KEMENTERIAN SAINS, TEKNOLOGI DAN INOVASI
MINISTRY OF SCIENCE, TECHNOLOGY AND INNOVATION

CyberSecurity MALAYSIA
An agency under MOSTI

Projek MySEAL:

Kriteria Penyerahan dan Penilaian

Versi 2.0 [2018]

Name of author: KUMPULAN FOKUS MySEAL

File name: CD-5-RPT-0218-Kriteria MySEAL Versi 2.0

Date of document: 08 April 2018

Document classification : Public

For inquiry about this document please contact:
Hazlin Abdul Rani
Head, Cryptography Development Department
hazlin@cybersecurity.my

For general inquiry about us or our services,
please email: info@cybersecurity.my

Prakata

Dokumen ini disediakan bagi memberi maklumat kepada khalayak umum berhubung kriteria penyerahan dan penilaian bagi pemilihan algoritma yang akan disenaraikan dalam Senarai Algoritma Kriptografi Terpercaya Negara (MySEAL) / *National Trusted Cryptographic Algorithm List*. MySEAL akan digunakan sebagai rujukan keperluan dan garis panduan bagi penggunaan algoritma kriptografi dalam semua produk kriptografi di Malaysia.

Dokumen ini mengandungi 8 bahagian. Bahagian 1 mengandungi pengenalan kepada MySEAL. Bahagian 2, Bahagian 3 dan Bahagian 4 menerangkan tentang keperluan umum, kriteria penyerahan dan kriteria penilaian bagi setiap primitif. Bahagian 5 mengandungi keperluan pelesenan bagi setiap penyerahan manakala Bahagian 6 memaparkan garis masa bagi projek ini. Bahagian 7 menerangkan tentang keperluan formal bagi penyerahan, manakala Bahagian 8 mengandungi maklumat umum yang lain berkenaan projek MySEAL.

Dokumen ini telah dibangunkan oleh ahli Kumpulan Fokus MySEAL yang terdiri daripada pakar kriptografi negara daripada universiti awam, universiti swasta dan agensi kerajaan. Sebarang pertanyaan berhubung dokumen ini boleh diajukan kepada myseal.fg@cybersecurity.my.

Penghargaan

Ribuan terima kasih diucapkan kepada organisasi / institusi berikut di atas penghasilan dokumen ini (mengikut urutan abjad): -

a. CyberSecurity Malaysia
b. MIMOS Berhad
c. Universiti Kebangsaan Malaysia
d. Universiti Malaya
e. Universiti Malaysia Sabah
f. Universiti Multimedia
g. Universiti Pertahanan Nasional Malaysia
h. Universiti Putra Malaysia
i. Universiti Sains Islam Malaysia
j. Universiti Sains Malaysia
k. Universiti Teknikal Malaysia Melaka
l. Universiti Teknologi MARA
m. Universiti Tenaga Nasional
n. Universiti Tunku Abdul Rahman

Kandungan

1.0 Pengenalan kepada MySEAL

Senarai Algoritma Kriptografi Terpercaya Negara (MySEAL) ialah projek untuk membangunkan portfolio algoritma kriptografi terpercaya negara. Projek ini direka khusus untuk menyediakan senarai algoritma kriptografi yang sesuai bagi pelaksanaan dalam konteks Malaysia yang menyokong Dasar Kriptografi Negara (*National Cryptography Policy*, NCP). NCP yang merupakan dokumen panduan bagi Malaysia untuk mencapai kedaulatan dalam bidang kriptografi akan disokong oleh MySEAL dalam aspek kriptografi dan kripanalisis..

Usaha ini bertujuan untuk menggalakkan kerjasama strategik, penyelidikan dan penghasilan sistem kriptografi oleh industri tempatan yang mana industri berkenaan dapat menyerahkan algoritma kriptografi mereka untuk diuji dan disahkan oleh pemeriksa sebelum algoritma tersebut dapat diakui sebagai algoritma kriptografi terpercaya di peringkat negara. Algoritma kriptografi yang akan disenaraikan dalam MySEAL perlu mematuhi kriteria seperti yang dinyatakan dalam dokumen ini. Kriteria tersebut telah dibangunkan berdasarkan standard antarabangsa yang telah diterima dan keperluan yang ditentukan oleh jawatankuasa Kumpulan Fokus MySEAL. Jawatankuasa ini diterajui oleh CyberSecurity Malaysia dan disokong oleh ahli yang terdiri daripada institusi di Malaysia.

Pusingan pertama projek MySEAL hanya menerima primitif *symmetric (block cipher* dan *stream cipher)* , *asymmetric* dan *cryptographic hash function* sahaja . Selepas dipertimbangkan selanjutnya oleh ahli Kumpulan Fokus MySEAL, dua lagi primitif kriptografi; primitif *cryptographic key generation* dan primitif *cryptographic pseudo random number generator* perlu dimasukkan kedalam projek MySEAL. Algoritma bagi setiap primitif akan diperoleh melalui dua cara; panggilan penyerahan algoritma baharu dan algoritma daripada standard yang sedia ada termasuk projek penyenaraian algoritma kriptografi lain yang telah dipilih oleh jawatankuasa yang dilantik. Semua algoritma akan disemak terlebih dahulu dan kemudian dinilai dengan terperinci berdasarkan kriteria penilaian. Akhir sekali, algoritma yang terpilih akan diumumkan.

Inisiatif MySEAL bukanlah satu perkara yang kecil. Pelaksanaan MySEAL merupakan satu detik yang cukup penting bagi Malaysia dan selaras dengan pendokumenan Agenda IT Negara (NITA) pada tahun 1996 yang menyenaraikan e-Kedaulatan sebagai salah satu objektif Malaysia apabila memasuki era Teknologi Maklumat (IT). Inisiatif ini telah membuka jalan untuk Malaysia agar dapat menceburi arena asas keselamatan maklumat. Arena yang mencabar ini akan menjadi bukti kepada kegigihan dan

keupayaan Malaysia dalam melindungi prasarana maklumat negara pada peringkat algoritma kriptografi.

Di samping menyediakan cabaran dan aspirasi kepada ahli kriptografi, projek ini juga bertujuan untuk memupuk bakat baharu dan mengekalkan bakat yang sedia ada. Sehubungan itu, inisiatif MySEAL telah memberikan peluang keemasan kepada Malaysia untuk menyediakan platform kerjasama sesama entiti kerajaan, industri dan institusi pengajian tinggi bagi mempromosi dan menggalakkan peserta agar membangunkan algoritma kriptografi baharu serta pada masa yang sama melahirkan lebih ramai ahli kriptografi. Seterusnya, projek MySEAL akan membawa Malaysia lebih hampir untuk merealisasikan cabaran keenam Wawasan 2020, yakni *mewujudkan masyarakat saintifik dan progresif, masyarakat yang mempunyai daya perubahan tinggi dan berpandangan ke depan, yang bukan sahaja menjadi pengguna teknologitetapi juga penyumbang kepada tamadun sains dan teknologi masa depan.*

2.0 Keperluan MySEAL

Bahagian ini menerangkan tentang keperluan MySEAL; kategori primitif kriptografi yang dipertimbangkan dalam projek MySEAL, kelayakan peserta, kelayakan algoritma kriptografi yang akan diserahkan dan kriteria pemilihan umum projek yang diperlukan bagi semua primitif. Keselamatan, kecekapan dan kefleksibelan primitif kriptografi akan dibincangkan dengan lebih lanjut di bawah subbahagian kriteria pemilihan umum.

2.1 Kategori Primitif Kriptografi

MySEAL mengalu-alukan penyerahan primitif kriptografi yang kukuh dalam kategori yang dinyatakan di bawah:

a) Primitif *Block Cipher*
b) Primitif *Stream Cipher*
c) Primitif *Asymmetric Cryptographic Primitive*
d) Primitif *Cryptographic Hash Function*
e) Primitif *Cryptographic Key Generation*
f) Primitif *Cryptographic Pseudo Random Number Generator*

2.2 Kelayakan Peserta

Penyerahan dibuka kepada pencipta dan/atau pemilik algoritma.

2.3 Kelayakan Algoritma Kriptografi

Sebarang algoritma kriptografi yang tidak disenaraikan dalam mana-mana standard sedia ada dan projek penyenaraian algoritma kriptografi yang lain.

2.4 Kriteria Pemilihan Umum

Kriteria pemilihan utama melibatkan keselamatan, kecekapan dan kefleksibelan.

1) Keselamatan

Penilaian ke atas tahap keselamatan kriptografi asas melalui kripanalisis adalah mandatori.

2) Kecekapan

Sesuatu algoritma kriptografi hendaklah dapat dilaksanakan pada perkakasan dan/atau perisian dengan cekap.

3) Kefleksibelan

Sesuatu algoritma kriptografi adalah sebaik-baiknya sesuai untuk digunakan dalam pelbagai persekitaran.

3.0 Kriteria Penyerahan bagi Setiap Primitif

Bahagian ini menyediakan maklumat tentang kriteria penyerahan bagi setiap primitif seperti yang dinyatakan dalam subbahagian yang berikut.

