# Shabal

E. Bresson
C. Clavier
T. Icart
P. Paillier
C. Thuillet
A. Canteaut
T. Fuhr
J.-F. Misarsky
T. Pornin
M. Videau
B. Chevallier-Mames
A. Gouget
M. Naya-Plasencia
J.-R. Reinhard

Cryptolog, DCSSI, EADS, France Télécom, Gemalto, INRIA, Sagem Sécurité

initiated by the Saphir project

The first SHA-3 NIST Conference, February 27, 2009

# Main characteristics of Shabal

## Parameters.

- Internal state: 44 words (1408 bits).
- Message blocks: 16 words (512 bits).

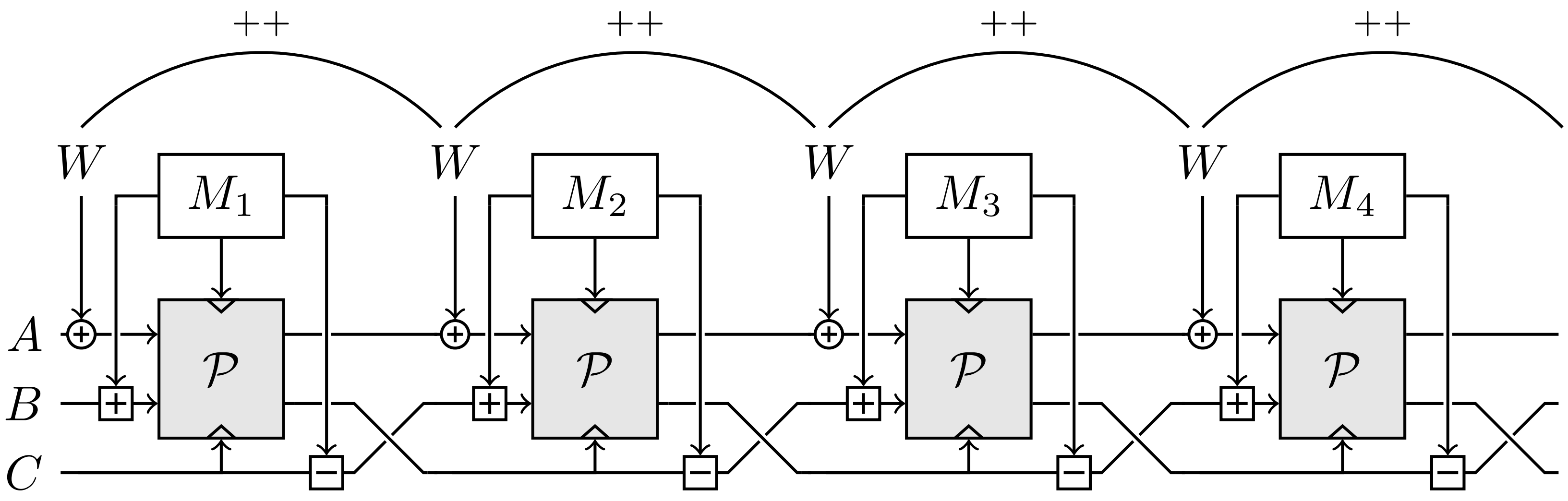
## Generic construction.

- Message rounds: iterate a keyed permutation with respect to a provably secure mode of operation;
- Final rounds: 3 slightly different additional rounds;
- Output: $\ell_h$ bits from the internal state;
- Keyed permutation: operates on a $28$-word input, parameterized by two $16$-word values.

