Lattice-Based Public-Key Encryption

Dr. Essam Ghadafi

CyBOK Mapping

The lecture maps to the following CyBOK Knowledge Areas:

- Systems Security $\rightarrow$ Cryptography
- Infrastructure Security $\rightarrow$ Applied Cryptography

Outline

- IND-CPA vs IND-CCA Security for Public-Key Encryption
  - A reminder
- Regev's Encryption Scheme
  - Overview and mechanism of encryption
  - Security of the scheme
  - Extending the Message Space
- Kyber Encryption
  - Overview and mechanism of encryption
  - Security & Efficiency of the scheme

Public-Key Encryption – Syntax

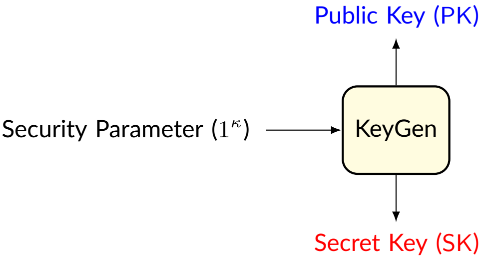

Figure: Key Generation: $(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{KeyGen}(1^\kappa)$

Public-Key Encryption – Syntax

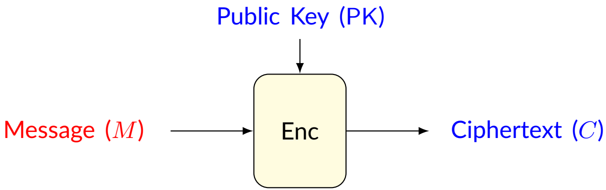

Figure: Encryption ($C \leftarrow \mathsf{Enc}(\mathsf{PK}, M)$)

Public-Key Encryption – Syntax

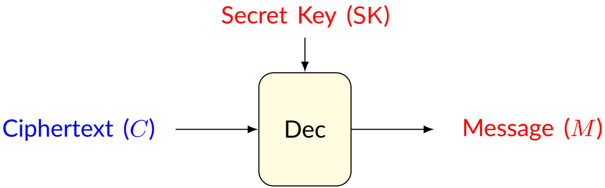

Figure: Decryption (M $=$ Dec(SK, C))

Correctness of Public-Key Encryption

Correctness: For all security parameters κ, for all key pairs $(\mathsf{PK}, \mathsf{SK})$ from KeyGen and all messages M:

$$C \leftarrow \mathsf{Enc}(\mathsf{PK}, M), \quad \mathsf{Dec}(\mathsf{SK}, C) = M$$

PKE Security – Intuition

Indistinguishability under Chosen-Plaintext Attack (IND-CPA):
Adversary sees ciphertexts of chosen messages, but cannot tell which message was encrypted

PKE Security – Intuition

Indistinguishability under Chosen-Plaintext Attack (IND-CPA):
Adversary sees ciphertexts of chosen messages, but cannot tell which message was encrypted

Indistinguishability under Chosen-Ciphertext Attack (IND-CCA):
Stronger: Adversary can also query a decryption oracle on any ciphertext (except the challenge), yet still cannot break it

IND-CPA for Public-Key Encryption

Indistinguishability under Chosen-Plaintext Attack (IND-CPA)

PK ←

Attacker

Challenge messages M_0, M_1 →

$C^* = \mathsf{Enc}(\mathsf{PK}, M_b)$ for $b \leftarrow \{0,1\}$ ←

Guess b^* →

Challenger

$(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{KeyGen}(1^\kappa)$

The attacker's advantage is given by:

$$\left| \Pr[b^* = b] - \frac{1}{2} \right|$$

The scheme is *IND-CPA* secure if the advantage is negligible for all efficient attackers

IND-CCA for Public-Key Encryption

Indistinguishability under Chosen-Ciphertext Attack (IND-CCA) is defined similarly to IND-CPA, except the attacker is allowed to request the decryption of any ciphertext other than C^*

