Secure Data in PCIe/NVMe SSD

Larry Ko

Larry.ko@sage-micro.com

VP Engineering, Sage Microelectronics Corp.

Introduction

- NVMe is a low latency, highly scalable, and highly parallel interface

  -A SSD controller with multi-core architecture is one of solutions.

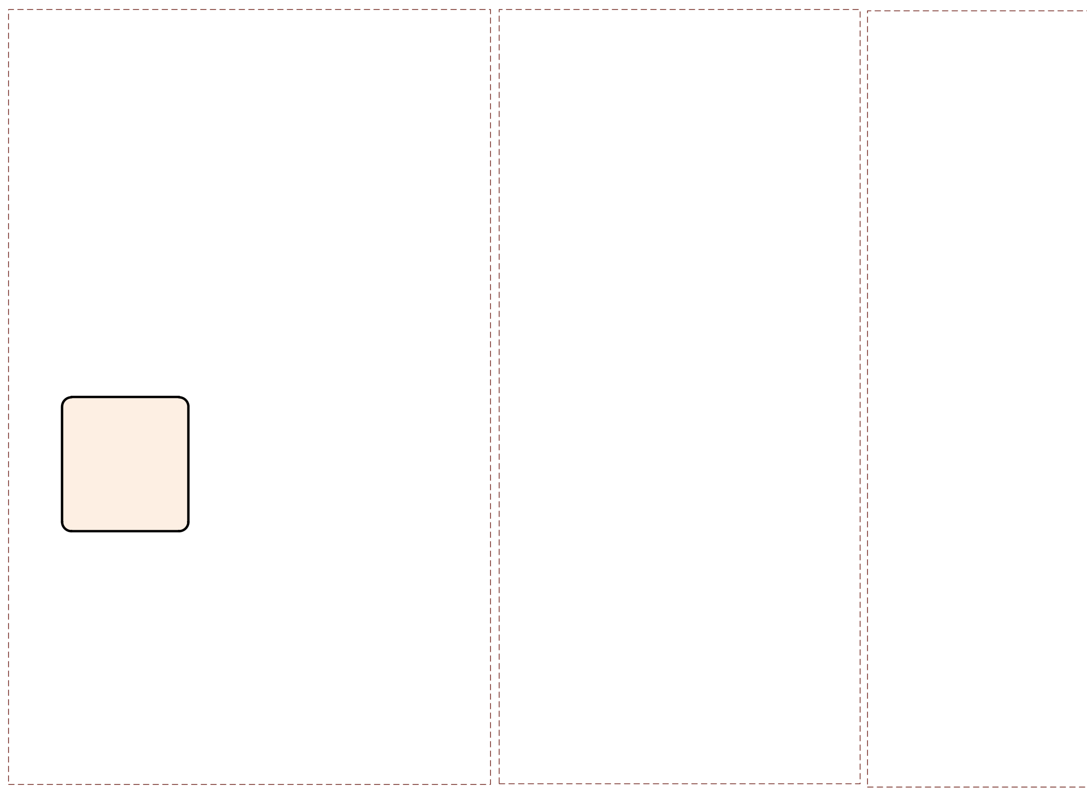
  -Performance/cost trade-off consideration

- Crypto algorithms are embedded in the controller to secure data in the SSD

  -Industry standard: RSA, SHA-1/2 and AES-128/256

  -China: SM2, SM3 and SM4.

Multi-core Architecture



Performance/Cost Trade-off

- Controller with DDR3 DRAM

-Keep mapping tables in DRAM to eliminate the overhead for mapping table update and garbage collection on NAND to optimize the IOPS rate, especially on random write. However, cost effect needs to be considered.

- Controller without DDR3 DRAM

-IOPS on random write will be suffered the most for NAND mode, but the cost is lowest.

-Select eMMC mode, a decent IOPS on random write with reasonable cost can be achieved.

Random IOPS: 80K

China Cipher Algorithm – SM2

- SM2 algorithm

-Published by China in 2010

-An asymmetric cryptographic algorithm based on elliptic curves cryptography (ECC).

-Recommended parameters for EC over 256 bit prime field

Equation: $y^2 = x^3 + ax + b$, $a, b \in Fp$.