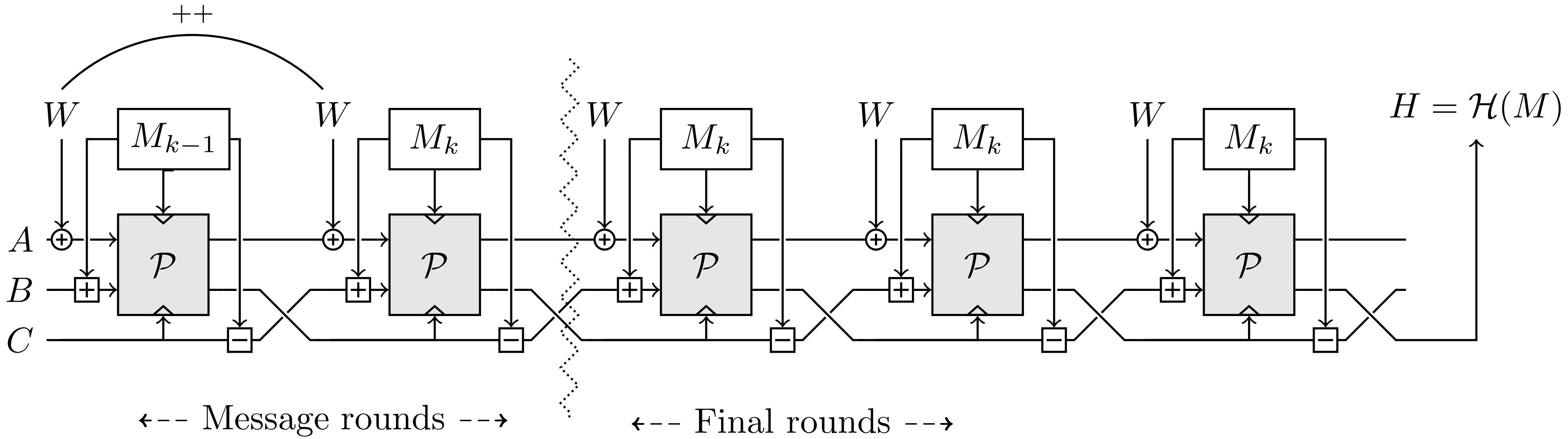
# Message rounds

$|\boldsymbol{A}| = \mathbf{12}$ words and $|\boldsymbol{B}| = |\boldsymbol{C}| = \mathbf{16}$ words.

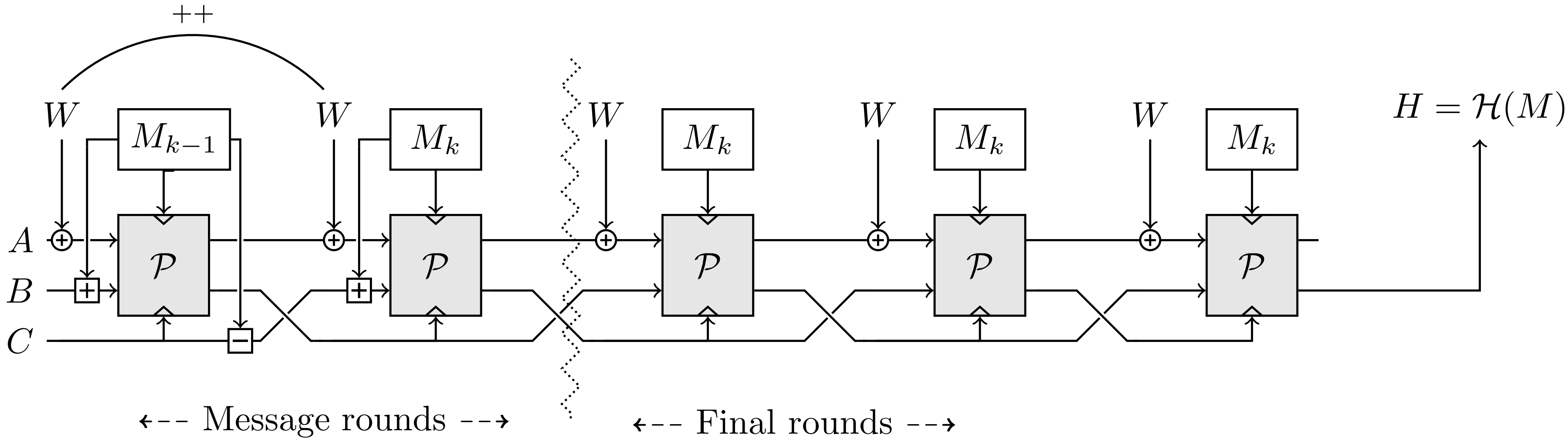
$\boldsymbol{W}$: $\mathbf{2}$-word counter.