3.1 Primitif Block Cipher

Primitif *block cipher* dibahagikan kepada dua kategori; *block cipher* serba guna dan *lightweight block cipher*. Kriteria penyerahan adalah seperti berikut:

1) Block Cipher

a. Panjang kekunci sekurang-kurangnya 128 bit.
b. Panjang blok sekurang-kurangnya 128 bit.
c. Laporan analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:
    i. Ujian statistik NIST
    ii. *Linear cryptanalysis*
    iii. *Differential cryptanalysis*
d. Laporan pelaksanaan dan prestasi:
    i. Perisian yang disasarkan dan/atau
    ii. Perkakasan yang disasarkan
e. Pewajaran bagi prinsip reka bentuk algoritma. Lihat **Lampiran A** sebagai contoh.
f. Vektor ujian
    - Bilangan kunci: - sekurang-kurangnya 3 untuk setiap saiz kunci
    - Bilangan pasangan *plaintext-ciphertext*: - 3 untuk setiap kunci
    - Sampel pemprosesan mestilah dalam *ECB mode* dengan *padding* bit '0'
    - Output pertengahan bagi setiap pusingan

2) Lightweight Block Cipher

a. Panjang kekunci sekurang-kurangnya 80 bit.
b. Panjang blok sekurang-kurangnya 64 bit.
c. Laporan analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:
    i. Ujian statistik NIST

ii. *Linear cryptanalysis*

iii. *Differential cryptanalysis*

d. Laporan pelaksanaan dan prestasi:

i. Perisian yang disasarkan dan/atau

a) Saiz kod program

b) Saiz RAM

ii. Perkakasan yang disasarkan

a) Keluasan cip

b) Kitaran

c) Bit bagi setiap kitaran

d) Kuasa

e) Tenaga

e. Pewajaran bagi prinsip reka bentuk algoritma. Lihat **Lampiran B** sebagai contoh.

f. Vektor ujian

- Bilangan kunci: - sekurang-kurangnya 3 untuk setiap saiz kunci
- Bilangan pasangan *plaintext-ciphertext*: - 3 untuk setiap kunci
- Sampel pemprosesan mestilah dalam *ECB mode* dengan *padding* bit '0'
- Output pertengahan bagi setiap pusingan

3.2 Primitif Stream Cipher

Kriteria penyerahan adalah seperti berikut:

a. Pengendalian: *Synchronous/self-synchronous stream cipher*

b. Perkakasan

i. Panjang kekunci sekurang-kurangnya 80 bit.

ii. Ingatan dalaman sekurang-kurangnya 160 bit.

c. Perisian

i. Panjang kekunci sekurang-kurangnya 128 bit.

ii. Ingatan dalaman sekurang-kurangnya 256 bit.

d. Laporan analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:

i. Ujian statistik NIST

ii. *Algebraic attack*

iii. *Correlation attack*

iv. *Distinguishing attack*

v. *Guess-and-Determine attack*

e. Laporan pelaksanaan dan prestasi:
   i. Perisian yang disasarkan dan/atau
   ii. Perkakasan yang disasarkan

f. Pewajaran bagi prinsip reka bentuk algoritma. Lihat **Lampiran C** sebagai contoh.

g. Vektor ujian
   - Bilangan kunci: - sekurang-kurangnya 3 untuk setiap saiz kunci
   - Bilangan Initialization Vectors: - 3 untuk setiap kunci
   - Panjang *keystream*: - 256 bits
   - *Internal state* selepas menjana 256 bit *keystream*

3.3 Primitif Asymmetric Cryptographic

Kriteria penyerahan adalah seperti berikut:

a. Skema:
   i. Penyulitan (*Encryption*)
   ii. Perjanjian kekunci (*Key agreement*)
   iii. Tandatangan digital (*Digital signature*)

b. Bukti ketepatan

c. Analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:
   i. Masalah dan andaian matematik yang sukar
   ii. Panjang kekunci minimum yang diperlukan untuk mencapai tahap keselamatan[1] 2^{128}
   iii. Model keselamatan berserta dengan bukti[2]

d. Laporan pelaksanaan dan prestasi:
   i. Perisian yang disasarkan, dan/atau
   ii. Perkakasan yang disasarkan

e. Pewajaran bagi prinsip reka bentuk algoritma.

f. Vektor ujian
   - Bilangan pasangan kunci: - sekurang-kurangnya 3 pasangan kunci
   - Bilangan sampel pemprosesan untuk setiap pasangan kunci: - 2 sampel

[1] Tahap keselamatan merujuk kepada bilangan langkah bagi serangan yang paling diketahui terhadap primitif kriptografi

[2] Teknik yang diterima termasuklah teknik pengurangan, *peralihan serangan (game-hopping)* atau *kebolehgubahan universal (universal composability)*

3.4 Primitif Cryptographic Hash Function

Primitif *cryptographic hash function* dibahagikan kepada dua kategori; *cryptographic hash function* serba guna dan *lightweight cryptographic hash function*. Kriteria penyerahan adalah seperti berikut:

1) Cryptographic Hash Function

a. Saiz cerna sebanyak 224 bit, 256 bit, 384 bit, 512 bit atau lebih besar
b. Panjang mesej maksimum sebanyak 2^{64}-1 bit
c. Laporan analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:
    i. *Pre-image resistance*
    ii. *Second pre-image resistance*
    iii. *Collision resistance*
d. Laporan pelaksanaan dan prestasi:
    i. Perisian yang disasarkan, dan/atau
    ii. Perkakasan yang disasarkan
e. Pewajaran bagi prinsip reka bentuk algoritma. Lihat **Lampiran D** sebagai contoh.
f. Vektor ujian
    - Bilangan sampel untuk setiap saiz data: - 3 sampel
    - *Intermediate state* bagi setiap pusingan

2) Lightweight Cryptographic Hash Function

a. Saiz cerna sebanyak 80 bit, 128 bit dan 160 bit.
b. Panjang mesej maksimum sebanyak 2^{64}-1 bit.
c. Laporan analisis keselamatan hendaklah mengandungi, tetapi tidak terhad kepada:
    i. *Pre-image resistance*
    ii. *Second pre-image resistance*
    iii. *Collision resistance*
d. Laporan pelaksanaan dan prestasi:
    i. Perisian yang disasarkan dan/atau
        a) Saiz kod program
        b) Saiz RAM

ii. Perkakasan yang disasarkan

a) Keluasan cip

b) Kitaran

c) Bit bagi setiap kitaran

d) Kuasa

e) Tenaga

e. Pewajaran bagi prinsip reka bentuk algoritma. Lihat **Lampiran E** sebagai contoh.

f. Vektor ujian

- Bilangan sampel untuk setiap saiz data: - 3 sampel
- *Intermediate state* bagi setiap pusingan

3.5 Primitif *Cryptographic Key Generation*

Kriteria penyerahan adalah seperti berikut:

a. Skop:

i. Penjana Nombor Perdana (*Prime Number Generator*)

b. Laporan analisis hendaklah mengandungi, tetapi tidak terhad kepada:

i. *Probabilistic Prime Generators*

a) Bukti sekurang-kurangnya 75% tepat (setanding ujian *Miller Rabin Primality*)

b) *Primality test* menghasilkan output “input adalah nombor perdana”, “input adalah nombor komposit” atau “ujian tidak muktamad”

c) Berjalan dalam masa polinomial

d) Vektor ujian

- Saiz nombor perdana: - 512, 1024, 2048, 3072, 4096, 7680, 15360 bit
- Bilangan *seed* bagi setiap saiz nombor perdana: - 3 *seed* (128, 256, 512 bit)

ii. *Deterministic Prime Generators*

a) Bukti ketepatan

b) Berjalan dalam masa polinomial

iii. Dapat membezakan nombor *Carmichael* dari nombor perdana

iv. Dapat menghasilkan sampel *pseudo primes* dari penjana

v. Ujian statistik NIST

c. Laporan pelaksanaan dan prestasi:
    i. Perisian yang disasarkan, dan/atau
    ii. Perkakasan yang disasarkan

3.6 Primitif *Cryptographic Pseudo Random Number Generator*

Kriteria penyerahan adalah seperti berikut:

a. Skop:
    i. Penjana Nombor Rawak Pseudo (*Pseudo Random Number Generator* (PRNG))
b. Laporan analisis hendaklah mengandungi, tetapi tidak terhad kepada:
    i. PRNG berdasarkan kepada metodologi *asymmetric*
        a) Bukti ketepatan
        b) Berjalan dalam masa polinomial
        c) Jika *internal state* PRNG mengandungi n bit, tempoh mestilah sekurang-kurangnya 2^n.
        d) Vektor ujian
            - Saiz *seed*: - 128, 256, 512 bit
        e) Menggunakan parameter *asymmetric* yang kuat
            - *Integer Factorization Problem* (IFP):- 1024,2048, 4096 bits
            - *Discrete Logarithm Problem* (DLP):- 1024, 2048, 4096 bit
    ii. PRNG berdasarkan kepada metodologi *symmetric*
        a) Berjalan dalam masa polinomial
        b) Vektor ujian
            - Saiz *seed*: - 128, 256, 512 bit
        c) Menggunakan parameter *symmetric* yang kuat
            - AES
            - TDES
    iii. PRNG tidak berdasarkan kepada metodologi *asymmetric* atau *symmetric*
        a) Pewajaran bagi prinsip reka bentuk algoritma.
        b) Bukti ketepatan
        c) Berjalan dalam masa polinomial
        d) Vektor ujian
            - Saiz *seed*: - 128, 256, 512 bit

iv. Ujian statistik NIST

c. Laporan pelaksanaan dan prestasi:

i. Perisian yang disasarkan, dan/atau

ii. Perkakasan yang disasarkan

4.0 Kriteria Penilaian bagi Setiap Primitif

Bahagian ini menerangkan tentang kriteria penilaian bagi setiap primitif.

4.1 Primitif Block Cipher

Kriteria penilaian adalah seperti berikut:

a. Keselamatan
   i. Mencapai tahap keselamatan yang sebanding dengan saiz kekunci berdasarkan serangan kripanalisis. Sebagai contoh, tahap keselamatan 128-bit untuk kekunci 128-bit.
   ii. Lulus dalam kesemua ujian statistik NIST berdasarkan sembilan kategori data. Lihat **Lampiran F**.

b. Kos dan Prestasi
   i. *Block Cipher*: Kecekapan pengkomputeran dan keperluan ingatan yang sebanding dengan AES.
   ii. *Lightweight Block Cipher*: Ukuran pelaksanaan perkakasan dan/atau perisian yang sebanding dengan PRESENT.

c. Ciri Pelaksanaan
   i. Kefleksibelan algoritma boleh mengandungi, tetapi tidak terhad kepada:
      a) Algoritma boleh menampung saiz kekunci tambahan dan blok tambahan.
      b) Algoritma boleh dilaksanakan dengan selamat dan cekap dalam pelbagai platform dan aplikasi.
      c) Algoritma boleh dikendalikan sebagai *stream cipher*, penjana kod pengesahan mesej (MAC), penjana nombor pseudorawak (PRNG) atau *hash function*.
   ii. Kesesuaian perisian dan perkakasan
      Sesuatu algoritma dilihat sebagai mempunyai kelebihan tambahan jika dapat dilaksanakan dengan cekap dalam perisian dan juga perkakasan.
   iii. Kesederhanaan reka bentuk
      Sesuatu reka bentuk dilihat sebagai mempunyai kelebihan jika kelihatan elegan, ringkas, kemas dan mudah difahami.

d. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

4.2 Primitif Stream Cipher

Kriteria penilaian adalah seperti berikut:

a. Keselamatan
    i. Mencapai tahap keselamatan yang sebanding dengan saiz kekunci berdasarkan serangan kripanalisis. Sebagai contoh, tahap keselamatan 128-bit untuk kekunci 128-bit.
    ii. Lulus semua ujian statistik NIST.

b. Kos dan Prestasi

    Kecekapan pengkomputeran dan keperluan ingatan yang sebanding dengan ChaCha20.

c. Ciri Pelaksanaan
    i. Kefleksibelan algoritma boleh mengandungi, tetapi tidak terhad kepada:
        a) Algoritma boleh menampung saiz kekunci tambahan.
        b) Algoritma boleh dilaksanakan dengan selamat dan cekap dalam pelbagai platform dan aplikasi.
        c) Algoritma boleh dikendalikan sebagai penjana nombor pseudorawak (PRNG).
    ii. Kesesuaian perisian dan perkakasan

        Sesuatu algoritma dilihat sebagai mempunyai kelebihan tambahan jika dapat dilaksanakan dengan cekap dalam perisian dan juga perkakasan.
    iii. Kesederhanaan reka bentuk

        Sesuatu reka bentuk dilihat sebagai mempunyai kelebihan jika kelihatan elegan, ringkas, kemas dan mudah difahami.

d. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

4.3 Primitif Asymmetric Cryptographic

Kriteria penilaian adalah seperti berikut:

a. Keselamatan
    i. Masalah dan andaian matematik yang sukar (*Hard mathematical problems and assumptions*), tetapi tidak terhad kepada:
        a) Konvensional:
            1. Masalah Logaritma Diskrit dan variasi (*Discrete Logarithm Problem and variations*)

2. Masalah Pemfaktoran Integer dan variasi (*Integer Factorization Problem and variations*)

b) Pasca kuantum:

1. Masalah berasaskan Kekisi (*Lattice based Problems*)
2. Masalah berasaskan Kod (*Code based Problems*)

ii. Tahap keselamatan[3] hendaklah sekurang-kurangnya 2^{128}

iii. Ketepatan model keselamatan berserta dengan bukti[4]

b. Kos dan Prestasi

i. Saiz parameter bagi setiap tahap keselamatan:

a) Penyulitan (*Encryption*) – saiz kekunci, saiz teks sifer

b) Perjanjian kekunci (*Key agreement*) – saiz kekunci, bilangan penghantaran, lebar jalur

c) Tandatangan digital (*Digital signature*) – saiz kekunci, saiz tandatangan

ii. Kerumitan pengkomputeran

c. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

d. Bukti ketepatan yang sah.

e. Analisis perbandingan dilihat sebagai satu kelebihan.

4.4 Primitif Cryptographic Hash Function

Kriteria penilaian adalah seperti berikut:

a. Keselamatan

Mencapai tahap keselamatan yang sebanding dengan saiz cerna berdasarkan serangan kripanalisis. Sebagai contoh, tahap keselamatan 112-bit untuk saiz cerna 224-bit.

b. Kos dan Prestasi

i. *Cryptographic Hash Function*:

Kecekapan pengkomputeran dan keperluan ingatan yang sebanding dengan SHA-3.

ii. *Lightweight Cryptographic Hash Function*:

Ukuran pelaksanaan perkakasan dan/atau perisian yang sebanding dengan SPONGENT.

[3] Tahap keselamatan merujuk kepada bilangan langkah bagi serangan yang paling diketahui terhadap algoritma kriptografi

[4] Teknik yang diterima termasuklah teknik pengurangan, *peralihan serangan (game-hopping)* atau *kebolehgubahan universal (universal composability)*

c. Ciri Pelaksanaan

i. Kefleksibelan algoritma boleh mengandungi, tetapi tidak terhad kepada:

a) Algoritma boleh menampung saiz cerna tambahan.

b) Algoritma boleh dilaksanakan dengan selamat dan cekap dalam pelbagai platform dan aplikasi.

ii. Kesederhanaan reka bentuk

Sesuatu reka bentuk dilihat sebagai mempunyai kelebihan jika kelihatan elegan, ringkas, kemas dan mudah difahami.

d. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

4.5 Primitif *Cryptographic Key Generation*

Kriteria penilaian adalah seperti berikut:

a. Keselamatan

i. *Safe pseudo random prime number generator*

ii. *Integer Factorization Problem (IFP) safe*

a) Nombor perdana yang kukuh

b) Pasangan nombor perdana yang kukuh

iii. *Discrete Logarithm Problem (DLP) safe*

c) Nombor perdana yang kukuh

iv. Lulus semua ujian statistik NIST.

b. Kos dan Prestasi

i. Berjalan dalam masa polinomial

c. Ciri Pelaksanaan

i. Kefleksibelan algoritma boleh mengandungi, tetapi tidak terhad kepada:

a) Algoritma boleh menampung saiz blok tambahan.

b) Dapat menjana nombor perdana dalam tempoh yang diberikan.

c) Algoritma boleh dilaksanakan dengan cekap dalam pelbagai platform dan aplikasi.

ii. Kesesuaian perisian dan perkakasan

Sesuatu algoritma dilihat sebagai mempunyai kelebihan tambahan jika dapat dilaksanakan dengan cekap dalam perisian dan juga perkakasan.

iii. Kesederhanaan reka bentuk

Sesuatu reka bentuk dilihat sebagai mempunyai kelebihan jika kelihatan elegan, ringkas, kemas dan mudah difahami.

d. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

e. Analisis pembandingan dengan penjana nombor perdana dilihat sebagai mempunyai kelebihan.

4.6 Primitif *Cryptographic Pseudo Random Number Generator*

Kriteria penilaian adalah seperti berikut:

a. Keselamatan
   i. *Safe pseudo random prime number generator*
   ii. *Asymmetric based safe*
      a) *Integer Factorization Problem* (IFP)
      b) *Discrete Logarithm Problem* (DLP) *safe*
   iii. *Symmetric based safe*
   iv. Laporan analisis keselamatan berkenaan dengan *seed entropy*.
   v. Lulus semua ujian statistik NIST.

b. Kos dan Prestasi
   i. Berjalan dalam masa polinomial

c. Ciri Pelaksanaan
   i. Kefleksibelan algoritma boleh mengandungi, tetapi tidak terhad kepada:
      a) Algoritma boleh menampung saiz blok tambahan.
      b) Algoritma boleh dilaksanakan dengan cekap dalam pelbagai platform dan aplikasi.
   ii. Kesesuaian perisian dan perkakasan

   Sesuatu algoritma dilihat sebagai mempunyai kelebihan tambahan jika dapat dilaksanakan dengan cekap dalam perisian dan juga perkakasan.
   iii. Kesederhanaan reka bentuk

   Sesuatu reka bentuk dilihat sebagai mempunyai kelebihan jika kelihatan elegan, ringkas, kemas dan mudah difahami.

d. Ketepatan pewajaran bagi prinsip reka bentuk algoritma.

f. Analisis pembandingan dengan penjana nombor perdana dilihat sebagai mempunyai kelebihan.

5.0 Keperluan Pelesenan

Bahagian ini membincangkan tentang keperluan pelesenan bagi algoritma yang diserahkan.

1. Sekiranya dipilih oleh MySEAL, algoritma yang diserahkan hendaklah bebas daripada royalti. Jika ini tidak dapat dilakukan, maka akses hendaklah dalam bentuk yang tidak berdiskriminasi (algoritma kriptografi tersebut hendaklah tidak dihadkan atau memihak kepada pengguna tertentu sahaja).
2. Penyerah hendaklah menyatakan kedudukan berhubung hak milik intelektual algoritma yang diserahkan. Penyataan hak milik intelektual adalah seperti yang diterangkan dalam Bahagian 7.0 D. Penyataan ini hendaklah dikemas kini apabila perlu.

6.0 Garis Masa Projek MySEAL

Untuk memudahkan pelaksanaan proses bagi keseluruhan projek ini, jadual waktu bagi projek MySEAL telah disediakan seperti di bawah.

Tahun	Aktiviti
2016	S4 : Panggilan penyerahan algoritma kriptografi
2017	S4 : Tarikh akhir penyerahan algoritma kriptografi bagi primitif *symmetric, asymmetric, cryptographic hash function* dan *cryptographic key generation*.
2018	S1 – S2 : Fasa pertama penilaian S2 : (a) Tarikh akhir penyerahan algoritma kriptografi bagi primitif cryptographic pseudo random number generator. (b) Pengumuman algoritma kriptografi yang telah disenarai pendek bagi fasa pertama S3 – S4 : Fasa kedua penilaian
2019	S1 : (a) Fasa kedua penilaian (b) Pengumuman algoritma kriptografi yang telah disenarai pendek bagi fasa kedua S2 – S4 : Fasa akhir penilaian S4 : Pengumuman akhir MySEAL
2020	Penerbitan dan promosi MySEAL

7.0 Keperluan Penyerahan Formal

Bahagian ini membincangkan tentang keperluan penyerahan formal bagi algoritma yang telah diserah.

1. Projek MySEAL akan mengendalikan semua algoritma kriptografi yang telah diserah termasuk sebarang maklumat, data dan dokumen berkaitan yang diterima secara sulit bagi tujuan seperti yang dinyatakan dalam dokumen ini.
2. Projek ini akan memastikan proses pemilihan dan pengendalian dijalankan secara sulit.
3. Mana-mana pakar luar yang dilantik bagi tujuan yang dinyatakan adalah berkewajipan untuk mengendalikan tugas mereka secara sulit.
4. Projek ini berhak untuk menolak algoritma kriptografi yang diserahkan yang tidak dinyatakan dengan jelas dan difahami dengan mudah atau gagal untuk memenuhi keperluan MySEAL dalam beberapa perkara.
5. Setelah algoritma kriptografi diserahkan, penyerah tidak akan dihubungi sehinggalah algoritma tersebut dipilih, melainkan: -
   a. projek MySEAL memerlukan maklumat lanjut atau dokumen sokongan
   b. penyerah telah membuat pertanyaan, aduan atau untuk menjelaskan perkara yang memerlukan maklumat tambahan

7.1 Pakej Penyerahan Algoritma

Perkara berikut perlu disediakan dengan sebarang penyerahan algoritma kriptografi:

A. Lembaran hadapan yang mengandungi maklumat berikut:

[Sila rujuk **Lampiran G**: Borang Penyerahan Algoritma]

1. Maklumat penyerahan sama ada individu atau organisasi
2. Nama penyerah utama, nombor telefon pejabat, nombor mudah alih, nombor faks, alamat e-mel dan alamat surat-menyurat
3. Nama penyerah tambahan (jika ada)
4. Nama pencipta/pembangun algoritma
5. Nama pemilik algoritma (jika berbeza daripada nama penyerah)
6. Tandatangan penyerah
7. Maklumat organisasi (untuk penyerahan organisasi sahaja)
8. Nama algoritma
9. Jenis algoritma yang diserahkan, tahap keselamatan yang dicadangkan dan

persekitaran yang dicadangkan.