Prime p, coeficients of equation a, b, base point G(x,y) and order n

-Key pair Generation:

-Randomly select a integer $d \in [1, n-2]$

-Calculate $P = [d]G$ over elliptic Curve

-d is the private key and P is the public key

-Once the key pair (d, P) is generated, a variety of cryptosystems such as public key encryption, digital signature, key exchange can be set up.

China Cipher Algorithm – SM2

-Public Key Encryption Algorithm

-Encryption with Public key

User A's data: elliptic curve parameters, message M with length klen, and public key P_B.

1. Generate the random number $k \in [1, n-1]$
2. Compute EC point $C_1=[k]G=(x_1,y_1)$
3. Compute EC point $S=[h]P_B$, report error if S is infinity
4. Compute EC point $[k]P_B=(x_2,y_2)$
5. Calculate $t=KDF(x_2||y_2, klen)$ through Key Derivation function, go to step 1 if t is all zero
6. Compute $C_2 = M \oplus t$
7. Compute the hash value $C_3=Hash(x_2 || M || y_2)$
8. Output the ciphertext $C = C_1||C_2||C_3$

China Cipher Algorithm – SM2

-Public Key Encryption Algorithm

-Decryption with Private key

User B's data: elliptic curve parameters, ciphertext, d_B.

Let *klen* be the bit length of C_2

1. Get C_1 from C, verify C_1 if satisfies the elliptic curve equation, report error if not.
2. Calculate EC point $S=[h]C_1$, report error if S is infinity
3. Compute the point $[d_B]C_1=(x_2,y_2)$
4. Compute $t=KDF(x_2||y_2, klen)$ through Key Derivation function
5. Get C_2 from C, calculate $M'=C_2 \oplus t$
6. Caculate hash $u=Hash(x_2||M'||y_2)$, report error if u is not equal to C_3
7. Output the plaintext M'

China Cipher Algorithm – SM3

- SM3 hash algorithm

  -Published by China in 2010

  -A Chinese hash function which is very similar to SHA-256.

  -Input: Message with length < 2^{64}

  -Output: 256-bit hash value

  -Algorithm:

    1. Pad the message to be a multiple of 512 bit blocks

       -Pad message with '1' then k zero bits

       -Append a 64 bits block represents message length

       -((Message length + 1 + k) mod 512) + 64 = 512

    2. Message expansion:

       -Expand each 512-bit padded message into 132 words, $W_0 \sim W_{67}$, $W'_0 \sim W'_{63}$

       -$W_0 \sim W_{63}$ and $W'_0 \sim W'_{63}$ are used for hash computation

China Cipher Algorithm – SM3

3. Iterative Compression:

Let A, B, C, D, E, F, G be 32-bit word registers; $SS1$, $SS2$, $TT1$ and $TT2$ be intermediate 32-bit variables.

For i = 0 to n-1 { //n: number of blocks in the padded message

$ABCDEFG = V(\mathrm{i})$ //$V(0)$: initial hash value

For j = 0 to 63 { //64 rounds for each 512-bit block

$SS1 = ((A{<}{<}{<}12) + E + (Tj{<}{<}{<}j)){<}{<}{<}7$ //circular shift-left, $\mathrm{mod}2^{32}$

$SS2 = SS1 \;\; (A{<}{<}{<}12)$

$TT1 = FFj(A,B,C) + D + SS2 + W'_j$ //FF_j: a boolean function

$TT2 = GGj(E,F,G) + H + SS1 + W_j$ //GG_j: a boolean function

$D = C$

$C = B {<}{<}{<} 9$

… }

$V(\mathrm{i+1}) = ABCDEFGH \oplus V(\mathrm{i})$; }

-Output 256-bit Hash Value: $ABCDEFGH = V(\mathrm{n})$

China Cipher Algorithm – SM4

- SM4 algorithm

-The 1st commercial block cipher algorithm published by China in 2006. Formally known as SMS4.

-Similar to AES, symmetric cryptographic algorithm.

-Block size and key size are all 128-bit word.

-Consists of 32 identical rounds, comparing to 14 non-identical rounds for AES-256.

-Subsitution 'T' is used for each round.