# Final rounds



# Final rounds: equivalent view



# Padding and initialization

## Padding.

The message is post-padded with a 1 followed by as many 0 as required so that the length is a multiple of 512 bits.

## Initialization.

- Prefix approach: the message is prefixed with two 512-bit blocks

$$(\ell_h, \ldots, \ell_h + 15), (\ell_h + 16, \ldots, \ell_h + 31)$$

where $\ell_h$ is the output length.

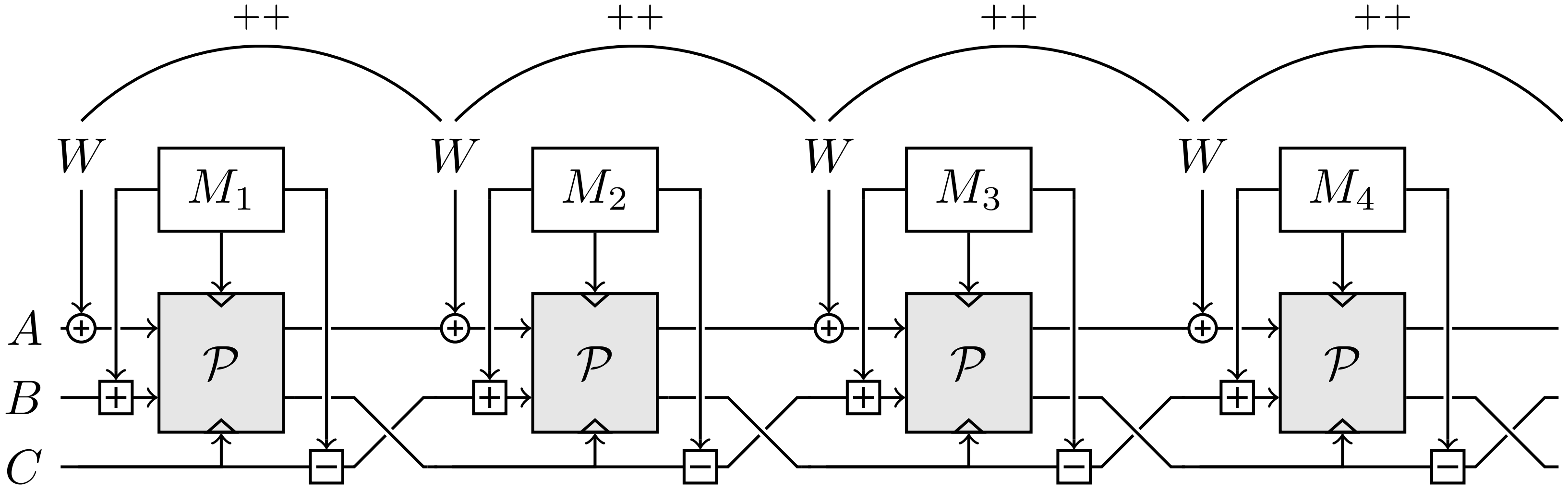
$$\text{internal state } \leftarrow \mathbf{0}, \text{ counter } \leftarrow -1.$$

- IV approach:

$$\text{internal state } \leftarrow IV_{\ell_h}, \text{ counter } \leftarrow 1.$$

# Shabal generic operating mode

$|\boldsymbol{A}| = \boldsymbol{\ell_a}$ bits and $|\boldsymbol{B}| = |\boldsymbol{C}| = |\boldsymbol{M_i}| = \boldsymbol{\ell_m}$ bits.



## A provably secure operating mode

If the keyed permutation $\mathcal{P}$ is viewed as a random keyed permutation, we can prove:

**Indifferentiability from a random oracle.**

- Shabal behaves like a random oracle up to

$$\mathbf{2}^{\frac{\ell_a+\ell_m}{2}} = \mathbf{2}^{448}$$

  evaluations of $\mathcal{P}$ or $\mathcal{P}^{-1}$.

- Internal collisions require no less than $\mathbf{2}^{448}$ evaluations of $(\mathcal{P}, \mathcal{P}^{-1})$;

- Shabal is collision resistant when the collision finder is bounded to $\mathbf{2}^{\ell_h/2}$ evaluations of $(\mathcal{P}, \mathcal{P}^{-1})$.