B. Spesifikasi algoritma dan dokumentasi sokongan:

1. Huraian yang lengkap dan jelas bagi algoritma dalam bentuk yang paling sesuai, seperti huraian matematik, huraian teks berserta gambar rajah atau pseudokod. Spesifikasi algoritma menggunakan kod adalah tidak dibenarkan. Vektor ujian hendaklah dalam bentuk perenambelasan. Bagi algoritma tidak simetri, kaedah untuk penjanaan kekunci dan pemilihan parameter hendaklah dinyatakan.
2. Penyataan bahawa tiada kelemahan tersembunyi telah dimasukkan oleh pereka bentuk. Lihat **Lampiran G(D.2)**.
3. Penyataan berhubung dakwaan terhadap sifat keselamatan dan tahap keselamatan yang dijangkakan, bersama-sama dengan analisis terhadap algoritma berhubung serangan kripanalitis standard. Kekunci lemah juga hendaklah dipertimbangkan.
4. Penyataan yang menerangkan tentang kekuatan dan batas keupayaan algoritma.
5. Rasional reka bentuk yang menerangkan tentang pilihan reka bentuk.
6. Penyataan berhubung anggaran kecekapan pengkomputeran dalam perisian. Anggaran diperlukan untuk subpengendalian yang berbeza seperti penyediaan kekunci, penyediaan primitif dan penyulitan/penyahsulitan (selagi mana bersesuaian). Kecekapan hendaklah dianggarkan dalam kiraan kitaran bagi setiap bait dan kitaran bagi setiap blok serta menunjukkan jenis pemproses dan ingatan. Sekiranya prestasi berbeza mengikut saiz input, maka nilai bagi sesetengah saiz yang biasa hendaklah disediakan. Sebagai pilihan, pereka bentuk boleh menyediakan anggaran bagi prestasi dalam perkakasan (keluasan, kelajuan, kiraan get, huraian dalam *Hardware Description Language* - HDL).
7. Huraian tentang teknik asas untuk pelaksana bagi mengelakkan kelemahan pelaksanaan.

C. Pelaksanaan dan nilai ujian:

1. Jumlah vektor ujian yang mencukupi bagi setiap parameter.
2. Pelaksanaan rujukan dalam bahasa pengaturcaraan C. MySEAL akan menentukan set API kriptografi yang dapat digunakan hanya untuk algoritma

simetri dan *hash function*, yang akan disediakan di http://myseal.cybersecurity.my. Semua penyerahan perlu melaksanakan API tersebut supaya sistem ujian akan serasi dengan semua penyerahan.

3. Sebagai pilihan, pelaksanaan yang dioptimumkan bagi sesetengah seni bina, pelaksanaan yang menggunakan JAVA atau bahasa himpunan.

D. Penyataan Hak Milik Intelektual:

Penyataan yang menyatakan kedudukan berhubung Hak Milik Intelektual dan dasar royalti bagi algoritma (jika dipilih). Penyataan ini hendaklah disertakan dengan perjanjian untuk memaklumkan perkembangan semasa kepada projek MySEAL sekiranya perlu. Lihat **Lampiran G(D.3)**.

7.2 Arahan untuk penyerahan

1. Semua penyerahan hendaklah dibuat sama ada dalam Bahasa Melayu atau Bahasa Inggeris.
2. Perkara A, B dan D hendaklah diserahkan dalam bentuk bercetak dan juga elektronik. Borang elektronik hendaklah dalam format baca sahaja.
3. Perkara C hendaklah diserahkan menggunakan borang elektronik dalam format baca sahaja.
4. Penyerahan perlu dibuat menggunakan dua cakera yang berasingan. Cakera yang pertama mengandungi perkara A dan D. Cakera yang kedua (dan cakera yang seterusnya) mengandungi perkara B dan C.
5. Setiap cakera perlu dilabelkan dengan nama penyerah, nama algoritma dan tarikh penyerahan. Setiap cakera perlu mengandungi fail teks berlabel "README" yang menyenaraikan semua fail yang terkandung dalam cakera berserta dengan huraian ringkas tentang kandungan setiap fail. Kedua-dua penyerahan bercetak dan cakera optik hendaklah dibuat menggunakan satu pakej tertutup dan dilabelkan seperti yang diterangkan dalam **Lampiran I**.
6. Penyerahan hendaklah tiba pada atau sebelum 17 November 2017 di alamat seperti berikut:

   Sekretariat Kumpulan Fokus MySEAL
   CyberSecurity Malaysia,
   Level 5, Sapura@Mines,
   No 7, Jalan Tasik,
   The Mines Resort City,
   43300 Seri Kembangan

Selangor Darul Ehsan,
Malaysia.

Perakuan akan dihantar melalui e-mel dalam tempoh 3 hari bekerja selepas penerimaan.

7. Sebarang pertanyaan umum boleh dimajukan ke myseal.fg@cybersecurity.my. Jawapan bagi pertanyaan yang berkaitan akan dipaparkan di http://myseal.cybersecurity.my.

8.0 Maklumat Umum

Maklumat umum berkenaan projek MySEAL boleh didapati di http://myseal.cybersecurity.my.

LAMPIRAN A

Pewajaran bagi Prinsip Reka Bentuk Rijndael

Penyataan berikut diekstrak daripada dokumen yang bertajuk **AES Proposal : Rijndael** untuk menunjukkan contoh penyataan bagi prinsip reka bentuk. Dokumen lengkap boleh didapati di http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf.

Design rationale

The three criteria taken into account in the design of Rijndael are the following:

a. Resistance against all known attacks;

b. Speed and code compactness on a wide range of platforms;

c. Design simplicity.

In most ciphers, the round transformation has the Feistel Structure. In this structure typically part of the bits of the intermediate State are simply transposed unchanged to another position. The round transformation of Rijndael does not have the Feistel structure. Instead, the round transformation is composed of three distinct invertible uniform transformations, called layers. By “uniform”, we mean that every bit of the State is treated in a similar way.

The specific choices for the different layers are for a large part based on the application of the Wide Trail Strategy, a design method to provide resistance against linear and differential cryptanalysis. In the Wide Trail Strategy, every layer has its own function:

The linear mixing layer: guarantees high diffusion over multiple rounds.

The non-linear layer: parallel application of S-boxes that have optimum worst-case nonlinearity properties.

The key addition layer: A simple EXOR of the Round Key to the intermediate State.

Before the first round, a key addition layer is applied. The motivation for this initial key addition is the following. Any layer after the last key addition in the cipher (or before the first in the context of known-plaintext attacks) can be simply peeled off without knowledge of the key and therefore does not contribute to the security of the cipher. (e.g., the initial and final permutation in the DES). Initial or terminal key addition is applied in several designs, e.g., IDEA, SAFER and Blowfish.

In order to make the cipher and its inverse more similar in structure, the linear mixing layer of the last round is different from the mixing layer in the other rounds. It can be shown that this does not improve or reduce the security of the cipher in any way. This is similar to the absence of the swap operation in the last round of the DES.

Motivation for design choices

In the following subsections, we will motivate the choice of the specific transformations and constants. We believe that the cipher structure does not offer enough degrees of freedom to hide a trap door.

The reduction polynomial $m(x)$

The polynomial $m(x)$ ('$11B$') for the multiplication in $GF(2^8)$ is the first one of the list of irreducible polynomials of degree 8.

The ByteSub S-box

The design criteria for the S-box are inspired by differential and linear cryptanalysis on the one hand and attacks using algebraic manipulations, such as interpolation attacks, on the other:

1. Invertibility
2. Minimisation of the largest non-trivial correlation between linear combinations of input bits and linear combination of output bits
3. Minimisation of the largest non-trivial value in the EXOR table
4. Complexity of its algebraic expression in $GF(2^8)$
5. Simplicity of description

For invertible S-boxes operating on bytes, the maximum input/output correlation can be made as low as 2^{-3} and the maximum value in the EXOR table can be as low as 4 (corresponding to a difference propagation probability of 2^{-6}).

By definition, the selected mapping has a very simple algebraic expression. This enables algebraic manipulations that can be used to mount attacks such as interpolation attacks. Therefore, the mapping is modified by composing it with an additional invertible affine transformation. This affine transformation does not affect the properties with respect tot the first three criteria, but if properly chosen, allows the S-box to satisfy the fourth criterion.

We have chosen an affine mapping that has a very simple description per se, but a complicated algebraic expression if combined with the 'inverse' mapping. It can be seen as modular polynomial multiplication followed by an addition:

$$b(x) = (x^7 + x^6 + x^2 + x) + a(x)(x^7 + x^6 + x^5 + x^4 + 1)\ mod\ x^8 + 1$$

The modulus has been chosen as the simplest modulus possible. The multiplication polynomial has been chosen from the set of polynomials coprime to the modulus as the one with the simplest description. The constant has been chosen in such a way that that the S-box has no fixed points ($S - box(a) = a$) and no 'opposite fixed points' ($S - box(a) = \bar{a}$).

Note: other S-boxes can be found that satisfy the criteria above. In the case of suspicion of a trapdoor being built into the cipher, the current S-box might be replaced by another one. The cipher structure and number of rounds as defined even allow the use of an S-box that does not optimise the differential and linear cryptanalysis properties (criteria 2 and 3). Even an S-box that is "average" in this respect is likely to provide enough resistance against differential and linear cryptanalysis.

The MixColumn transformation

MixColumn has been chosen from the space of 4-byte to 4-byte linear transformations according to the following criteria:

1. Invertibility;
2. Linearity in $GF(2)$
3. Relevant diffusion power
4. Speed on 8-bit processors
5. Symmetry
6. Simplicity of description

Criteria 2, 5 and 6 have lead us to the choice to polynomial multiplication modulo $x^4 + 1$. Criteria 1, 3 and 4 impose conditions on the coefficients. Criterion 4 imposes that the coefficients have small values, in order of preference '00', '01', '02', '03'…The value '00' implies no processing at all, for '01' no multiplication needs to be executed, '02' can be implemented using *xtime* and '03' can be implemented using *xtime* and an additional EXOR.

