Information for each entry item is restricted to the designated pages. However, the applicant may decide how much page space to assign for any individual entry item.

Receipt Number	

Cryptographic Techniques Overview

1. Name of Cryptographic Technique: *Camellia*

Categories
1.Asymmetric Cryptographic Schemes
[✓ 2.Symmetric Ciphers]
3.Hash Functions
4.Pseudo-random Number Generators

Security Functions of Asymmetric Cryptographic Schemes
1.confidentiality 2. authentication 3. signature 4. key- sharing

Subcategories of Symmetric Ciphers
1. stream ciphers 2. 64-bits block ciphers [✓ 3. 128-bits block ciphers]

2. Cryptographic Techniques Overview

2.1 Design policy

(1) Design rationale:

(a) Interface and components:

- Block size is 128-bits, and key lengths are 128-, 192-, and 256-bits.
- Camellia consists of s-boxes and logical operations, but no arithmetic operations.

(b) Design of round function:

- Following design rationale of E2's P-Function for designing a linear transformation layer (P-function).
- Using an inverse function over GF (2^8) for designing s-boxes.
- Producing four series of s-boxes by using different affine transformations.

(c) Design of FL- and FL^{-1}-functions:

- Following design rationale of MISTY's FL-function.

(d) Design of key scheduling function:

- Enabling to generate subkeys on-the-fly.
- Shortening time for subkey generation to less than that for one block encryption.
- Enabling to construct the key scheduling function for 128-bit key as a part of the function for 192- and 256-bit keys.

(2) Security assessments:

(a) Camellia is designed to provide sufficient invulnerability to differential cryptanalysis, linear cryptanalysis, and truncated differential attack.

(b) Camellia has been confirmed to be sufficiently invulnerable to higher order differential attack, interpolation attack, related-key attack, impossible differential attack, slide attack, and so on.

(c) Camellia has no equivalent key.

(3) Implementation:

(a) Camellia can implement the round function efficiently according to applicable circumstances.

- 64-bit CPU, 32-bit CPU, high-end smart card, and low-end smart card for software
- Small size implementation and high-speed implementation for hardware.

(b) Camellia can provide efficiency at least comparable to that of the AES finalists in software implementation.

(c) Camellia can occupy small RAM and ROM in software implementation.

(d) Camellia can implement an encryption circuit with smallest size among all existing 128-bit block ciphers as far as we know.

2.2 Intended applications

Camellia is applicable to any circumstance in which symmetric block ciphers are applied. In particular, it fits secret communication and authentication.

In addition, according to applicable circumstances, Camellia can be implemented efficiently by using implementation techniques suitable for 32-bit CPU, 64-bit CPU, high-end smart card, low-end smart card, and hardware.

Information for each entry item is restricted to the designated pages. However, the applicant may decide how much page space to assign for any individual entry item.

Receipt Number	

2.3 Basic theory and techniques

(1) The theory and technique of designing Camellia are based on those used in designing E2 (ref. [5]) and MISTY (ref. [6]):

- Design of P-function:
  The design of the P-function follows that of E2's round function, i.e. only XOR operations are used. It provides the best security against differential and linear cryptanalyses (ref. [4]).
- Design of FL- and FL^{-1}-functions:
  The technique of designing these functions is expected to improve security against differential cryptanalysis, linear cryptanalysis and other attacks (including unknown attacks) without a large impact on efficiency. It follows that of MISTY's FL-function (ref. [6]).

(2) Security assessments:

- Security against differential and linear cryptanalyses is assessed from the upper bounds of the maximum differential and linear characteristic probabilities (ref. [3]).
- The search algorithm is used to assess security against truncated differential attack (ref. [8][9]).

(3) Implementation Techniques:

- Implementation is based on the inversion function over GF(2^8) which uses subfield GF(2^4) (ref. [7]).
- Round function implementation differs with the target machine to increase the ease and efficiency of implementation (ref. [2]).

References of submission

[1] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, T. Tokita, "Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms," SAC2000, LNCS to appear.

[2] K. Aoki, H. Ueda, "Optimized Software Implementations of E2", IEICE Trans., Vol.E83-A, No.1, 2000.

[3] M. Kanda, "Practical Security Evaluation against Differential and Linear Attacks for Feistel ciphers with SPN Round Function," SAC2000, LNCS to appear.

[4] M. Kanda, Y. Takashima, T. Matsumoto, K. Aoki, K. Ohta, "A Strategy for Constructing Fast Round Functions with Practical Security Against Differential and Linear Cryptanalysis," SAC'98, LNCS 1556.

[5] NTT corporation, "E2: Efficient Encryption algorithm," http://info.isl.ntt.co.jp/e2/, (Summary version appears in IEICE Trans., Vol.E83-A, No.1, 2000)

[6] M. Matsui, "New Block Encryption Algorithm MISTY," FSE'97, LNCS 1267.

[7] M. Matsui, T. Inoue, A. Yamagishi, H. Yoshida, "A note on calculation circuits over GF(2^{2n})," IEICE Technical Report IT88-14, (in Japanese)

[8] M. Matsui, T. Tokita, "Cryptanalysis of a Reduced Version of the Block Cipher E2," FSE'99, LNCS 1636.

[9] S. Moriai, M. Sugita, K. Aoki, M. Kanda, "Security of E2 against Truncated Differential Cryptanalysis," SAC'99, LNCS 1758.

IEICE Trans.: IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science
FSE: Fast Software Encryption – Annual International Workshop
LNCS: Springer — Lecture Notes in Computer Science series
SAC: Annual Workshop on Selected Areas in Cryptography

Previous use: None