ERROR REPORT

LWE Implementations

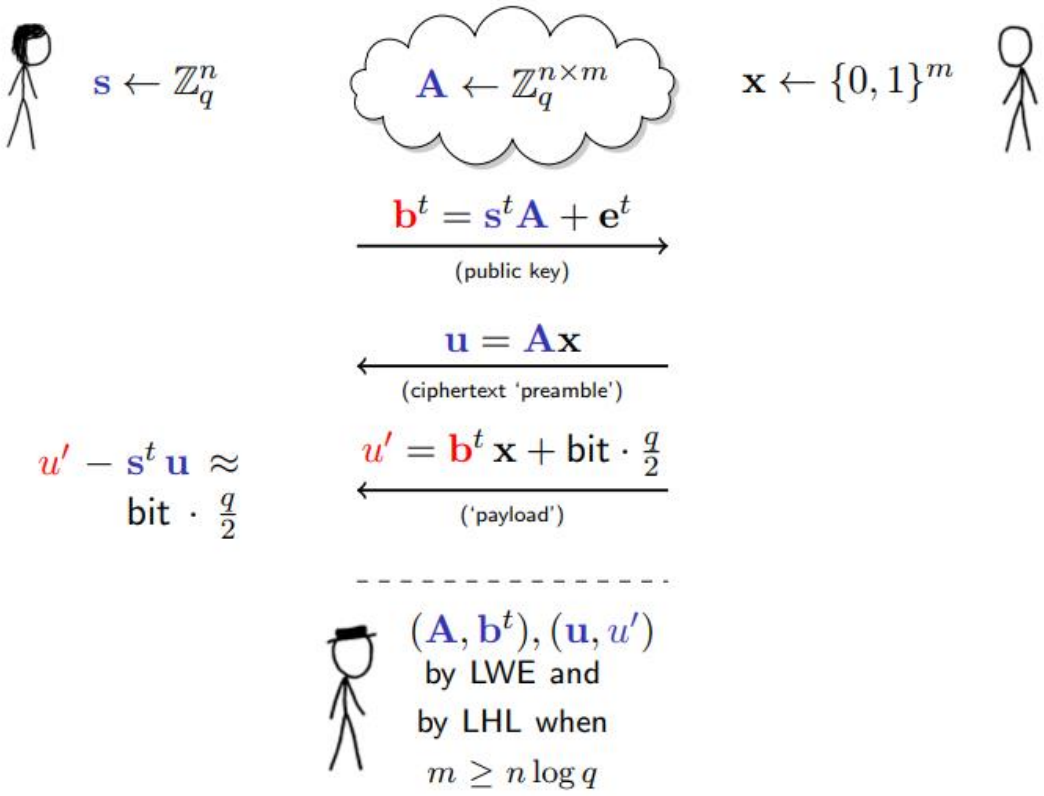
Abstract

Includes the issues found in Public Key Cryptosystem and Dual Cryptosystem implementations.

Contents

1.0 Public Key Cryptosystem

1.1 Notation and Operation



Source: Lattice-Based Crypto & Applications, Bar-Ilan University, Israel 2012

1.2 Issue

For the public-key cryptosystem when using q < 2^{16} (65 536), when encrypting and decrypting bit 0, the accuracy is around 90% to 100%. For bit 1, 100% accuracy is achieved.

1.3 Observations for bit 0

For q = 1024 out of 1000 tests,

Encryption and Decryption work 95.6% of the time.

Max eTx = 187

Max bTxMAX = 1023

Error instances = 44

1.3.1 Scenario 1

When “bTx” is close to “q” and “sTu mod q” is close to 0, the results are wrong.

Ex:-

1.

[DEBUG] bTx: 1020

[DEBUG] sTu before mod: 920789003

[DEBUG] sTu mod q: 11

[DEBUG] u_: 1020

[DEBUG] eTx: -15

[DEBUG] recovered: 1

[DEBUG] u_ - sTu: 1009

2.

[DEBUG] bTx: 984

[DEBUG] sTu before mod: 1088840713

[DEBUG] sTu mod q: 9

[DEBUG] u_: 984

[DEBUG] eTx: -49

[DEBUG] recovered: 1

[DEBUG] u_ - stu: 975

When encrypting: $u' = bTx + bit \cdot q/2$

When bit = 0, $u' = bTx$

When decrypting $u' = bTx$,

$bit \cdot q/2 + eTx = u' - sTu = bTx - sTu$

When “bTx” is close to “q” and “sTu mod q” is close to 0, the above value can be greater than q/2. Therefore it is decrypted to bit 1.

1.3.2 Scenario 2

When “bTx” is close to 0 and “sTu mod q” is close to “q”, the results are wrong.

1.

[DEBUG] bTx: 16

[DEBUG] sTu before mod: 1199177704

[DEBUG] sTu mod q: 1000

[DEBUG] u_: 16

[DEBUG] eTx: 40

[DEBUG] recovered: 1

[DEBUG] u_ - stu: -984

2.

