BILL, RECORD LECTURE!!!!

BILL RECORD LECTURE!!!

Public Key LWE Cipher

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Alice Wants to Send $b \in \{0, 1\}$.

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Alice Wants to Send $b \in \{0, 1\}$.

1. Alice picks random set $\vec{r}$.

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Alice Wants to Send $b \in \{0, 1\}$.

1. Alice picks random set $\vec{r}$.
2. Alice computes $C \equiv \vec{r} \cdot \vec{k}$ and $e \in^r \{-\gamma, \ldots, \gamma\}$.

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Alice Wants to Send $b \in \{0, 1\}$.

1. Alice picks random set $\vec{r}$.
2. Alice computes $C \equiv \vec{r} \cdot \vec{k}$ and $e \in^r \{-\gamma, \ldots, \gamma\}$.
3. To send b Alice sends $(\vec{r}; D)$ where $D \equiv C + e + \frac{bp}{4}$.

Recall Private Key LWE Cipher

Private Key $\vec{k}$. Both Alice and Bob have this.
Public Info p, the mod. All math is mod p. Params γ, n.

Alice Wants to Send $b \in \{0, 1\}$.

1. Alice picks random set $\vec{r}$.
2. Alice computes $C \equiv \vec{r} \cdot \vec{k}$ and $e \in^r \{-\gamma, \ldots, \gamma\}$.
3. To send b Alice sends $(\vec{r}; D)$ where $D \equiv C + e + \frac{bp}{4}$.
4. Bob computes $\vec{r} \cdot \vec{k} \equiv C$. If $D \sim C$, $b = 0$, else $b = 1$.

Thoughts on a Public Key LWE Cipher

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.
  In public key, **only** Alice has the key $\vec{k}$.

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.
  In public key, **only** Alice has the key $\vec{k}$.
- Alice **Cannot** publish key $\vec{k}$.

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.
  In public key, **only** Alice has the key $\vec{k}$.
- Alice **Cannot** publish key $\vec{k}$.
  Alice **Can** publishes noisy equations that $\vec{k}$ satisfies.

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.
  In public key, **only** Alice has the key $\vec{k}$.
- Alice **Cannot** publish key $\vec{k}$.
  Alice **Can** publishes noisy equations that $\vec{k}$ satisfies.
  Eve won't be able to use the noisy equations to find key.

Thoughts on a Public Key LWE Cipher

- In private key, **both** Alice and Bob have $\vec{k}$.
  In public key, **only** Alice has the key $\vec{k}$.
- Alice **Cannot** publish key $\vec{k}$.
  Alice **Can** publishes noisy equations that $\vec{k}$ satisfies.
  Eve won't be able to use the noisy equations to find key.
  How can Bob use the noisy equations to encode a bit?

Recall: Noisy Equations

Everything is mod p, some prime p.

Recall: Noisy Equations

Everything is mod p, some prime p.

Let $\vec{k} = (k_1, \ldots, k_n)$, $\vec{r} = (r_1, \ldots, r_n)$, and C be such that

$$r_1 k_1 + \cdots + r_n k_n = C$$

Recall: Noisy Equations

Everything is mod p, some prime p.

Let $\vec{k} = (k_1, \ldots, k_n)$, $\vec{r} = (r_1, \ldots, r_n)$, and C be such that

$$r_1 k_1 + \cdots + r_n k_n = C$$

$r_1 x_1 + \cdots + r_n x_n = C$ is an **equation** that $\vec{k}$ satisfies.

Recall: Noisy Equations

Everything is mod p, some prime p.

Let $\vec{k} = (k_1, \ldots, k_n)$, $\vec{r} = (r_1, \ldots, r_n)$, and C be such that

$$r_1 k_1 + \cdots + r_n k_n = C$$

$r_1 x_1 + \cdots + r_n x_n = C$ is an **equation** that $\vec{k}$ satisfies.

Pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Think of γ as small.

$r_1 x_1 + \cdots + r_n x_n \sim C + e$ is **noisy eq** that $\vec{k}$ satisfies.

Recall: Noisy Equations

Everything is mod p, some prime p.

