The SPEED Cipher *

Yuliang Zheng

School of Computing, Monash University
McMahons Road, Frankston, Melbourne, VIC 3199, Australia
Email: yzheng@fcit.monash.edu.au

Abstract. SPEED is a private key block cipher. It supports three variable parameters: (1) data length — the length of a plaintext/ciphertext of SPEED can be 64, 128 or 256 bits. (2) key length — the length of an encryption/decryption key of SPEED can be any integer between 48 and 256 (inclusive) and divisible by 16. (3) rounds — the number of rounds involved in encryption/decryption can be any integer divisible by 4 but not smaller than 32.

SPEED is compact, which is indicated by the fact that the object code of a straightforward implementation of SPEED in the programming language C occupies less than 3 kilo-bytes. It makes full use of current, and more importantly, emerging CPU architectures which host a large number of high-speed hardware registers directly available to application programs. Another important feature of SPEED is that it is built on recent research results on highly nonlinear cryptographic functions, as well as other counter-measures against differential and linear cryptanalytic attacks.

It is hoped that the compactness, high throughput and adjustable parameters offered by SPEED, together with the fact that the cipher is in the public domain, would make it an attractive alternative cipher for security applications including electronic financial transactions.

1 Design Philosophy

The aim of this paper is to introduce a private key cipher that is suitable for software implementation and takes the maximum advantage of emerging computer architectures that host an increasing number of fast internal hardware registers directly available to application programs. The cipher is called SPEED which stands for a Secure Package for Encrypting Electronic Data.

Cryptographic strength of SPEED is built on recent research results on constructing highly nonlinear Boolean functions [15, 16]. Operation efficiency is an important factor that has been taken into account in the process of design. Another design goal is to provide the cipher with applicability to fast one-way hashing and efficient generation of cryptographically strong pseudo-random numbers. Encryption and pseudo-random number generation have direct applications in

* The source code of SPEED implemented in the programming language C is located at the following URL: http://pscit-www.fcit.monash.edu.au/~yuliang/
providing data confidentiality, whereas one-way hashing is essential for efficient authentication and digital signature.

While most smart cards use 8-bit CPUs, workstations and personal computers are mainly based on 32-bit CPUs which support fast processing of 8, 16 and 32-bit data. Similarly, emerging 64-bit CPUs support efficient handling of 8, 16, 32 and 64-bit data. This results in our decision for the basic data unit for the encryption/decryption operation of SPEED to be a 8-bit, 16-bit or 32-bit word. As a plaintext/ciphertext of SPEED consists of 8 words, choosing a 8-bit word as the basic data unit results in a block cipher on 64-bit data, a 16-bit word results in a block cipher on 128-bit data, and a 32-bit word results in a block cipher on 256-bit data. The process of SPEED is composed of 4 passes, each involving 8 or more consecutive rounds. Thus similarly to RC5 [14], SPEED supports three variable parameters, namely data length, key length and the number of rounds. Relevant ideas on variable parameters were previously used in a one-way hashing algorithm called HAVAL [18].

A bit-wise nonlinear Boolean operation is employed in each round. To strengthen the cipher against the differential attack proposed by Biham and Shamir [1], a data-dependent cyclic shift is applied on the output of the operation. This technique was inspired by RC5. The use of a maximally nonlinear Boolean function in a bit-wise Boolean operation would help thwart the linear attack discovered by Matsui [9].

The remainder of this paper is organized as follows: Section 2 details the specification of SPEED, Section 3 provides background information on the round transform used in SPEED, and Section 4 discusses the construction and properties of the five nonlinear Boolean functions used in SPEED. A preliminary analysis of the strength of the cipher against cryptanalysis is reported in Section 5, while a comparison of SPEED with other ciphers in terms of its throughput (the number of bits encrypted/decrypted per unit of time) is provided in Section 6. Finally applications of SPEED in one-way hashing and pseudo-random number generation are suggested in Sections 7 and 8.

2 Description of SPEED

First we introduce a few terms used in this paper. As a common practice, a byte is composed of 8 bits. As we mentioned earlier, by a word we mean a string of 8, 16 or 32 bits. All bits in a byte or a word are indexed, starting with 0, from right to left hand side. It is convenient to call right hand side bits lower bits, while left hand side bits upper bits. Three types of operations are applied to data. The first is bit-wise Boolean operations, the second is cyclic shifts (i.e., rotation) to the right or left, and the third is modular additions.

In the following discussions we use $w$ to indicate the length of (i.e, the number of bits in) a plaintext/ciphertext, $\ell$ the length of a key, and $r$ the number of rounds. $w$ can be chosen to be 64, 128 or 256, $\ell$ an integer between 48 and 256 (inclusive) and divisible by 16, and $r$ an integer larger than or equal to 32 and divisible by 4. SPEED with parameters $w$, $\ell$ and $r$ may be denoted by $(w, \ell, r)-$