Regev's PKE

Regev's PKE [3] is based on the Decisional LWE (D-LWE) assumption

Key Generation: This exactly the input generation in LWE

- Choose a vector $\mathbf{s} \in \mathbb{Z}_q^n$ uniformly at random
- Sample public matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ uniformly at random
- Sample noise $\mathbf{e} \in \chi^m$
- Let $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod q$
- Set PK $= \mathbf{A}' = [\mathbf{A} \mid \mathbf{b}] \in \mathbb{Z}_q^{m \times (n+1)}$
- Set SK $= \mathbf{s} \in \mathbb{Z}_q^n$

Regev's PKE

Intuition: One can think of $\mathbf{b}$ part of PK as m secret-key encryptions (using the secret key SK $= \mathbf{s}$) of the message 0

- A secret-key encryption scheme can be constructed by moving the sampling of $\mathbf{A}$ and $\mathbf{e}$ to the encryption process

Note: By $[\mathbf{A}_1|\mathbf{A}_2]$, we denote the concatenation of both matrices as columns

Regev's PKE

Encryption: Encrypting a message $x \in \{0,1\}$ using PK $= \mathbf{A}'$

- Represent message $x \in \{0,1\}$ as $\tilde{x} = x \cdot \lfloor \frac{q}{2} \rfloor$
- Choose a random vector $\mathbf{r} \in \{0,1\}^m$
- Compute ciphertext: $\mathbf{c} = \mathbf{r}^T \mathbf{A}' + (\mathbf{0}^n, \tilde{x}) \pmod{q}$

Regev's PKE

Encryption: Encrypting a message $x \in \{0,1\}$ using PK $= \mathbf{A}'$

- Represent message $x \in \{0,1\}$ as $\tilde{x} = x \cdot \lfloor \frac{q}{2} \rfloor$
- Choose a random vector $\mathbf{r} \in \{0,1\}^m$
- Compute ciphertext: $\mathbf{c} = \mathbf{r}^T \mathbf{A}' + (\mathbf{0}^n, \tilde{x}) \pmod q$

Decryption: Decrypting ciphertext $\mathbf{c}$ using secret key SK $= \mathbf{s}$

- Compute
  $$\breve{x} = \mathbf{c} \begin{bmatrix} \mathbf{s} \\ -1 \end{bmatrix} = \mathbf{r}^T(\mathbf{A}\mathbf{s} - \mathbf{b}) - \tilde{x} \pmod q = -\mathbf{r}^T\mathbf{e} - \tilde{x} \pmod q$$
- If the noise vector used is small ($\|\mathbf{e}\|_1 < \frac{q}{4}$), we can recover the plaintext x from $\breve{x}$ as follows:
  - 0 if the result of decryption is close to 0
  - 1 if the result of decryption is close to $\lfloor \frac{q}{2} \rfloor$

Regev's PKE

Encryption: Encrypting a message $x \in \{0,1\}$ using PK $= \mathbf{A}'$

- Represent message $x \in \{0,1\}$ as $\tilde{x} = x \cdot \lfloor \frac{q}{2} \rfloor$
- Choose a random vector $\mathbf{r} \in \{0,1\}^m$
- Compute ciphertext: $\mathbf{c} = \mathbf{r}^T \mathbf{A}' + (\mathbf{0}^n, \tilde{x}) \pmod q$

Decryption: Decrypting ciphertext $\mathbf{c}$ using secret key SK $= \mathbf{s}$

- Compute
$$\breve{x} = \mathbf{c} \begin{bmatrix} \mathbf{s} \\ -1 \end{bmatrix} = \mathbf{r}^T(\mathbf{A}\mathbf{s} - \mathbf{b}) - \tilde{x} \pmod q = -\mathbf{r}^T \mathbf{e} - \tilde{x} \pmod q$$
- If the noise vector used is small ($\|\mathbf{e}\|_1 < \frac{q}{4}$), we can recover the plaintext x from $\breve{x}$ as follows:
  - 0 if the result of decryption is close to 0
  - 1 if the result of decryption is close to $\lfloor \frac{q}{2} \rfloor$

Correctness of the scheme is easy to check

Security of Regev's PKE — Part 1

Theorem

Regev's PKE is IND-CPA Secure if D-LWE problem is hard

Security of Regev's PKE — Part 1

Theorem

Regev's PKE is IND-CPA Secure if D-LWE problem is hard

Prof Sketch

- Game_0: The real IND-CPA game where PK is normal, i.e. $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod q$