Let $\vec{k} = (k_1, \ldots, k_n)$, $\vec{r} = (r_1, \ldots, r_n)$, and C be such that

$$r_1 k_1 + \cdots + r_n k_n = C$$

$r_1 x_1 + \cdots + r_n x_n = C$ is an **equation** that $\vec{k}$ satisfies.

Pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Think of γ as small.

$r_1 x_1 + \cdots + r_n x_n \sim C + e$ is **noisy eq** that $\vec{k}$ satisfies.

Say $\vec{k}$ satisfies the noisy equations

$$r_1 x_1 + \cdots + r_n x_n \sim C_1 + e_1$$

$$s_1 x_1 + \cdots + s_n x_n \sim C_2 + e_2$$

Recall: Noisy Equations

Everything is mod p, some prime p.

Let $\vec{k} = (k_1, \ldots, k_n)$, $\vec{r} = (r_1, \ldots, r_n)$, and C be such that

$$r_1 k_1 + \cdots + r_n k_n = C$$

$r_1 x_1 + \cdots + r_n x_n = C$ is an **equation** that $\vec{k}$ satisfies.

Pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Think of γ as small.

$r_1 x_1 + \cdots + r_n x_n \sim C + e$ is **noisy eq** that $\vec{k}$ satisfies.

Say $\vec{k}$ satisfies the noisy equations

$$r_1 x_1 + \cdots + r_n x_n \sim C_1 + e_1$$

$$s_1 x_1 + \cdots + s_n x_n \sim C_2 + e_2$$

Does $\vec{k}$ satisfy the sum?

$$(r_1 + s_1) x_1 + \cdots + (r_k + s_k) x_k \sim C_1 + C_2 + e_1 + e_2$$

Sums of Noisy Equations

Everything is mod p, some prime p.

Sums of Noisy Equations

Everything is mod p, some prime p.

Say $\vec{k}$ satisfies the noisy equations

$$r_1 x_1 + \cdots r_k x_k \sim C_1 + e_1$$

$$s_1 x_1 + \cdots s_k x_k \sim C_2 + e_2$$

Sums of Noisy Equations

Everything is mod p, some prime p.

Say $\vec{k}$ satisfies the noisy equations

$$r_1x_1 + \cdots r_kx_k \sim C_1 + e_1$$

$$s_1x_1 + \cdots s_kx_k \sim C_2 + e_2$$

Does $\vec{k}$ satisfy the sum?

$$(r_1 + s_1)x_1 + \cdots (r_k + s_k)x_k \sim C_1 + C_2 + e_1 + e_2$$

Sums of Noisy Equations

Everything is mod p, some prime p.
Say $\vec{k}$ satisfies the noisy equations

$$r_1x_1 + \cdots r_kx_k \sim C_1 + e_1$$

$$s_1x_1 + \cdots s_kx_k \sim C_2 + e_2$$

Does $\vec{k}$ satisfy the sum?

$$(r_1 + s_1)x_1 + \cdots (r_k + s_k)x_k \sim C_1 + C_2 + e_1 + e_2$$

The error is in $\{-2\gamma, \ldots, 2\gamma\}$.
We take γ small so that $\vec{k}$ still satisfies the noisy equation.

Sums of Noisy Equations

Everything is mod p, some prime p.

Say $\vec{k}$ satisfies the noisy equations

$$r_1x_1 + \cdots r_kx_k \sim C_1 + e_1$$

$$s_1x_1 + \cdots s_kx_k \sim C_2 + e_2$$

Does $\vec{k}$ satisfy the sum?

$$(r_1 + s_1)x_1 + \cdots (r_k + s_k)x_k \sim C_1 + C_2 + e_1 + e_2$$

The error is in $\{-2\gamma, \ldots, 2\gamma\}$.
We take γ small so that $\vec{k}$ still satisfies the noisy equation.

We add lots of equations, so γ **very** small.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$. She picks 4 rand $\vec{r}$.
   $(4, 9, 1, 89)$, $(9, 98, 8, 1)$, $(44, 55, 10, 8)$, $(9, 3, 11, 99)$.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$. She picks 4 rand $\vec{r}$.
   $(4, 9, 1, 89)$, $(9, 98, 8, 1)$, $(44, 55, 10, 8)$, $(9, 3, 11, 99)$.
   She picks 4 random $e \in \{-1, 0, 1\}$: 1,-1,0,1.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1,0,1\}$.