[DEBUG] bTx: 4

[DEBUG] sTu before mod: 1333685190

[DEBUG] sTu mod q: 966

[DEBUG] u_: 4

[DEBUG] eTx: 62

[DEBUG] recovered: 1

[DEBUG] u_ - stu: -962

When encrypting: $u' = bTx + bit \cdot q/2$

When bit = 0, $u' = bTx$

When decrypting $u' = bTx$,

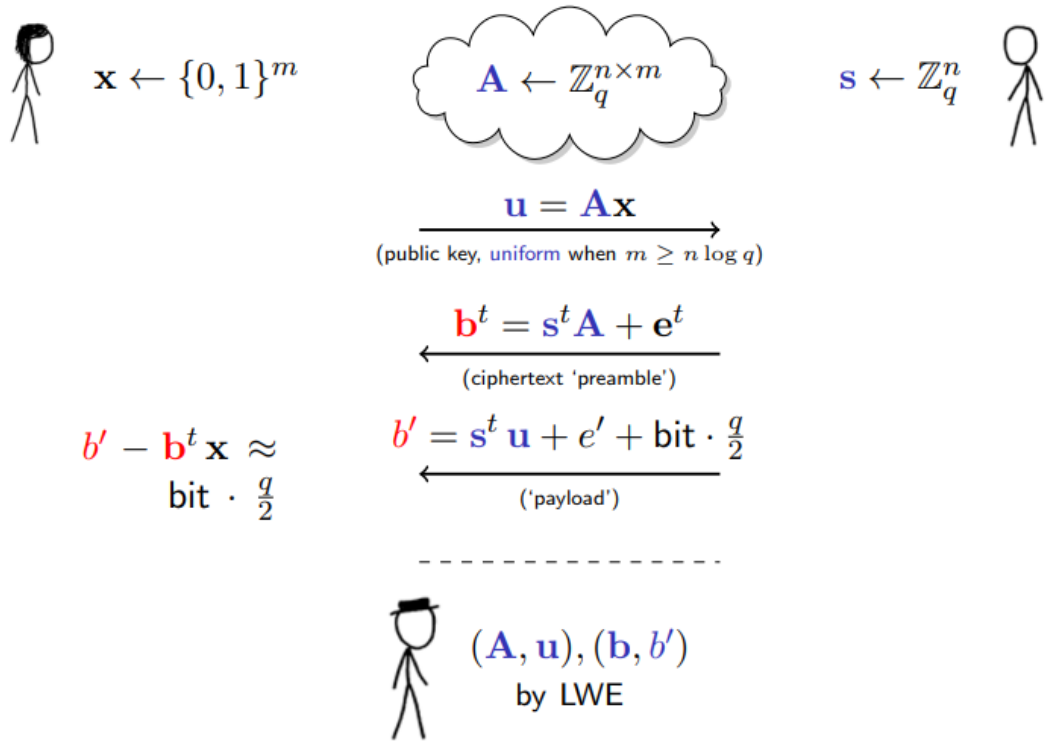
$bit \cdot q/2 + eTx = u' - sTu = bTx - sTu$

When “bTx” is close to 0 and “sTu mod q” is close to “q”, the above value can be smaller than -q/2. Therefore it is decrypted to bit 1 because the absolute value of “$u' - sTu$” > $bit \cdot q/2$.

Because of these 2 reasons encryption and decryption of bit 0 give inaccurate results.

2.0 Dual Cryptosystem

2.1 Notation and Operation



Source: Lattice-Based Crypto & Applications, Bar-Ilan University, Israel 2012

2.2 Issue

For the public-key cryptosystem when using q < 150 000, when encrypting and decrypting bit 0, the accuracy is around 90% to 100%. For bit 1, 100% accuracy is achieved.

2.3 Observations for bit 0

For q = 1024 out of 1000 tests,

Encryption and Decryption work 96.2% of the time.

2.3.1 Scenario 1

When “ b’ ” is close to “q” and “bTx mod q” is close to 0, the results are wrong.

Ex:-

1.

[DEBUG] b' = sTu + e': 1013

[DEBUG] bTx before mod: 63496

[DEBUG] bTx mod q: 8

[DEBUG] b'-bTx: 1005

[DEBUG] eTx: -19

2.

[DEBUG] b' = sTu + e': 974

[DEBUG] bTx before mod: 66590

[DEBUG] bTx mod q: 30

[DEBUG] b'-bTx: 944

[DEBUG] eTx: -80

When encrypting: $b' = sTu + e' + bit \cdot q/2$

When bit = 0, $b' = sTu + e'$

When decrypting $b' = sTu + e'$,

$bit \cdot q/2 \approx b' - bTx$

When “ b’ ” is close to “q” and “bTx mod q” is close to 0, the above value can be greater than q/2. Therefore it is decrypted to bit 1.

2.3.2 Scenario 2

When “bTx” is close to 0 and “sTu mod q” is close to “q”, the results are wrong.

Ex:-

1.

[DEBUG] b' = sTu + e': 6

[DEBUG] bTx before mod: 60415

[DEBUG] bTx mod q: 1023

[DEBUG] b'-bTx: -1017

[DEBUG] eTx: 7

iteration = 798

2.

[DEBUG] b' = sTu + e': 7

[DEBUG] bTx before mod: 64500

[DEBUG] bTx mod q: 1012

[DEBUG] b'-bTx: -1005

[DEBUG] eTx: 19

iteration = 854

When encrypting: $b' = sTu + e' + bit \cdot q/2$

When bit = 0, $b' = sTu + e'$

When decrypting $b' = sTu + e'$,

$bit \cdot q/2 \approx b' - bTx$

When “ b’ ” is close to 0 and “bTx mod q” is close to “q”, the above value can be lesser than -q/2. Therefore it is decrypted to bit 1 because the absolute value of “$b' - bTx$” > $bit \cdot q/2$.

Because of these 2 reasons encryption and decryption of bit 0 give inaccurate results.