공개키 암호의 구현

Part 2.Ep 5: Rabin 구현
YouTube: https://youtu.be/iyx0EG_AMYA
Git: https://github.com/minpie/CryptoCraftLab-minpie_public

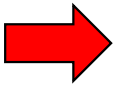
한성대학교 HANSUNG UNIVERSITY

CryptoCraft LAB

발표 계획 목록

Rabin C언어 구현

발표 계획: 24.07.19ver

- Part 1. 대칭키 암호 단일블록 C언어 구현
  - Ep1. AES
  - Ep2. DES
- Part 2. 64비트 이상 키 길이의 공개키 암호 C언어 구현
  - Ep3. GMP 라이브러리
  - Ep4. RSA 구현
  - Ep5. Rabin 구현 

  - Ep6. Elgamal 구현
  - Ep7. ECDSA 구현
- Part 3. AES-운영모드 with 병렬컴퓨팅
  - Ep8. OpenMPI 라이브러리
  - Ep9. OpenMPI-AES
  - Ep10. CUDA C
  - Ep11. CUDA-AES

Rabin C언어 구현 - 개요

Rabin cryptosystem

13 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

This article is about the textbook public-key encryption scheme. For the digital signature scheme it was based on, see Rabin signature.

The **Rabin cryptosystem** is a family of public-key encryption schemes based on a trapdoor function whose security, like that of RSA, is related to the difficulty of integer factorization.[1][2]

- 기본적인 수준으로 Rabin 암호를 구현.
- 기반문제: 인수분해 문제

Rabin C언어 구현 – 전체 흐름

```
75  int main(void)
76  {
77      mpz_t p, q, n, plain, cipher;
78      mpz_t plain2[4];
79      mpz_inits(p, q, n, plain, cipher, NULL);
80      mpz_inits(plain2[0], plain2[1], plain2[2], plain2[3], NULL);
81
82      // test:
83      mpz_set_ui(p, 23);
84      mpz_set_ui(q, 7);
85      mpz_set_ui(plain, 24);
86
87      mpz_mul(n, p, q); // n = p * q
88      Encrypt(cipher, plain, n);
89      Decrypt(plain2, cipher, p, q);
90
91      printf("Encryption:\n");
92      gmp_printf("P=%Zd, C=%Zd\n", plain, cipher);
93
94      printf("Decryption:\n");
95      gmp_printf("C=%Zd\n", cipher);
96      gmp_printf("P1=%Zd\n", plain2[0]);
97      gmp_printf("P2=%Zd\n", plain2[1]);
98      gmp_printf("P3=%Zd\n", plain2[2]);
99      gmp_printf("P4=%Zd\n", plain2[3]);
100
101     mpz_clears(p, q, n, plain, cipher, NULL);
102     mpz_clears(plain2[0], plain2[1], plain2[2], plain2[3], NULL);
103
104     return 0;
105 }
```

- 암호화-복호화 과정을 위한 main() 코드

Rabin C언어 구현 – Encrypt()

```
36  void Encrypt(mpz_t cipher, mpz_t plain, mpz_t n)
37  {
38      mpz_powm_ui(cipher, plain, 2, n);
39  }
```