Alice Wants to Enable Bob to Send $b \in \{0,1\}$.

1. She picks rand: $(1,10,21,89)$. She picks 4 rand $\vec{r}$.
   $(4,9,1,89)$, $(9,98,8,1)$, $(44,55,10,8)$, $(9,3,11,99)$.
   She picks 4 random $e \in \{-1,0,1\}$: 1,-1,0,1.
   She forms 4 noisy eqs which have $(1,10,21,89)$ as "answer."

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$. She picks 4 rand $\vec{r}$.
   $(4, 9, 1, 89)$, $(9, 98, 8, 1)$, $(44, 55, 10, 8)$, $(9, 3, 11, 99)$.
   She picks 4 random $e \in \{-1, 0, 1\}$: 1,-1,0,1.
   She forms 4 noisy eqs which have $(1, 10, 21, 89)$ as "answer."

$$4k_1 + 9k_2 + 21k_3 + 89k_4 \equiv 84$$

$$9k_1 + 98k_2 + 8k_3 + k_4 \equiv 99$$

$$44k_1 + 558k_2 + 10k_3 + 8k_4 \equiv 179$$

$$9k_1 + 3k_2 + 11k_3 + 99k_4 \equiv 105$$

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$. She picks 4 rand $\vec{r}$.
   $(4, 9, 1, 89)$, $(9, 98, 8, 1)$, $(44, 55, 10, 8)$, $(9, 3, 11, 99)$.
   She picks 4 random $e \in \{-1, 0, 1\}$: 1,-1,0,1.
   She forms 4 noisy eqs which have $(1, 10, 21, 89)$ as "answer."

$$4k_1 + 9k_2 + 21k_3 + 89k_4 \equiv 84$$

$$9k_1 + 98k_2 + 8k_3 + k_4 \equiv 99$$

$$44k_1 + 558k_2 + 10k_3 + 8k_4 \equiv 179$$

$$9k_1 + 3k_2 + 11k_3 + 99k_4 \equiv 105$$

These equations are published.

Example of Setting Up The LWE-Public Cipher

Public Info Prime: 191. Length of Vector: 4. Error: $\{-1, 0, 1\}$.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. She picks rand: $(1, 10, 21, 89)$. She picks 4 rand $\vec{r}$.
   $(4, 9, 1, 89)$, $(9, 98, 8, 1)$, $(44, 55, 10, 8)$, $(9, 3, 11, 99)$.
   She picks 4 random $e \in \{-1, 0, 1\}$: 1,-1,0,1.
   She forms 4 noisy eqs which have $(1, 10, 21, 89)$ as "answer."

$$4k_1 + 9k_2 + 21k_3 + 89k_4 \equiv 84$$

$$9k_1 + 98k_2 + 8k_3 + k_4 \equiv 99$$

$$44k_1 + 558k_2 + 10k_3 + 8k_4 \equiv 179$$

$$9k_1 + 3k_2 + 11k_3 + 99k_4 \equiv 105$$

   These equations are published.
   Note Any sum of the eqs also has $(1, 10, 21, 89)$ as "answer."

Bob Wants to Send a Bit

Bob wants to send bit 0.

Bob Wants to Send a Bit

Bob wants to send bit 0.

Pick two of the equations, add them, and sends publicly:

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **close to** solution, so $b = 0$.

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **close to** solution, so $b = 0$.

Bob want to send bit 1.

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **close to** solution, so $b = 0$.

Bob want to send bit 1.
Pick two of the equations, add them, add 50, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 49$$

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **close to** solution, so $b = 0$.

Bob want to send bit 1.
Pick two of the equations, add them, add 50, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 49$$

Eve She sees this equation but does not know which equations were added to form this one.

Bob Wants to Send a Bit

Bob wants to send bit 0.
Pick two of the equations, add them, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 189$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **close to** solution, so $b = 0$.

Bob want to send bit 1.
Pick two of the equations, add them, add 50, and sends publicly:

$$13k_1 + 12k_2 + 32k_3 + 188k_4 \equiv 49$$

Eve She sees this equation but does not know which equations were added to form this one.
Alice She finds that $(1, 10, 21, 99)$ is **far from** solution, so $b = 1$.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0, 1\}$.