Security of Regev's PKE — Part 1

Theorem

Regev's PKE is IND-CPA Secure if D-LWE problem is hard

Prof Sketch

- Game_0: The real IND-CPA game where PK is normal, i.e. $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod{q}$
- Game_1: We replace $\mathbf{b}$ part of PK with a random vector $\mathbf{b} \in \mathbb{Z}_q^m$
  - PK is uniformly random in $\mathbb{Z}_q^{m\times(n+1)}$ and is independent of SK

Security of Regev's PKE — Part 1

Theorem

Regev's PKE is IND-CPA Secure if D-LWE problem is hard

Prof Sketch

- Game$_0$: The real IND-CPA game where PK is normal, i.e. $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod{q}$

- Game$_1$: We replace $\mathbf{b}$ part of PK with a random vector $\mathbf{b} \in \mathbb{Z}_q^m$
  - PK is uniformly random in $\mathbb{Z}_q^{m \times (n+1)}$ and is independent of SK

- Game$_2$: Same as Game$_1$, but challenge ciphertext $\mathbf{c}_b$ is chosen uniformly at random from $\mathbb{Z}_q^{n+1}$
  - $\mathbf{c}_b$ is now independent of $x_b \Rightarrow$ the attacker's advantage in guessing the bit b is exactly $\frac{1}{2}$

Security of Regev's PKE — Part 2

- Claim$_1$: Game$_0 \approx_c$ Game$_1$, i.e. they are computationally indistinguishable by the D-LWE assumption
- Claim$_2$: Game$_1 \approx_s$ Game$_2$, i.e. they are statistically indistinguishable by the Leftover Hash Lemma

Extending Regev's PKE Message Space

Regev's PKE originally supports encryption of bits, i.e., the message space is $\mathbb{Z}_2$

Extending Regev's PKE Message Space

Regev's PKE originally supports encryption of bits, i.e., the message space is $\mathbb{Z}_2$

To extend to $\mathbb{Z}_p$ (for some prime p), we modify the encoding as follows:

- Change encoding from $\tilde{x} = x \cdot \lfloor \frac{q}{2} \rfloor$ to $\tilde{x} = x \cdot \lfloor \frac{q}{p} \rfloor$
- Decrypt by rounding $\tilde{x}$ to the nearest multiple of $\frac{q}{p}$

This works if $\|\mathbf{e}\|_\infty < \frac{q}{2p}$

Kyber Encryption

Kyber [1], which was standardized by NIST as ML-KEM (FIPS 203) [2], is a lattice-based key-encapsulation mechanism

IND-CCA secure and relies on Decisional M-LWE (D-M-LWE)

3 Security Levels:

- Kyber-512: Equivalent to AES-128, i.e. $k = 2$
- Kyber-768: Equivalent to AES-192, i.e. $k = 3$
- Kyber-1024: Equivalent to AES-256, i.e. $k = 4$

Kyber IND-CPA Public-Key Encryption

Let $R_q = \mathbb{Z}_q[X]/(X^{256}+1)$, $q = 3329$, Kyber security level is parametrised by $k \in \{2, 3, 4\}$
Also, we require two distributions χ_e and χ_s over R_q

Kyber IND-CPA Public-Key Encryption

Let $R_q = \mathbb{Z}_q[X]/(X^{256}+1)$, $q = 3329$, Kyber security level is parametrised by $k \in \{2,3,4\}$
Also, we require two distributions χ_e and χ_s over R_q

- Key Generation:
  - Sample a uniform matrix $\mathbf{A} \in R_q^{k \times k}$
  - Sample secret vector $\mathbf{s} \in R_q^k$ according to χ_s
  - Sample error vector $\mathbf{e} \in R_q^k$ according to χ_e
  - Secret key: SK $= \mathbf{s} \in R_q^k$
  - Public key: PK $= (\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}) \in R_q^{k \times k} \times R_q^k$

