2018-2 연구 생활

2018. 12. 21

석사과정 안규황

HSU
HANSUNG
UNIVERSITY

Issue

Microsoft Q# 이란?

- 2017년 12월 11일 Microsoft에서 양자 개발 킷을 공개 함
  →Quantum Development Kit[1]
- Q#을 이용한다면 양자 컴퓨터가 아닌 일반 컴퓨터에서 양자 개발 할 수 있음
- 양자 컴퓨터는 코어 프레서서를 사용하고 GPU 혹은 FPGA를 프로그래밍 한 다음 CPU에서 가속도를 함
  →Q#은 이와 비슷하게 디자인 되었음
- Windows, linux, macOS 모두 제공하지만, linux에서 실험

HSU HANSUNG UNIVERSITY

[1] https://www.microsoft.com/en-us/quantum/development-kit

초 경량 블록암호 CHAM 구현_64x128

```
function ROL_64_128 (k : Int, x : Int) : (Int)
{
    return ((k <<< x) ||| (k >>> 16-x)) % 0x10000;
}

function ROL1_64_128 (k : Int) : (Int)
{
    return ((k <<< 1) ||| (k >>> 15)) % 0x10000;
}

function ROL8_64_128 (k : Int) : (Int)
{
    return ((k <<< 8) ||| (k >>> 8)) % 0x10000;
}

function ROR_64_128 (k : Int, x : Int) : (Int)
{
    return ((k >>> x) ||| ((k <<< 16-x))) % 0x10000;
}

function KeyExpansion_64_128(k : Int[]) : (Int[])
{
    mutable arr = new Int[16];

    for (i in 0..7){
        //Message(ToStringI(i));
        set arr[i] = k[i] ^^^ ROL_64_128(k[i], 1) ^^^ ROL_64_128(k[i], 8);
        set arr[(i+8) ^^^ 0x0001] = k[i] ^^^ ROL_64_128(k[i], 1) ^^^ ROL_64_128(k[i], 11);
        // Message(ToStringI(arr[(i+8) ^^^ 0x0001]));
        // Message(ToStringI((i+8) ^^^ 0x0001));
    }

    return arr;
}
```

```
function Encryption_64_128(pt : Int[], rk : Int[]) : (Int[]){
    mutable tmp = 0;
    mutable local_pt = new Int[4];
    mutable ct = new Int[4];

    for (i in 0..3){
        set local_pt[i] = pt[i];
    }

    for (i in 0..19){
        // Message(ToStringI(i));
        set tmp = ROL_64_128(local_pt[1], 1) ^^^ rk[(4*i) % 16];
        set ct[0] = ((local_pt[0] ^^^ (4*i)) + tmp) % 0x10000;
        set ct[0] = ROL_64_128(ct[0], 8);
        // Message(ToStringI(ct[0]));

        set tmp = ROL_64_128(local_pt[2], 8) ^^^ rk[(4*i+1) % 16];
        set ct[1] = ((local_pt[1] ^^^ (4*i+1)) + tmp) % 0x10000;
        set ct[1] = ROL_64_128(ct[1], 1);
        // Message(ToStringI(ct[1]));

        set tmp = ROL_64_128(local_pt[3], 1) ^^^ rk[(4*i+2) % 16];
        set ct[2] = ((local_pt[2] ^^^ (4*i+2)) + tmp) % 0x10000;
        set ct[2] = ROL_64_128(ct[2], 8);
        // Message(ToStringI(ct[2]));

        set tmp = ROL_64_128(ct[0], 8) ^^^ rk[(4*i+3) % 16];
        set ct[3] = ((local_pt[3] ^^^ (4*i+3)) + tmp) % 0x10000;
        set ct[3] = ROL_64_128(ct[3], 1);
        // Message(ToStringI(ct[3]));

        set local_pt[0] = ct[0];
        set local_pt[1] = ct[1];
        set local_pt[2] = ct[2];
        set local_pt[3] = ct[3];
    }

    return ct;
}
```

HANSUNG UNIVERSITY

https://github.com/kyu-h/QS_CHAM

2018-2 연구실적_논문

- 게재

· 효율적인 한국 암호 모듈 검증 제도를 위한 암호 모듈 자동 검증 시스템, 한국정보보호학회 동계학술대회, 2018.

· OpenMP를 활용한 LSH DRBG 병렬 최적 구현, 한국정보보호학회 동계학술대회, 2018.

· 국산 암호 알고리즘 부채널 분석에 대한 고찰, 한국정보보호학회 동계학술대회, 2018.

HSU
HANSUNG
UNIVERSITY

2018-2 연구실적_논문

- 작성 완료

· Web Assembly를 활용한 AES 구현

· Web Assembly를 활용한 CHAM 구현

· LINK 블록체인을 블랙박스에 적용한 교통사고 검증 서비스

- 작성 중

· QS을 활용한 CHAM 구현

HSU
HANSUNG
UNIVERSITY

2018-2 연구실적_대외 활동 및 공모전

- 대외 활동

· 국가보안기술연구소 전문인력 양성 수료

- 공모전

· 제 5회 ICT 공모전 우수상

· 연구산업 신서비스 분야 공모전 우수상

· 2018 서울대 의대 정보 의학 해커톤 장려상

· 라인 블록체인 공모전

(금일 참가 → 결승 진출 팀 최소 장려상 지급, 따라서 최소 장려상)

HSU
HANSUNG
UNIVERSITY

Thank You!

HSU
HANSUNG
UNIVERSITY