The criterion 3 induces a more complicated conditions on the coefficients.

Branch number

In our design strategy, the following property of the linear transformation of MixColumn is essential. Let F be a linear transformation acting on byte vectors and let the byte weight of a vector be the number of nonzero bytes (not to be confused with the usual significance of Hamming weight, the number of nonzero bits). The byte weight of a vector is denoted by $W(a)$. The Branch Number of a linear transformation is a measure of its diffusion power:

Definition: The branch number of a linear transformation F is

$$min_{a \neq 0}(W(a) + W(F(a))).$$

A non-zero byte is called an active byte. For MixColumn it can be seen that if a state is applied with a single active byte, the output can have at most 4 active bytes, as MixColumn acts on the columns independently. Hence, the upper bound for the branch number is 5. The coefficients have been chosen in such a way that the upper bound is reached. If the branch number is 5, a difference in 1 input (or output) byte propagates to all 4 output (or input) bytes, a 2-byte input (or output) difference to at least 3 output (or input) bytes. Moreover, a linear relation between input and output bits involves bits from at least 5 different bytes from input and output.

The ShiftRow offsets

The choice from all possible combinations has been made based on the following criteria:

1. The four offsets are different and C0 = 0
2. Resistance against attacks using truncated differentials
3. Resistance against the Square attack
4. Simplicity

For certain combinations, attacks using truncated differentials can tackle more rounds (typically only one) than for other combinations. For certain combinations the Square attack can tackle more rounds than others. From the combinations that are best with respect to criteria 2 and 3, the simplest ones have been chosen.

The key expansion

The key expansion specifies the derivation of the Round Keys in terms of the Cipher Key. Its function is to provide resistance against the following types of attack:

- Attacks in which part of the Cipher Key is known to the cryptanalyst
- Attacks where the Cipher Key is known or can be chosen, e.g., if the cipher is used as the compression function of a hash function
- Related-key attacks. A necessary condition for resistance against related-key attacks is that there should not be two different Cipher Keys that have a large set of Round Keys in common.

The key expansion also plays an important role in the elimination of symmetry:

- Symmetry in the round transformation: the round transformation treats all bytes of a state in very much the same way. This symmetry can be removed by having round constants in the key schedule;
- Symmetry between the rounds: the round transformation is the same for all rounds. This equality can be removed by having round-dependent round constants in the key schedule.

The key expansion has been chosen according to the following criteria:

- It shall use an invertible transformation, i.e., knowledge of any N_k consecutive words of the Expanded Key shall allow to regenerate the whole table;
- Speed on a wide range of processors;
- Usage of round constants to eliminate symmetries;
- Diffusion of Cipher Key differences into the Round Keys;
- Knowledge of a part of the Cipher Key or Round Key bits shall not allow to calculate many other Round Key bits.
- Enough non-linearity to prohibit the full determination of Round Key differences from Cipher Key differences only;
- Simplicity of description.

In order to be efficient on 8-bit processors, a light-weight, byte oriented expansion scheme has been adopted. The application of SubByte ensures the non-linearity of the scheme, without adding much space requirements on an 8-bit processor.

Number of rounds

We have determined the number of rounds by looking at the maximum number of rounds for which shortcut attacks have been found and added a considerable security margin. (A shortcut attack is an attack more efficient than exhaustive key search.)

For Rijndael with a block length and key length of 128 bits, no shortcut attacks have been found for reduced versions with more than 6 rounds. We added 4 rounds as a security margin. This is a conservative approach, because:

- Two rounds of Rijndael provide "full diffusion" in the following sense: every state bit depends on all state bits two rounds ago, or, a change in one state bit is likely to affect half of the state bits after two rounds. Adding 4 rounds can be seen as adding a "full diffusion" step at the beginning and at the end of the cipher. The high diffusion of a Rijndael round is thanks to its uniform structure that operates on all state bits. For so-called Feistel ciphers, a round only operates on half of the state bits and full diffusion can at best be obtained after 3 rounds and in practice it typically takes 4 rounds or more.
- Generally, linear cryptanalysis, differential cryptanalysis and truncated differential attacks exploit a propagation trail through n rounds in order to attack $n + 1$ or $n + 2$ rounds. This is also the case for the Square attack that uses a 4-round propagation structure to attack 6 rounds. In this respect, adding 4 rounds actually doubles the number of rounds through which a propagation trail has to be found.

For Rijndael versions with a longer Key, the number of rounds is raised by one for every additional 32 bits in the Cipher Key, for the following reasons:

- One of the main objectives is the absence of shortcut attacks, i.e., attacks that are more efficient than exhaustive key search. As with the key length the workload of exhaustive key search grows, shortcut attacks can afford to be less efficient for longer keys.
- Known-key (partially) and related-key attacks exploit the knowledge of cipher key bits or ability to apply different cipher keys. If the cipher key grows, the range of possibilities available to the cryptanalyst increases.

As no threatening known-key or related-key attacks have been found for Rijndael, even for 6 rounds, this is a conservative margin.

For Rijndael versions with a higher block length, the number of rounds is raised by one for every additional 32 bits in the block length, for the following reasons:

- For a block length above 128 bits, it takes 3 rounds to realise full diffusion, i.e., the diffusion power of a round, relative to the block length, diminishes with the block length.
- The larger block length causes the range of possible patterns that can be applied at the input/output of a sequence of rounds to increase. This added flexibility may allow to extend attacks by one or more rounds.

We have found that extensions of attacks by a single round are even hard to realise for the maximum block length of 256 bits. Therefore, this is a conservative margin.

LAMPIRAN B

Pewajaran bagi Prinsip Reka Bentuk PRESENT

Penyataan berikut diekstrak daripada dokumen yang bertajuk **PRESENT: An Ultra-Lightweight Block Cipher** untuk menunjukkan contoh penyataan bagi prinsip reka bentuk. Dokumen lengkap boleh didapati di http:// www.ist-ubisecsens.org/publications/present_ches2007.pdf.

Design principles of PRESENT

1. Goals and environment of use
    a. The cipher is to be implemented in hardware.
    b. Applications will only require moderate security levels. Consequently, 80-bit security will be adequate. Note that this is also the position taken for hardware profile stream ciphers submitted to eSTREAM.
    c. Applications are unlikely to require the encryption of large amounts of data. Implementations might therefore be optimised for performance or for space without too much practical impact.
    d. In some applications it is possible that the key will be fixed at the time of device manufacture. In such cases there would be no need to re-key a device (which would incidentally rule out a range of key manipulation attacks).
    e. After security, the physical space required for an implementation will be the primary consideration. This is closely followed by peak and average power consumption, with the timing requirements being a third important metric.
    f. In applications that demand the most efficient use of space, the block cipher will often only be implemented as encryption-only. In this way it can be used within challenge-response authentication protocols and, with some careful state management, it could be used for both encryption and decryption of communications to and from the device by using the counter mode.

2. The permutation layer

When choosing the mixing layer, our focus on hardware efficiency demands a linear layer that can be implemented with a minimum number of processing elements, i.e. transistors. This leads us directly to bit permutations. Given our focus on simplicity, we have chosen a regular bit-permutation and this helps to make a clear security analysis.

3. The S-box

We use a single 4-bit to 4-bit S-box S: F42 → F42 in present. This is a direct consequence of our pursuit of hardware efficiency, with the implementation of such an S-box typically being much more compact than that of an 8-bit S-box. Since we use a bit permutation for the linear diffusion layer, AES-like diffusion techniques are not an option for present. Therefore, we place some additional conditions on the S-boxes to improve the so-called avalanche of change.

LAMPIRAN C

Pewajaran bagi Prinsip Reka Bentuk CHACHA20

Penyataan berikut diekstrak daripada dokumen yang bertajuk **ChaCha, a variant of Salsa20** untuk menunjukkan contoh penyataan bagi prinsip reka bentuk. Dokumen lengkap boleh didapati di https://cr.yp.to/chacha/chacha-20080120.pdf.

Design principles of ChaCha20

1. Introduction

ChaCha follows the same basic design principles as Salsa20, but I changed some of the details, most importantly to increase the amount of diffusion per round. I speculate that the minimum number of secure rounds for ChaCha is smaller than the minimum number of secure rounds for Salsa20.

This extra diffusion does not come at the expense of extra operations. A ChaCha round has 16 additions and 16 xors and 16 constant-distance rotations of 32-bit words, just like a Salsa20 round. Furthermore, ChaCha has the same levels of parallelism and vectorizability as Salsa20, and saves one of the 17 registers used by a "natural" Salsa20 implementation. So it is reasonable to guess that a ChaCha round can achieve the same software speed as a Salsa20 round—and even better speed than a Salsa20 round on some platforms. Consequently, if ChaCha has the same minimum number of secure rounds as Salsa20, then ChaCha will provide better overall speed than Salsa20 for the same level of security.

Of course, performance should be measured, not guessed! I wrote and posted new public-domain software for ChaCha, and timed that software, along with the fastest available Salsa20 software, on several computers, using the latest version (20080120) of the eSTREAM benchmarking framework.

2. The quarter-round

ChaCha, like Salsa20, uses 4 additions and 4 xors and 4 rotations to invertibly update 4 32-bit state words. However, ChaCha applies the operations in a different order, and in particular updates each word twice rather than once. Specifically, ChaCha updates a, b, c, d as follows:

```
a += b; d ^= a; d <<<= 16;
c += d; b ^= c; b <<<= 12;
a += b; d ^= a; d <<<= 8;
c += d; b ^= c; b <<<= 7;
```

3. The matrix

ChaCha, like Salsa20/*r*, builds a 4 × 4 matrix, invertibly transforms the matrix through *r* rounds, and adds the result to the original matrix to obtain a 16-word (64-byte) output block. There are three differences in the details. First, ChaCha permutes the order of words in the output block to match the permutation described above. This has no effect on security; it saves time on SIMD platforms; it makes no difference in speed on other platforms. Second, ChaCha builds the initial matrix with all attacker-controlled input words at the bottom.