# A provably secure operating mode (2)

## (Second)-preimage resistance.

- Shabal is preimage resistant when the preimage finder is limited to

$$\min\left(2^{\ell_h}, 2^{\ell_a+\ell_m-\log(\ell_m+1)-2}\right) = \min\left(2^{\ell_h}, 2^{885}\right) = 2^{\ell_h}$$

  evaluations of $(\mathcal{P}, \mathcal{P}^{-1})$.

- Shabal is second preimage-resistant for $\kappa$-bit messages up to

$$\min\left(2^{\ell_h}, 2^{\ell_a+\ell_m-\log k^*}\right) = \min\left(2^{\ell_h}, 2^{903-\log \kappa}\right)$$

  evaluations of $(\mathcal{P}, \mathcal{P}^{-1})$ where $k^* = \lceil(\kappa+1)/\ell_m\rceil$.

# Sébastien Chabal

http://commons.wikimedia.org/wiki/File:Sebastien-Chabal_large.jpg

THE Sun

Tuesday, February 10, 2009



HOME | MY Sun | MOBILE | SUN LITE | SITE MAP | CONTACT US | RSS | BOOKMARK US

THE SUN

SPORT
Football
Transfer News
Dream Team
Sports Videos
Columnists
F1 & Motorsport
Wrestling
UFC
Boxing
Cricket
Rugby Union
Rugby League
Tennis
Golf
Sport USA
Racing
Top 10s
Sporting Snaps
Sports Babes
Bet
Poker
Casino
Planet Sport
Alert Me
VIDEO
NEWS
Forces
Royals
Sun Justice
Sun Money
more
SHOWBIZ
Bizarre
Bizarre USA
Film
Music
more

RUGBY UNION

EXCLUSIVE

# French monster EATS babies!

BAST THE BEAST ... French powerhouse Sebastien Chabal

By PHIL THOMAS

Published: 11 Oct 2007

ADD YOUR COMMENTS

LOCK up your children! This is the picture which proves just what England will be up against when they come face to face with

RELATED STORIES

Robinson is after a nifty fifty

JASON ROBINSON wants to make his 50th England game

http://www.thesun.co.uk/sol/homepage/sport/rugby_union/article326083.ece

## Chabal eats Gröstl for breakfast

## The keyed permutation

**Input:** $M, A, B, C$ **Output:** $A, B$

**for** $i$ from $0$ to $15$ **do**

$B[i] \leftarrow B[i] \lll 17$

**end for**

**for** $j$ from $0$ to $2$ **do**

**for** $i$ from $0$ to $15$ **do**

$$\begin{aligned} A[i + 16j \bmod 12] \leftarrow\ & \mathcal{U}\big(A[i + 16j \bmod 12] \oplus C[8 - i \bmod 16] \\ & \oplus \mathcal{V}(A[i - 1 + 16j \bmod 12] \lll 15)\big) \\ & \oplus M[i] \oplus B[i + 13 \bmod 16] \\ & \oplus (B[i + 9 \bmod 16] \wedge \overline{B[i + 6 \bmod 16]}) \end{aligned}$$