1. Alice picks random $\vec{k}$ of length n, her private key.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0,1\}$.

1. Alice picks random $\vec{k}$ of length n, her private key.
2. Alice picks m random $\vec{r}$. For each $\vec{r}$ pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Let $D = \vec{r} \cdot \vec{k} + e$.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0,1\}$.

1. Alice picks random $\vec{k}$ of length n, her private key.
2. Alice picks m random $\vec{r}$. For each $\vec{r}$ pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Let $D = \vec{r} \cdot \vec{k} + e$. Broadcast all $(\vec{r}; D)$.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0,1\}$.

1. Alice picks random $\vec{k}$ of length n, her private key.
2. Alice picks m random $\vec{r}$. For each $\vec{r}$ pick $e \in^r \{-\gamma, \ldots, \gamma\}$.
   Let $D = \vec{r} \cdot \vec{k} + e$. Broadcast all $(\vec{r}; D)$.
   Note $\vec{k}$ satisfies the noisy equations and any sum of them.

Public Key LWE Cipher

Public Info p, the mod. Math is mod p. Param γ, n, m.

Alice Wants to Enable Bob to Send $b \in \{0,1\}$.

1. Alice picks random $\vec{k}$ of length n, her private key.
2. Alice picks m random $\vec{r}$. For each $\vec{r}$ pick $e \in^r \{-\gamma, \ldots, \gamma\}$. Let $D = \vec{r} \cdot \vec{k} + e$. Broadcast all $(\vec{r}; D)$.
   Note $\vec{k}$ satisfies the noisy equations and any sum of them.
3. Bob wants to send bit b. He picks a uniform random set of the public noisy equations and adds them, AND adds $\frac{bp}{2}$.

$$s_1 x_1 + \cdots + s_n x_n \sim D' + \frac{bp}{2} \text{ iff } b = 0$$

D' is sum of Ds. Broadcasts $(\vec{s}; F)$ where $F = D' + \frac{bp}{2}$.

Public Key LWE Cipher (cont)

Where were we:

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.
3. Alice computes $\vec{s} \cdot \vec{k} - F$.

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.
3. Alice computes $\vec{s} \cdot \vec{k} - F$.
   IF SMALL then $b = 0$.
   If LARGE then $b = 1$.

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.
3. Alice computes $\vec{s} \cdot \vec{k} - F$.
   IF SMALL then $b = 0$.
   If LARGE then $b = 1$.

Details omitted, but:

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.
3. Alice computes $\vec{s} \cdot \vec{k} - F$.
   IF SMALL then $b = 0$.
   If LARGE then $b = 1$.

Details omitted, but:

- Will need to take $\gamma \leq \frac{p}{2m}$.

Public Key LWE Cipher (cont)

Where were we:

1. Alice has $\vec{k}$.
2. Bob send Alice $(\vec{s}, F)$ where $F = D' + \frac{bp}{2}$.
3. Alice computes $\vec{s} \cdot \vec{k} - F$.
   IF SMALL then $b = 0$.
   If LARGE then $b = 1$.

Details omitted, but:

- Will need to take $\gamma \leq \frac{p}{2m}$.
- Will need p large so that $\frac{p}{2m}$ is large enough for a variety of error values for increased security.

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

Learning With Errors Problem (LWE) Eve is given p, n, γ and told there is a key $\vec{k}$ of length n that she wants to find.

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

Learning With Errors Problem (LWE) Eve is given p, n, γ and told there is a key $\vec{k}$ of length n that she wants to find.

Eve is given a set of tuples $(\vec{r}, D)$ and told that

$$\vec{r} \cdot \vec{k} - D \in^r \{-\gamma, \ldots, \gamma\}.$$

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

Learning With Errors Problem (LWE) Eve is given p, n, γ and told there is a key $\vec{k}$ of length n that she wants to find.

Eve is given a set of tuples $(\vec{r}, D)$ and told that

$$\vec{r} \cdot \vec{k} - D \in^r \{-\gamma, \ldots, \gamma\}.$$

From these **noisy equations** she wants to learn $\vec{k}$.

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

Learning With Errors Problem (LWE) Eve is given p, n, γ and told there is a key $\vec{k}$ of length n that she wants to find.