LAMPIRAN D

Pewajaran bagi Prinsip Reka Bentuk Keccak

Penyataan berikut diekstrak daripada dokumen yang bertajuk **Keccak sponge function family main document** untuk menunjukkan contoh penyataan bagi prinsip reka bentuk. Dokumen lengkap boleh didapati di https:// keccak.noekeon.org/Keccak-main-2.1.pdf.

1) Choosing the sponge construction

Defining a generic attack:

Definition 1: A shortcut attack on a sponge function is a generic attack if it does not exploit specific properties of the underlying permutation (or transformation).

The Keccak hash function makes use of the sponge construction. This results in the following property:

Provability: It has a proven upper bound for the success probability, and hence also a lower bound for the expected workload, of generic attacks.

The design philosophy underlying Keccak is the hermetic sponge strategy. This consists of using the sponge construction for having provable security against all generic attacks and calling a permutation (or transformation) that should not have structural properties with the exception of a compact description. Additionally, the sponge construction has the following advantages over constructions that make use of a compression function:

a. **Simplicity:** Compared to the other constructions for which upper bounds have been proven for the success of generic attacks, the sponge construction is very simple, and it also provides a bound that can be expressed in a simple way.
b. **Variable-length output:** It can generate outputs of any length and hence a single function can be used for different output lengths.
c. **Flexibility:** Security level can be incremented at the cost of speed by trading in bitrate for capacity, using the same permutation (or transformation).
d. **Functionality:** Thanks to its long outputs and proven security bounds with respect to generic attacks, a sponge function can be used in a straightforward way as a MAC function, stream cipher, a re-seedable pseudorandom bit generator and a mask generating function.

To support arbitrary bit strings as input, the sponge construction requires a padding function. We refer to Section 3.2 of Keccak sponge function family main document for a rationale for the specific padding function we have used.

2) Choosing an iterated permutation

The sponge construction requires an underlying function f, either a transformation or a permutation. f should be such that it does not have properties that can be exploited in shortcut attacks. We have chosen a permutation, constructed as a sequence of almost identical rounds because of the following advantages:

a. **Block cipher experience:** An iterated permutation is an iterated block cipher with a fixed key. In its design one can build on knowledge obtained from block cipher design and cryptanalysis.
b. **Memory efficiency:** Often a transformation is built by taking a permutation and adding a feedforward loop. This implies that (at least part of) the input must be kept during the complete computation. This is not the case for a permutation, leading to a relatively small RAM footprint.
c. **Compactness:** Iteration of a single round leads to a compact specification and potentially compact code and hardware circuits.

3) Designing the Keccak-f permutations

The design criterion for the Keccak-f permutations is to have no properties that can be exploited in a shortcut attack when being used in the sponge construction. It is constructed as an iterated block cipher similar to Noekeon and Rijndael, with the key schedule replaced by some simple round constants. Here we give a rationale for its features:

a. **Bit-oriented structure Attacks:** Where the bits are grouped (e.g., in bytes), such as integral cryptanalysis and truncated trails or differentials, are unsuitable against the Keccak-f structure.
b. **Bitwise logical operations and fixed rotations:** Dependence on CPU word length is only due to rotations, leading to an efficient use of CPU resources on a wide range of processors. Implementation requires no large tables, removing the risk of table-lookup based cache miss attacks. They can be programmed as a fixed sequence of instructions, providing protection against timing attacks.
c. **Symmetry**: This allows to have very compact code in software and a very compact co-processor suitable for constrained environments.
d. **Parallelism**: Thanks to its symmetry and the chosen operations, the design is well-suited for ultra-fast hardware implementations and the exploitation of SIMD instructions and pipelining in CPUs.
e. **Round degree 2:** This makes the analysis with respect to differential and linear cryptanalysis easier, leads to relatively simple (albeit large) systems of algebraic equations and allows the usage of very powerful protection measures against differential power analysis (DPA) both in software and hardware that are not suited for most other nonlinear functions.
f. **Matryoshka structure:** The analysis of small versions is relevant for larger versions.
g. **Eggs in another basket:** The choice of operations is very different from that in SHA-1 and the members of the SHA-2 family on the one hand and from AES on the other.

4) Choosing the parameter values

In Keccak, there are basically three security-relevant parameters that can be varied:

a. b: width of Keccak-f,
b. c: capacity, limited by $c < b$,
c. n_r: number of rounds in Keccak-f.

The parameters of the candidate sponge functions have been chosen for the following reasons.

a. $c = 2n$: for the fixed-output-length candidates, we chose a capacity equal to twice the output length n. This is the smallest capacity value such that there are no generic attacks with expected complexity below 2^n.
b. $b = 1600$: The width of the Keccak-f permutation is chosen to favor 64-bit architectures while supporting all required capacity values using the same permutation.
c. Parameters for Keccak[]: for the variable-output-length candidate Keccak[], we chose a rate value that is a power of two and a capacity not smaller than 512 bits and such that their sum equals 1600. This results in $r = 1024$ and $c = 576$. This capacity value precludes generic attacks with expected complexity below 2288. A rate value that is a power of two may be convenient in some applications to have a block size which is a power of two, e.g., for a real-time application to align its data source (assumed to be organized in blocks of size a power of two) to the block size without the need of an extra buffer.
d. $n_r = 24$: The value of n_r has been chosen to have a good safety margin with respect to even the weakest structural distinguishers and still have good performance.

5) The difference between version 1 and version 2 of Keccak

For the 2nd round of the SHA-3 competition, we decided to modify Keccak. There are basically two modifications: the increase of the number of rounds in Keccak-f and the modification of the rate and capacity values in the four fixed-output-length candidates for SHA-3:

a. Increasing the number of rounds of Keccak-f from $12 + l$ to $12 + 2l$ (from 18 to 24 rounds for Keccak-f[1600]): this modification is due to the distinguishers that work on reduced-round variants of Keccak-f[1600] up to 16 rounds. In the logic of the hermetic sponge strategy, we want the underlying permutation to have no structural distinguishers. Sticking to 18 rounds would not contradict this strategy but would leave a security margin of only 2 rounds against a distinguisher of Keccak-f. In addition, we do think that this increase in the number of rounds increases the security margin with respect to distinguishers of the resulting sponge functions and attacks against those sponge functions.
b. For applications where the bitrate does not need to be a power of 2, the new parameters of the fixed-output-length candidates take better advantage of the performance-security trade-offs that the Keccak sponge function allows.

LAMPIRAN E

Pewajaran bagi Prinsip Reka Bentuk SPONGENT

Penyataan berikut diekstrak daripada dokumen yang bertajuk **SPONGENT: The Design of Lightweight Cryptographic Hashing** untuk menunjukkan contoh penyataan bagi prinsip reka bentuk. Dokumen lengkap boleh didapati di https://eprint.iacr.org/2011/697.pdf.

The overall design approach for SPONGENT is to target low area while favoring simplicity. The 4-bit S-box is the major block of functional logic in a serial low-area implementation of SPONGENT. It fulfills the present design criteria in terms of differential and linear properties. Moreover, any linear approximation over the S-box involving only single bits both in the input and output masks is unbiased. This aims to restrict the linear hull effect discovered in round-reduced PRESENT.

The function of the bit permutation pLayer is to provide good diffusion, by acting together with the S-box, while having a limited impact on the area requirements. This is its main design goal, while a bit permutation may occupy additional space in silicon. The counters lCounter and ɹǝɬnuoƆI are mainly aimed to prevent sliding properties and make prospective cryptanalysis approaches using properties like invariant subspaces more involving.

The structures of the bit permutation and the S-box in SPONGENT make it possible to prove the following differential property:

Theorem 1: Any 5-round differential characteristic of the underlying permutation of SPONGENT with $b \geq 64$ has a minimum of 10 active S-boxes. Moreover, any 6-round differential characteristic of the underlying permutation of SPONGENT with $b \geq 256$ has a minimum of 14 active S-boxes.

The concept of counting active S-boxes is central to the differential cryptanalysis. The minimum number of active S-boxes relates to the maximum differential characteristic probability of the construction. Since in the hash setting there are no random and independent key values added between the rounds, this relation is not exact (in fact that it is even not exact for most practical keyed block ciphers). However, differentially active S-boxes are still the major technique used to evaluate the security of SPN-based hash functions.

An important property of the SPONGENT S-box is that its maximum differential probability is 2^{-2}. This fact and the assumption of the independency of difference propagation in different rounds yield an upper bound on the differential characteristic probability of 2^{-20} over 5 rounds and of 2^{-28} over 6 rounds for $b \geq 256$ which follows from the claims of Theorem 1.

Theorem 1 is used to determine the number *R* of rounds in permutation π_b: *R* is chosen in a way that π_b provides at least *b* active S-boxes.

LAMPIRAN F

Kategori Data untuk Block Cipher

Teks sifer yang dihasilkan daripada *block cipher* hanya mengandungi jujukan bit yang panjangnya sama dengan saiz blok bagi *block cipher* (contohnya teks sifer bagi LBlock Cipher ialah 64-bit). Walau bagaimanapun, untuk menilai kerawakan algoritma kriptografi, teks sifer yang dihasilkan perlu mengandungi jujukan bit yang besar. Bagi mencapai tujuan ini, Kategori Data digunakan untuk menjana input (teks biasa atau kekunci) untuk *block cipher* agar menghasilkan teks sifer yang akan dirangkaikan dengan cara yang tertentu. Huraian terperinci bagi sembilan kategori data yang digunakan dalam block cipher adalah seperti di bawah:

i. *Strict Key Avalanche (SKA)*
ii. *Strict Plaintext Avalanche (SPA)*
iii. *Plaintext / Ciphertext Correlation (PCC)*
iv. *Cipher Block Chaining Mode (CBCM)*
v. *Random Plaintext / Random Key (RPRK)*
vi. *Low Density Key (LDK)*
vii. *High Density Key (HDK)*
viii. *Low Density Plaintext (LDP)*
ix. *High Density Plaintext (HDP)*

1. <u>Strict Key Avalanche (SKA)</u>

Kategori data *Strict Key Avalanche* digunakan untuk memeriksa sensitiviti *block cipher* terhadap perubahan dalam kekunci x −bit. Bagi blok teks biasa yang tetap, kesan *avalanche* dicapai apabila mana-mana bit kekunci dilengkapkan dan setiap bit blok teks sifer berubah dengan kebarangkalian satu perdua.