$B[i] \leftarrow (B[i] \lll 1) \oplus \overline{A[i + 16j \bmod 12]}$

**end for**

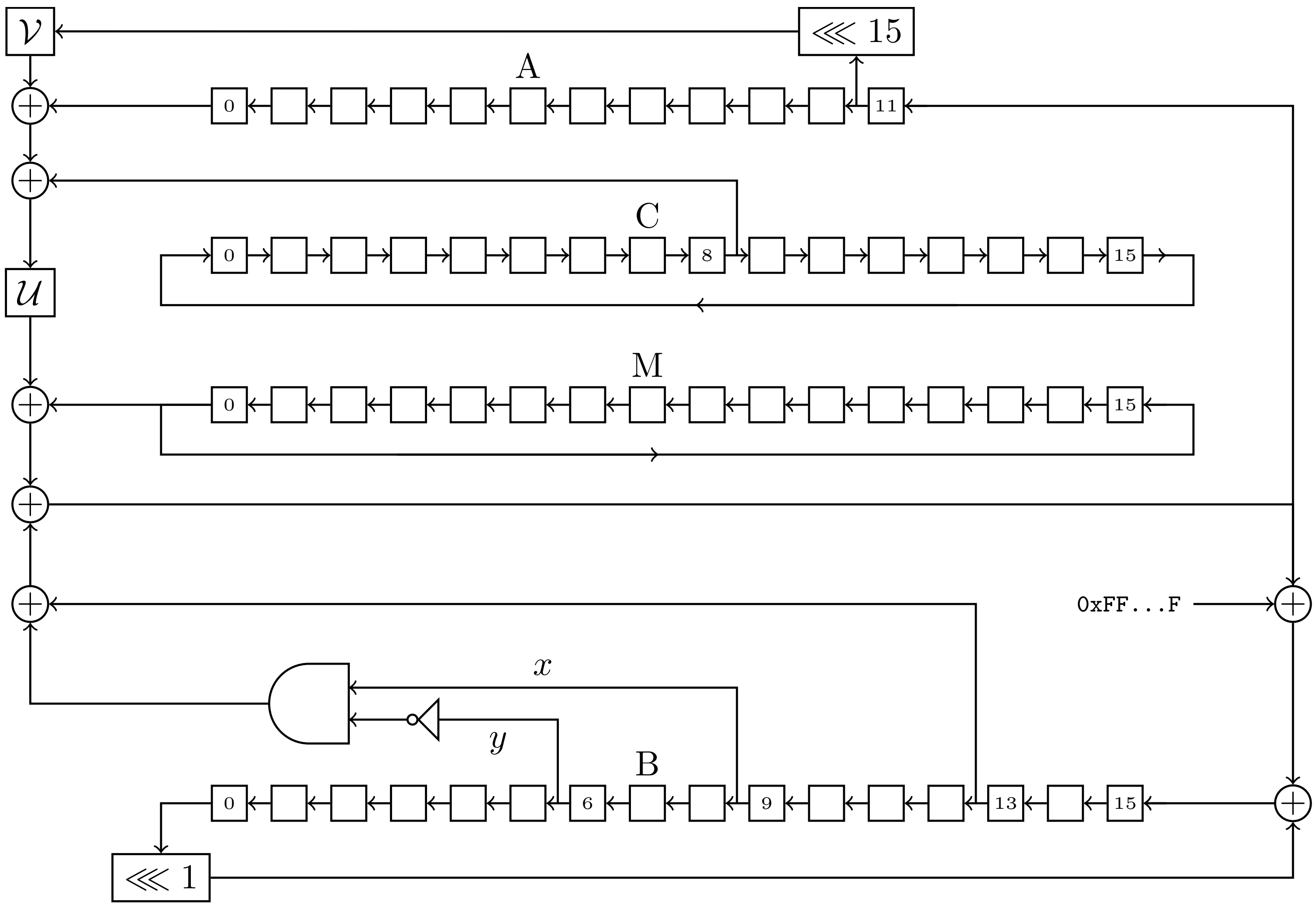
**end for**

**for** $j$ from $0$ to $35$ **do**

$A[j \bmod 12] \leftarrow A[j \bmod 12] + C[j + 3 \bmod 16]$

**end for**

# The keyed permutation (without the final update of $A$)

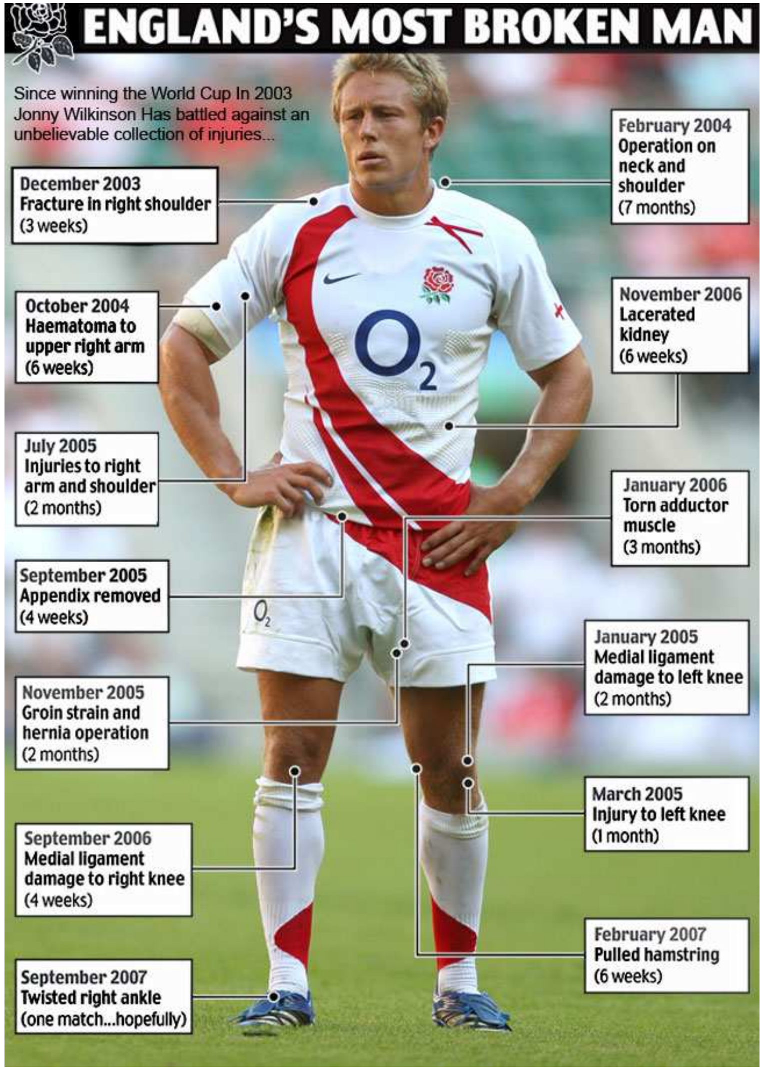


## Choice of the nonlinear permutations

$$\mathcal{U} : x \mapsto 3 \times x \bmod 2^{32}$$
$$\mathcal{V} : x \mapsto 5 \times x \bmod 2^{32}$$

- they avoid the use of look-up tables;
- they can be easily hard-coded (one bit shift and one addition);
- they cannot transform a symmetric difference (the all-one word) into a symmetric difference;
- one difference in the message block causes at least one difference between the inputs of $\mathcal{U}$ or of $\mathcal{V}$ after two rounds.

