core rounds)

T	EDP
2	$2^{-20.8}$
3	$2^{-28.9}$
4	$2^{-35.9}$
5	$2^{-42.9}$
6	$2^{-49.9}$
7	$2^{-56.8}$
8	$2^{-63.8}$
9	$2^{-70.8}$
10	$2^{-77.8}$
…	…
15	$2^{-112.7}$
16	$2^{-119.6}$
17	$2^{-126.6}$
18	$2^{-133.6}$

Concluding Analysis

- Since our ELP value for $T = 7$ is $2^{-56.8}$, we can attack (say) 8 or 9 rounds of BSPN-64 with a data complexity around 2^{59}
- And since our ELP value for $T = 16$ is $2^{-119.6}$, we can attack 17 or 18 rounds of BSPN-128 with a data complexity around 2^{122}

Low Diffusion

- The *branch number* of a byte-oriented linear transformation is the minimum number of nonzero bytes over all input/output pairs:

$$B = \min \{ wt_8(\mathbf{x}) + wt_8(\mathbf{y}) : \mathbf{y} = LT(\mathbf{x}),\ x \neq 0 \}$$

where $wt_8(\ )$ = byte-oriented Hamming weight (number of nonzero bytes)

$$\boxed{2 \leq B \leq m+1}$$

Low Diffusion

$$\boxed{2 \le B \le m+1}$$

- The branch number quantifies the ability of the linear transformation to spread (diffuse) the influence of the input bytes over the output bytes (or vice versa)
- A high branch number is desirable
- However, the BSPN LT branch number is 4 (independent of *m*)

Low Diffusion

branch number of BSPN LT = 4

- Use our "special" fixed points:

  $\mathbf{x} = [\mathbf{w}, \mathbf{w}, 0, 0, \ldots, 0] \qquad \mathbf{w} \neq 0$

  $y = LT(\mathbf{x}) = \mathbf{x}$

  $wt_8(x) + wt_8(y) = 4$

OTOH

- If $wt_8(x) = 1$, then $wt_8(\mathbf{y}) = m$
- If $wt_8(x) = 3$, then $wt_8(\mathbf{y}) \geq 3$