Setiap sampel bagi kategori data ini menggunakan teks biasa yang ditetapkan kepada semua sifar dan X blok x −bit kekunci asas yang rawak. Teks biasa yang kesemuanya sifar terlebih dahulu disulitkan menggunakan setiap kekunci asas. Kemudian, setiap kekunci asas dibalikkan pada bit ke-i , bagi $1 \leq i \leq x$ yang menghasilkan jumlah kekunci yang diubah sebanyak $(X * x)$ kali. Kemudian, teks biasa yang kesemuanya sifar disulitkan menggunakan setiap kunci yang diubah. Semua teks sifer yang terhasil menggunakan kekunci yang diubah akan melalui operasi XOR menggunakan teks sifer yang terhasil daripada penyulitan menggunakan kunci asas yang sepadan. Produk output daripada operasi XOR itu dipanggil blok terbitan dan akan dirangkaikan untuk membina jujukan bit yang besar.

2. <u>Strict Plaintext Avalanche (SPA)</u>

Kategori data *Strict Plaintext Avalanche* digunakan untuk memeriksa sensitiviti *block cipher* terhadap perubahan dalam teks biasa y −bit. Bagi kekunci yang tetap, kesan *avalanche* dicapai apabila mana-mana bit teks biasa dilengkapkan dan setiap bit blok teks sifer berubah dengan kebarangkalian satu perdua.

Setiap sampel bagi kategori data ini menggunakan kekunci yang ditetapkan kepada semua sifar dan Y blok y −bit teks biasa asas yang rawak. Setiap teks biasa asas terlebih dahulu disulitkan menggunakan kekunci yang kesemuanya sifar. Kemudian, setiap teks biasa asas dibalikkan pada bit ke-i , bagi $1 \leq i \leq y$ yang menghasilkan jumlah teks biasa yang diubah sebanyak $(Y * y)$ kali Kemudian, setiap teks biasa yang diubah akan disulitkan menggunakan kekunci yang kesemuanya sifar. Semua teks sifer yang terhasil daripada teks biasa yang diubah akan melalui operasi XOR menggunakan teks sifer yang terhasil daripada penyulitan teks biasa asas yang sepadan. Produk output daripada operasi XOR itu dipanggil blok terbitan dan akan dirangkaikan untuk membina jujukan bit yang besar.

3. <u>Plaintext / Ciphertext Correlation (PCC)</u>

Kategori data *Plaintext / Ciphertext Correlation* digunakan untuk memeriksa hubung kait antara pasangan teks biasa / teks sifer dan menggunakan mod operasi ECB.

Setiap sampel bagi kategori data ini menggunakan Y blok y −bit teks biasa yang rawak dan satu x −bit kekunci yang rawak. Setiap blok teks biasa disulitkan menggunakan x −bit kekunci yang rawak. Kemudian, teks sifer yang terhasil akan melalui operasi XOR menggunakan teks biasa yang sepadan. Produk output daripada operasi XOR itu dipanggil blok terbitan dan akan dirangkaikan untuk membina jujukan bit yang besar.

4. <u>Cipher Block Chaining Mode (CBCM)</u>

Kategori data *Ciphertext Block Chaining Mode* menggunakan mod operasi CBC. Dalam kategori data ini, setiap blok teks biasa akan melalui operasi XOR menggunakan blok teks sifer yang terdahulu sebelum disulitkan, manakala blok pertama akan melalui operasi XOR menggunakan vektor pemulaan. Perubahan satu-bit dalam mana-mana teks biasa atau vektor pemulaan akan memberi kesan terhadap semua blok teks sifer yang berikutnya.

Setiap sampel bagi kategori data ini menggunakan teks biasa yang ditetapkan kepada semua sifar (P), x −bit kekunci yang rawak (K) dan vektor pemulaan yang kesemuanya sifar (IV). Proses penyulitan dilakukan sebanyak I kali. Blok terbitan bagi kategori data ini ialah blok teks sifer dalam mod operasi CBC. Blok teks sifer yang pertama, C_1 ditakrifkan sebagai $C_1 = E_K(IV \oplus P_1)$, yang mana blok teks sifer yang berikutnya ditakrifkan sebagai $C_i = E_K(C_{i-1} \oplus P_i)$ bagi $1 \leq i \leq I$.

5. Random Plaintext / Random Key (RPRK)

Kategori data *Random Plaintext / Random Key* digunakan untuk memeriksa kerawakan teks sifer berdasarkan teks biasa rawak dan kekunci rawak. Setiap sampel bagi kategori data ini menggunakan Y blok y −bit teks biasa yang rawak dan satu x-bit kekunci yang rawak. Setiap blok teks biasa disulitkan menggunakan x −bit kekunci yang rawak. Blok terbitan bagi kategori data ini ialah blok teks sifer dalam mod operasi ECB yang akan dirangkaikan untuk membina jujukan bit yang besar.

6. Low Density Key (LDK)

Kategori data *Low Density Keys* dibentuk berdasarkan x −bit kekunci yang berkepadatan rendah. Setiap sampel bagi kategori data ini menggunakan Y blok y-bit teks biasa yang rawak dan X blok kekunci x −bit yang khusus. Blok teks biasa yang pertama disulitkan menggunakan x −bit kekunci yang kesemuanya sifar. Kemudian, blok teks biasa akan disulitkan menggunakan x-bit kekunci dengan hanya satu bit '1' dalam setiap kedudukan x-bit kekunci dan semua bit kekunci yang lain ditetapkan kepada bit '0'. Ini akan menghasilkan Y_1 blok teks sifer. Kemudian, blok teks biasa akan disulitkan menggunakan x-bit kekunci dengan dua bit '1' dalam setiap gabungan kedudukan dua bit kekunci dan semua bit kekunci yang lain ditetapkan kepada bit '0'. Ini akan menghasilkan C_r^n blok teks sifer, yang mana $n = x$ dan $r = 2$. Secara keseluruhannya, blok terbitan bagi kategori data ini ialah $Y = 1 + Y_1 + C_2^x$ blok teks sifer dalam mod operasi ECB dan akan dirangkaikan untuk membina jujukan bit yang besar.

7. High Density Key (HDK)

Kategori data *High Density Keys* dibentuk berdasarkan x −bit kekunci berkepadatan tinggi. Setiap sampel bagi kategori data ini menggunakan Y blok y −bit teks biasa yang rawak dan X blok kekunci x −bit yang khusus. Blok teks biasa yang pertama disulitkan menggunakan x −bit kekunci yang kesemuanya satu. Kemudian, blok teks biasa akan disulitkan menggunakan x −bit kekunci dengan hanya satu bit '0' dalam setiap kedudukan kekunci x −bit dan semua bit kekunci yang lain ditetapkan kepada bit '1'. Ini akan menghasilkan Y_1 blok teks sifer. Kemudian, blok teks biasa akan disulitkan menggunakan x −bit kekunci dengan dua bit '0' dalam setiap gabungan kedudukan dua bit kekunci dan semua bit kekunci yang lain ditetapkan kepada bit '1'. Ini akan menghasilkan C_r^n blok teks sifer, yang mana $n = x$ dan $r = 2$. Secara keseluruhannya, blok terbitan bagi kategori data ini ialah $Y = 1 + Y_1 + C_2^x$ blok teks sifer dalam mod operasi ECB dan akan dirangkaikan untuk membina jujukan bit yang besar.

8. Low Density Plaintext (LDP)

Kategori data *Low Density Plaintext* dibentuk berdasarkan y −bit blok teks biasa berkepadatan rendah. Setiap sampel bagi kategori data ini menggunakan X blok x −bit kekunci yang rawak dan Y blok y −bit blok teks biasa yang khusus. y −bit blok teks biasa yang kesemuanya sifar akan disulitkan menggunakan x −bit kekunci rawak yang pertama. Kemudian, blok teks biasa dengan hanya satu bit '1' dalam setiap kedudukan y-bit teks biasa dan semua bit teks biasa lain yang ditetapkan kepada bit '0', akan disulitkan menggunakan x −bit kekunci rawak yang lain. Ini akan menghasilkan X_1 blok teks sifer. Kemudian, blok teks biasa dengan dua bit '1' dalam setiap gabungan kedudukan dua bit teks biasa dan semua bit teks biasa lain yang ditetapkan kepada bit '0', akan disulitkan menggunakan x −bit kekunci rawak yang lain. Ini akan menghasilkan C_r^n blok teks sifer, yang mana $n = y$ dan $r = 2$. Secara keseluruhannya, blok terbitan bagi kategori data ini ialah $X = 1 + X_1 + C_2^y$ blok teks sifer dalam mod operasi ECB dan akan dirangkaikan untuk membina jujukan bit yang besar.

9. High Density Plaintext (HDP)

Kategori data *High Density Plaintext* dibentuk berdasarkan y −bit blok teks biasa berkepadatan tinggi. Setiap sampel bagi kategori data ini menggunakan X blok x −bit kekunci yang rawak dan Y blok y −bit blok teks biasa yang khusus. y −bit blok teks biasa yang kesemuanya sifar akan disulitkan menggunakan x −bit kekunci yang pertama. Kemudian, blok teks biasa dengan hanya satu bit '0' dalam setiap kedudukan y −bit teks biasa dan semua bit teks biasa lain yang ditetapkan kepada bit '1', akan disulitkan menggunakan x −bit kekunci rawak yang lain. Ini akan menghasilkan X_1 blok teks sifer. Kemudian, blok teks biasa dengan dua bit '0' dalam setiap gabungan kedudukan dua bit teks biasa dan semua bit teks biasa lain yang ditetapkan kepada bit '1', akan disulitkan menggunakan x −bit kekunci rawak yang lain. Ini akan menghasilkan C_r^n blok teks sifer, yang mana $n = y$ dan $r = 2$. Secara keseluruhannya, blok terbitan bagi kategori data ini ialah $X = 1 + X_1 + C_2^y$ blok teks sifer dalam mod operasi ECB dan akan dirangkaikan untuk membina jujukan bit yang besar.