It consists of non-linear substitution 'τ' and linear substitution 'L'. The ouput of τ is applied to the input of L, i.e. $T(.) = L(\tau(.))$.

-Non-linear substitution 'τ' applies 4 S-boxes in parallel

$B = \tau(Sbox(a_0), Sbox(a_1), Sbox(a_2), Sbox(a_3))$, where B is a 32-bit word. a_1, a_2, a_3 and a_4 are 8-bit bytes,

-Linear substitution 'L'

$C = L(B) = B\oplus (B<<<2)\oplus B(<<<10)\oplus(B<<<18)\oplus(B<<<24)$, C: 32-bit word

China Cipher Algorithm – SM4

-Key Expansion

Expand key with similar T structure to 32 round keys, rk_i, i = 0, …, 31

Decryption uses the same keys as encryption, but in reversed order.

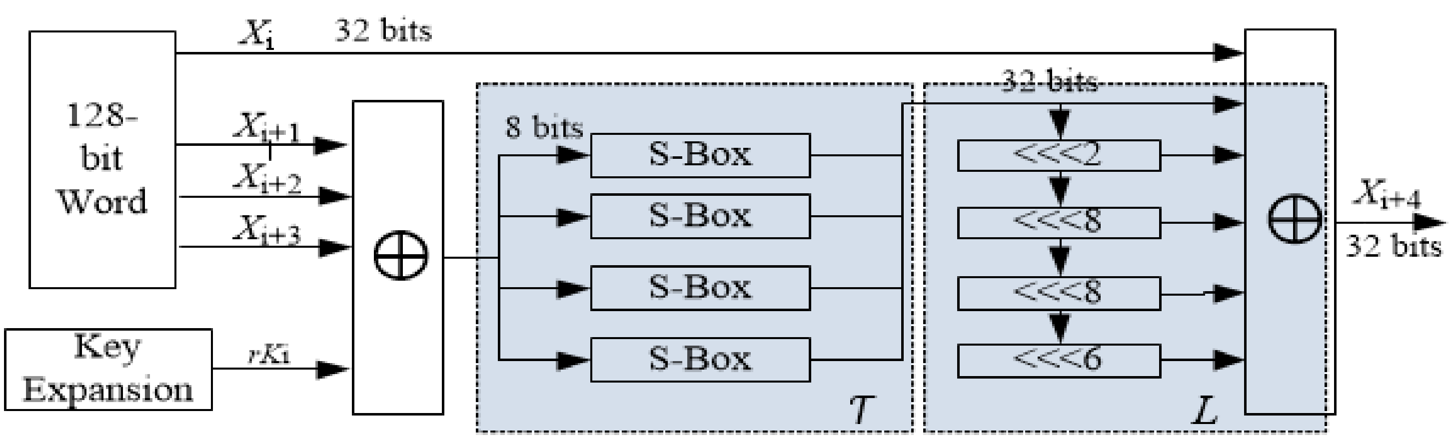
Encryption: $(rK_0, rK_1, \ldots, rK_{31})$

Decryption: $(rk_{31}, rK_{30}, \ldots, rK_0)$

-Round Function for encryption and decryption

$X_{i+4} = F(X_i, X_{i+1}, X_{i+2}, X_{i+3}, rk_i) = X_i \oplus T(X_{i+1} \oplus X_{i+2} \oplus X_{i+3} \oplus rk_i)$;

i = 0, 1, …, 31



32 rounds: $(x_0,x_1,x_2,x_3),(x_1,x_2,x_3,x_4),(x_2,x_3,x_4,x_5),\ldots,(x_{31},x_{32},x_{33},x_{34}) \rightarrow x_{35}$

-Output

$(Y_0,Y_1,Y_2,Y_3) = R(X_{32},X_{33},X_{34},X_{35})=(X_{35},X_{34},X_{33},X_{32})$, R: reverse order transform

China Cipher Algorithm

- SM2, SM3, and SM4 have been incorporated into TPM (Trusted Platform Module) 2.0.
- **O**ffice of the **S**tate **C**ommercial **C**ryptography **A**dministration (国家密码管理局商用密码管理办公室)

  website: http://www.oscca.gov.cn/

Thank You

Any questions?

please contact me at

Larry.ko@sage-micro.com