## Weakened versions of Shabal: Weakinson-xxx




http://www.dailymail.co.uk/sport/rugbyunion/article-480057/

Fast, efficient, with good statistics, but often broken.

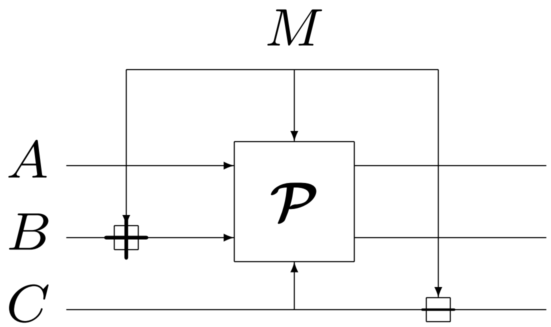
## Security analysis of Weakinson

### Distinguishers for the keyed permutation? [Aumasson09].

Distinguish $\mathcal{P}$ from some queries $\mathcal{P}_{M,C}(A,B)$ for fixed unknown values of $A, B, C$ and for different chosen values of $M$.

- distinguisher for $\mathcal{P}$ from $2^{12}$ queries [Aumasson09];
- distinguisher for $\mathcal{P}^{-1}$ from $2$ queries [Shabal, Section 11.6].

### Can such distinguishers be used?




- For Shabal: **no**;
- For Weakinson with $2$ loops instead of $3$ and without the final update of $A$ in $\mathcal{P}$: preimage attack with $2^{512}$ calls to $\mathcal{P}$ [Shabal, Section 11.6].