Kyber IND-CPA Public-Key Encryption

- Encryption of a message $\mathbf{m} \in \{0,1\}^n$:
  - Encode $\mathbf{m}$ as polynomial $m(X)$ (with $0/1$ coefficients) in R_q
    - Encode $\mathbf{m} = (m_0, m_1, \ldots, m_{n-1}) \in \{0,1\}^n$ as the polynomial $m(X) = \sum_{i=0}^{n-1} m_i X^i \ \in \ R_q$
  - Sample $\mathbf{r} \in R_q^k$ according to χ_s
  - Sample $\mathbf{e}_1 \in R_q^k, e_2 \in R_q$ according to χ_e
  - Compute ciphertext:

$$\mathbf{u} = \mathbf{A}^T\mathbf{r} + \mathbf{e}_1, \quad v = \mathbf{b}^T\mathbf{r} + e_2 + \left\lfloor \frac{q}{2} \right\rfloor m(X)$$

  - Ciphertext: $c = (\mathbf{u}, v) \in R_q^k \times R_q$

Kyber IND-CPA Public-Key Encryption

- Encryption of a message $\mathbf{m} \in \{0,1\}^n$:
  - Encode $\mathbf{m}$ as polynomial $m(X)$ (with $0/1$ coefficients) in R_q
    - Encode $\mathbf{m} = (m_0, m_1, \ldots, m_{n-1}) \in \{0,1\}^n$ as the polynomial $m(X) = \sum_{i=0}^{n-1} m_i X^i \in R_q$
  - Sample $\mathbf{r} \in R_q^k$ according to χ_s
  - Sample $\mathbf{e}_1 \in R_q^k, e_2 \in R_q$ according to χ_e
  - Compute ciphertext:

$$\mathbf{u} = \mathbf{A}^T \mathbf{r} + \mathbf{e}_1, \quad v = \mathbf{b}^T \mathbf{r} + e_2 + \left\lfloor \frac{q}{2} \right\rfloor m(X)$$

  - Ciphertext: $c = (\mathbf{u}, v) \in R_q^k \times R_q$

- Decryption:
  - Compute $v - \mathbf{s}^T \mathbf{u} \approx \left\lfloor \frac{q}{2} \right\rfloor m(X)$, then recover $\mathbf{m} \in \{0,1\}^n$ from the coefficients of $m(X)$ by thresholding around $\frac{q}{2}$

$$m_i = \begin{cases} 0, & \text{if coefficient } c_i \text{ is closer to } 0 \text{ than to } \frac{q}{2} \pmod{q}, \\ 1, & \text{if coefficient } c_i \text{ is closer to } \frac{q}{2} \text{ than to } 0 \pmod{q}. \end{cases}$$

Security of IND-CPA Kyber PKE

The scheme is IND-CPA secure if the Decisional M-LWE (D-M-LWE$_{q,k+1,k,\chi_s,\chi_e}$) problem is hard

Security of IND-CPA Kyber PKE

The scheme is IND-CPA secure if the Decisional M-LWE (D-M-LWE$_{q,k+1,k,\chi_s,\chi_e}$) problem is hard

To obtain IND-CCA security, one applies the **Fujisaki–Okamoto transformation**, which transforms any IND-CPA PKE into an IND-CCA secure PKE

Kyber IND-CCA KEM Efficiency

Table: Public key and ciphertext sizes for NIST-standardized Kyber IND-CCA KEMs

Variant	PK **(bytes)**	**Ciphertext (bytes)**
Kyber-512 (128-bit security)	800	768
Kyber-768 (192-bit security)	1184	1088
Kyber-1024 (256-bit security)	1568	1568

Key Takeaways

- Regev's Encryption is based on the Decisional LWE assumption
  - The message space can be extended to allow more efficient encoding of messages

- Kyber Encryption is based on the Decisional Module-LWE assumption
  - Efficient, IND-CCA secure, and comes in 3 security levels
  - Standardised by NIST

References

[1] J. Bos, et al. CRYSTALS - Kyber: A CCA-Secure Module-Lattice-Based KEM. IEEE European Symposium on Security and Privacy (EuroS&P), 353-367, 2018.

[2] National Institute of Standards and Technology. Module-Lattice-Based Key-Encapsulation Mechanism Standard. (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 203, 2024. Available: https://doi.org/10.6028/NIST.FIPS.203.

[3] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. J. ACM, 56(6), Sep 2009.