- 암호화 연산

Rabin C언어 구현 – Decrypt()

```
41  void Decrypt(mpz_t plains[4], mpz_t cipher, mpz_t p, mpz_t q)
42  {
43      mpz_t exp1, exp2, tmp1, tmp2, a1, a2, b1, b2;
44      mpz_inits(exp1, exp2, tmp1, tmp2, a1, a2, b1, b2, NULL);
45      mpz_add_ui(tmp1, p, 1);       // tmp1 = p + 1
46      mpz_add_ui(tmp2, q, 1);       // tmp2 = q + 1
47      mpz_fdiv_q_ui(exp1, tmp1, 4); // exp1 = tmp1 / 4 = (p+1) / 4
48      mpz_fdiv_q_ui(exp2, tmp2, 4); // exp2 = tmp2 / 4 = (q+1) / 4
49
50      // a1:
51      mpz_powm(a1, cipher, exp1, p); // a1 = (cipher ** exp1) mod p = (cipher ** ((p+1) / 4)) mod p
52
53      // a2:
54      mpz_powm(a2, cipher, exp1, p); // a2 = (cipher ** exp1) mod p = (cipher ** ((p+1) / 4)) mod p
55      mpz_mul_si(a2, a2, -1);       // a2 = -(a2)
56      mpz_mod(a2, a2, p);
57
58      // b1:
59      mpz_powm(b1, cipher, exp2, q); // b1 = (cipher ** exp2) mod p = (cipher ** ((q+1) / 4)) mod q
60
61      // b2:
62      mpz_powm(b2, cipher, exp2, q); // b2 = (cipher ** exp2) mod p = (cipher ** ((q+1) / 4)) mod q
63      mpz_mul_si(b2, b2, -1);       // b2 = -(b2)
64      mpz_mod(b2, b2, q);
65
66      // get plains:
67      GetChineseRemainderTheorem_num_2(plains[0], a1, b1, p, q);
68      GetChineseRemainderTheorem_num_2(plains[1], a1, b2, p, q);
69      GetChineseRemainderTheorem_num_2(plains[2], a2, b1, p, q);
70      GetChineseRemainderTheorem_num_2(plains[3], a2, b2, p, q);
71
72      mpz_clears(exp1, exp2, tmp1, tmp2, a1, a2, b1, b2, NULL);
73  }
```

- 복호화 연산
- CRT를 이용해 계산
- 해(평문)가 4개가 생성됨

Rabin C언어 구현 – CRT 계산

```
4   void GetChineseRemainderTheorem_num_2(mpz_t result, mpz_t a1, mpz_t a2, mpz_t m1, mpz_t m2)
5   {
6       mpz_t common_m, m1_div_common_m, m2_div_common_m, m1_div_common_m_inv, m2_div_common_m_inv;
7       mpz_t tmp1, tmp2;
8       mpz_inits(common_m, m1_div_common_m, m2_div_common_m, m1_div_common_m_inv, m2_div_common_m_inv, NULL);
9       mpz_inits(tmp1, tmp2, NULL);
10
11      // get common_m:
12      mpz_mul(common_m, m1, m2);
13
14      // get m1_div_common_m:
15      mpz_fdiv_q(m1_div_common_m, common_m, m1);
16      // get m2_div_common_m:
17      mpz_fdiv_q(m2_div_common_m, common_m, m2);
18
19      // get m1_div_common_m_inv:
20      mpz_invert(m1_div_common_m_inv, m1_div_common_m, m1);
21      // get m2_div_common_m_inv:
22      mpz_invert(m2_div_common_m_inv, m2_div_common_m, m2);
23
24      // get result:
25      mpz_mul(tmp1, a1, m1_div_common_m);
26      mpz_mul(tmp1, tmp1, m1_div_common_m_inv);
27      mpz_mul(tmp2, a2, m2_div_common_m);
28      mpz_mul(tmp2, tmp2, m2_div_common_m_inv);
29      mpz_add(result, tmp1, tmp2);
30      mpz_mod(result, result, common_m);
31
32      mpz_clears(common_m, m1_div_common_m, m2_div_common_m, m1_div_common_m_inv, m2_div_common_m_inv, NULL);
33      mpz_clears(tmp1, tmp2, NULL);
34  }
```

- CRT를 푸는 함수

Rabin C언어 구현 – 실행 결과

```
watermark@watermarkserver:/storage/drive1/pt1/codes/C/RabinCryptosystem$ ./main
Encryption:
P=24, C=93
Decryption:
C=93
P1=116
P2=24
P3=137
P4=45
watermark@watermarkserver:/storage/drive1/pt1/codes/C/RabinCryptosystem$
```

- 암호문은 1개
- 평문은 4개

Rabin C언어 구현 – 참고문헌

- https://en.wikipedia.org/wiki/Rabin_cryptosystem
- https://personal.utdallas.edu/~mxk055100/courses/crypto09s_files/rabin-overview.pdf

Q & A