LAMPIRAN G

BORANG PENYERAHAN ALGORITMA

Algorithm Submission Form

SENARAI ALGORITMA KRIPTOGRAFI TERPERCAYA NEGARA (MySEAL)

MAKLUMAT SERAHAN *Submission Information*	
☐ Individu *Individual*	☐ Organisasi *Organisation*

A. MAKLUMAT PENYERAH

Submitter Information

MAKLUMAT PENYERAH *Submitter Information*	
NAMA PENYERAH UTAMA *Principal Submitter's Name*	
NO TELEFON PEJABAT *Office Tel No*	
NO TELEFON MUDAH ALIH *Mobile No*	
NO FAKSIMILI *Fax No*	
ALAMAT E-MEL *E-mail Address*	
ALAMAT SURAT MENYURAT *Postal Address*	

NAMA PENYERAH TAMBAHAN (jika berkenaan) *Name of Auxiliary Submitter(s)* *(if any)*	
NAMA PEREKA CIPTA / PEMBANGUN ALGORITMA *Name of Algorithm Inventor(s) / Developer(s)*	
NAMA PEMILIK ALGORITMA (jika berlainan daripada penyerah utama) *Name of Algorithm's Owner* *(if different from the submitter)*	
TANDATANGAN PENYERAH *Signature of Submitter*	

MAKLUMAT ORGANISASI (untuk serahan organisasi sahaja) *Organisation Information (for organisation submission only)*	
ORGANISASI *Organisation*	
ALAMAT *Address*	

B. MAKLUMAT ALGORITMA

Algorithm Information

NAMA ALGORITMA *Name of algorithm*	
PRIMITIF ALGORITMA KRIPTOGRAFI *Cryptographic Algorithm Primitive*	☐ *a) Symmetric Block Cipher* ☐ *Block Cipher* ☐ *Lightweight Block Cipher* ☐ *b) Symmetric Stream Cipher* ☐ *Synchronous stream cipher* ☐ *Self- Synchronous stream cipher* ☐ *c) Asymmetric* ☐ *Encryption* ☐ *Key agreement* ☐ *Digital signature* ☐ *d) Cryptographic Hash Function* ☐ *Cryptographic hash function* ☐ *Lightweight hash function* ☐ *e) Cryptographic Key Generation* ☐ *f) Cryptographic Pseudo Random Number Generator Primitive*
CADANGAN TAHAP KESELAMATAN *Proposed Security Level*	☐ 40 – bit ☐ 80 – bit ☐ 128 – bit ☐ 192 – bit ☐ 256 – bit ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify* _____
CADANGAN PLATFORM *Proposed Environment*	☐ Perkakasan *Hardware* ☐ Perisian *Software*

C. MAKLUMAT TAMBAHAN

Additional Information

PENGGUNAAN *Implementation*	☐ *Bluetooth* ☐ *Global System for Mobile communications (GSM)* ☐ *Radio-Frequency Identification (RFID)* ☐ *Kad pintar (Smart cards)* ☐ *Peranti mudah alih (Mobile devices)* ☐ *Rangkaian pengesan (Sensor network)* ☐ *Mikropengawal (Microcontroller)* ☐ *Mikropemproses (Microprocessor)* ☐ *Field-programmable gate array (FPGA)* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify* _____

D. PENYATAAN PENYERAH

Statement by the Submitter

1. Submission statement

☐ I/We do hereby understand that my/our submitted algorithm may not be selected for inclusion in MySEAL. I/We also understand and agree that after the close of the submission period, my/our submission may not be withdrawn. I/We further understand that I/we will not receive financial compensation from MySEAL project for my/our submission.

2. Statement that there are no hidden weaknesses in the algorithm design

☐ I/We certify that, to the best of my knowledge, I/we have fully disclosed there are no hidden weaknesses in my/our algorithm.

☐ I/We hereby enclosed information on the known weaknesses of my/our algorithm […… (file/attachment name)]

3. Intellectual Property Statement for the Submission of …… [*name of algorithm*] to the MySEAL Project

☐ …… [Submitter] currently has patents pending / has not filed for patents on the …… [name of algorithm]. The …… [name of algorithm] is provided royalty-free for commercial and non-commercial use in non-embedded applications. Licenses for use of the …… [name of algorithm] in embedded applications may be obtained from …… [name]. Aside from legal restrictions applying to encryption algorithms (if any), these licenses will be issued on a non-discriminatory basis. We will undertake to update the MySEAL project when necessary.

Diserahkan oleh (Tandatangan & Cop): *Submitted by (Signature & Stamp):* Tarikh: *Date:*	Diterima oleh (Tandatangan & Cop): *Received by (Signature & Stamp):* Tarikh: *Date:*

LAMPIRAN H

SENARAI SEMAK

Checklist

Bil *No*	**Perkara / Dokumen diperlukan** *Document(s) needed*	**Disertakan oleh Penyerah (√)** *Supplied by Submitter (√)*	**Disemak oleh Penerima (√)** *Checked by Receiver (√)*	**Catatan** *Notes*
1	Profil Syarikat *Company Profile*			
2	Laporan analisis *Analysis Report* *a) Symmetric Block Cipher* ☐ Ujian statistik NIST *NIST statistical tests* ☐ *Linear cryptanalysis* ☐ *Differential cryptanalysis* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify*. _____ *b) Symmetric Stream Cipher* ☐ Ujian statistik NIST *NIST statistical tests* ☐ *Algebraic attack* ☐ *Correlation attack* ☐ *Distinguishing attack* ☐ *Guess-and-Determine attack* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify.* _____			

Bil ***No***	**Perkara / Dokumen diperlukan** ***Document(s) needed***	**Disertakan oleh Penyerah (√)** ***Supplied by Submitter (√)***	**Disemak oleh Penerima (√)** ***Checked by Receiver (√)***	**Catatan** ***Notes***
	c) Asymmetric Cryptographic Algorithm ☐ *Hard Mathematical Problems and assumptions* ☐ *Security Model and its proof* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify.* _____ *d) Cryptographic Hash Function* ☐ *Pre-image resistance* ☐ *Second pre-image resistance* ☐ *Collision resistance* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify.* _____ *e) Cryptographic Key Generation* ☐ *Probabilistic Prime Generators* ☐ *Deterministic Prime Generators* ☐ *Distinguishing Carmichael numbers from prime numbers* ☐ *Generation of pseudo primes samples from the generator* ☐ Ujian statistik NIST *NIST statistical tests* ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify.* _____			

Bil ***No***	**Perkara / Dokumen diperlukan** ***Document(s) needed***	**Disertakan oleh Penyerah (√)** ***Supplied by Submitter (√)***	**Disemak oleh Penerima (√)** ***Checked by Receiver (√)***	**Catatan** ***Notes***
	f) Cryptographic Pseudo Random Number Generator Primitive ☐ PRNG based on asymmetric methodologies ☐ PRNG based on symmetric methodologies ☐ PRNG not based on asymmetric or symmetric methodologies ☐ NIST statistical tests ☐ NIST statistical tests ☐ Lain-lain. Sila nyatakan. *Other(s). Please specify.* _____			
3	Laporan prestasi algoritma mengikut keupayaan perkakasan dan/atau perisian *Implementation and performance reports on hardware and/or software*			
4	Laporan reka bentuk *Justification on design principles*			
5	Vektor ujian *Test vectors*			
6	Penyata / perjanjian / pendedahan Harta Intelek *Intellectual Property statements / agreements / disclosures*			

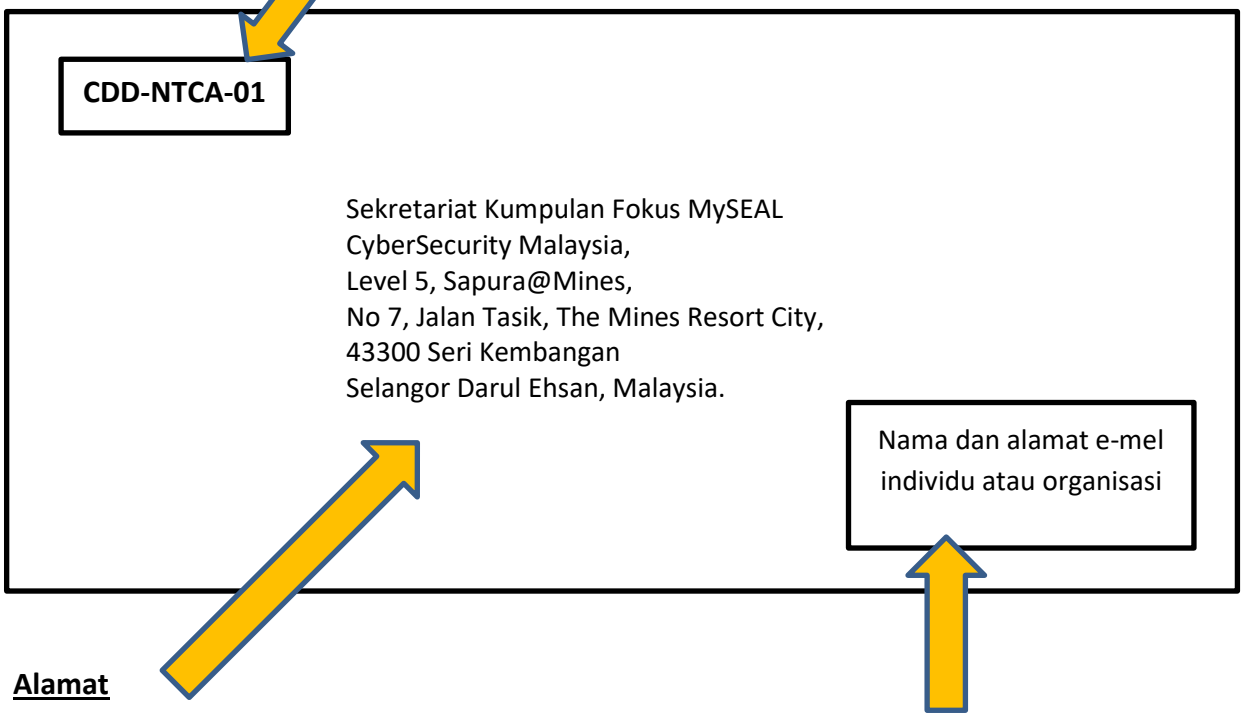
LAMPIRAN I

GARIS PANDUAN LABEL UNTUK PENYERAHAN PAKEJ PROJEK MySEAL

BAHAGIAN HADAPAN PAKEJ

Kod Projek

Tuliskan kod ini pada pakej anda



Alamat

Alamat rasmi untuk penyerahan

Maklumat Penyerah

Tuliskan nama dan alamat e-mel penyerah atau organisasi