## Security claims

For any $\ell_h \in \{192, 224, 256, 384, 512\}$.

### Collision resistance.

Finding a collision for Shabal-$\ell_h$ requires at least $2^{\ell_h/2}$ calls to the message round function.

### Preimage resistance.

Any preimage attack against Shabal-$\ell_h$ requires at least $2^{\ell_h}$ calls to the message round function.

### Second-preimage resistance.

Any second-preimage attack against Shabal-$\ell_h$ for messages shorter than $2^k$ bits requires at least $2^{\ell_h - k}$ calls to the message round function.

### Resistance to length-extension attacks.

Any length-extension attack against Shabal-$\ell_h$ requires at least $2^{256}$ calls to the message round function.

## Cycles/byte: AMD 64 Intel Core 2 Quad [eBASH]

| | long | 4096 bytes | 576 bytes |
|---|---|---|---|
| Edon-R-512 | 3.06 | 3.20 | 3.75 |
| Blue Midnight Wish-512 | 5.26 | 5.45 | 6.28 |
| Skein-512 | 6.71 | 6.89 | 8.00 |
| SHA-1 | 7.50 | 7.89 | 10.22 |
| **Shabal-512** | **8.03** | **8.56** | **11.72** |
| BLAKE-64 | 10.06 | 10.53 | 12.08 |
| Keccak[r=1024,c=576] | 10.45 | 10.90 | 12.39 |
| SIMD-256 | 11.50 | 11.79 | 13.47 |
| CubeHash 8/16 | 13.46 | 14.65 | 21.84 |
| SHA-512 | 14.17 | 14.83 | 17.36 |
| Grøstl-512 | 30.09 | 31.63 | 37.83 |
| MD6-512 | 52.60 | 40.61 | 102.14 |
| SHAvite-3-512 | 111.50 | 115.03 | 124.78 |
| LANE-512 | 139.97 | 148.46 | 219.31 |
| CubeHash 8/1 | 213.01 | 214.19 | 221.39 |

## Cycles/byte: x86 Intel Core 2 Duo [eBASH]

| | long | 4096 bytes | 576 bytes |
|---|---|---|---|
| Edon-R-256 | 8.10 | 8.30 | 9.50 |
| Blue Midnight Wish-256 | 9.86 | 10.11 | 11.50 |
| **Shabal-512** | **10.22** | **10.90** | **15.04** |
| CubeHash 8/16 | 12.70 | 13.92 | 21.32 |
| SIMD-256 | 13.46 | 13.80 | 15.93 |
| BLAKE-32 | 20.15 | 20.59 | 23.18 |
| Grøstl-256 | 22.73 | 23.38 | 27.33 |
| SHA-256 | 22.98 | 23.47 | 26.43 |
| LANE-256 | 26.27 | 27.17 | 32.62 |
| SHAvite-3-256 | 29.38 | 30.04 | 33.76 |
| Keccak[r=1024,c=576] | 31.52 | 32.67 | 35.40 |
| Skein-512 | 38.89 | 39.79 | 45.01 |
| MD6-224 | 84.95 | 79.20 | 157.10 |
| SHA-512 | 115.27 | 119.30 | 131.15 |
| CubeHash 8/1 | 202.52 | 204.02 | 213.18 |

# Smartcard platforms

**32-bit processor.**

code size: 2 kBytes

RAM: 300 bytes

for 256-byte messages: 195 cycles/byte (IV approach)

($2\times$ slower than SHA-1)

**8-bit 8051 smartcard.**

code size: 1.2 kBytes

RAM: 192 bytes

for 256-byte messages: 2930 cycles/byte (IV approach)

($2.5\times$ slower than SHA-1)

**8-bit smartcard with arithmetic coprocessor.**

code size: 1.2 kBytes

RAM: 256 bytes

for 256-byte messages: 625 cycles/byte (IV approach)

($3\times$ slower than SHA-1)

# Conclusions

http://www.flickr.com/photos/sam_herd/2620280308/

- fast, simple and efficient;
- based on a provably secure operating mode;
- important security margins;
- very few instructions requested;
- no S-box;
- fast on many different platforms.