Eve is given a set of tuples $(\vec{r}, D)$ and told that

$$\vec{r} \cdot \vec{k} - D \in^r \{-\gamma, \ldots, \gamma\}.$$

From these **noisy equations** she wants to learn $\vec{k}$.
Hard? We discuss why this problem is thought to be hard.

LWE-Public: Security

What problem does Eve need to solve to find the key? (Same one as LWE-private.)

Learning With Errors Problem (LWE) Eve is given p, n, γ and told there is a key $\vec{k}$ of length n that she wants to find.

Eve is given a set of tuples $(\vec{r}, D)$ and told that

$$\vec{r} \cdot \vec{k} - D \in^r \{-\gamma, \ldots, \gamma\}.$$

From these **noisy equations** she wants to learn $\vec{k}$.
Hard? We discuss why this problem is thought to be hard.
Nice Bonus Avg Case LWE is easy implies Worst Case LWE is easy.

LWE-Public: Security (cont)

Theorem If Eve can crack the LWE-public cipher then Eve can solve the LWE-problem. Note that this is the direction you want.

LWE-Public: Security (cont)

Theorem If Eve can crack the LWE-public cipher then Eve can solve the LWE-problem. Note that this is the direction you want.
Proof We won't prove this, but we note that it requires some work.

LWE-Public: Security (cont)

Theorem If Eve can crack the LWE-public cipher then Eve can solve the LWE-problem. Note that this is the direction you want.
Proof We won't prove this, but we note that it requires some work.

When discussing **LWE-Private** we just said

LWE-problem is thought to be hard.

LWE-Public: Security (cont)

Theorem If Eve can crack the LWE-public cipher then Eve can solve the LWE-problem. Note that this is the direction you want.
Proof We won't prove this, but we note that it requires some work.

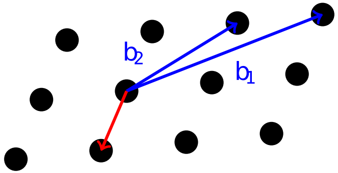
When discussing **LWE-Private** we just said

LWE-problem is thought to be hard.

We now go into that some more.

Shortest Vector Problem (SVP)

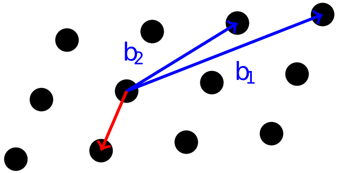
SVP Given a lattice, find the shortest Vector out of the origin.



(Picture by Sebastian Schmittner - Own work, CC BY-SA 4.0, `https://commons.wikimedia.org/w/index.php?curid=44488873`)

Shortest Vector Problem (SVP)

SVP Given a lattice, find the shortest Vector out of the origin.

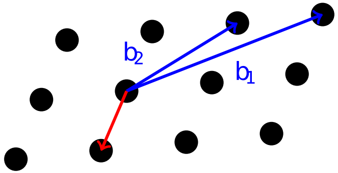


(Picture by Sebastian Schmittner - Own work, CC BY-SA 4.0, `https://commons.wikimedia.org/w/index.php?curid=44488873`)

Hardness Known to be NP-hard under randomized reductions.

Shortest Vector Problem (SVP)

SVP Given a lattice, find the shortest Vector out of the origin.



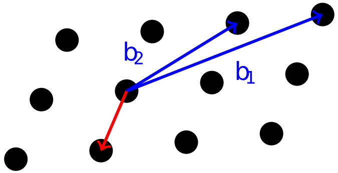
(Picture by Sebastian Schmittner - Own work, CC BY-SA 4.0, `https://commons.wikimedia.org/w/index.php?curid=44488873`)

Hardness Known to be NP-hard under randomized reductions.

Want SVP $\leq$ LWE $\leq$ LWE-Public.

Shortest Vector Problem (SVP)

SVP Given a lattice, find the shortest Vector out of the origin.



(Picture by Sebastian Schmittner - Own work, CC BY-SA 4.0,
`https://commons.wikimedia.org/w/index.php?curid=44488873`)

Hardness Known to be NP-hard under randomized reductions.

Want SVP $\le$ LWE $\le$ LWE-Public.

We don't have this but we have something similar.

Gap-Shortest Vector Problem (Gap-SVP)

SVP Given a lattice, find the shortest Vector out of the origin.

Gap-Shortest Vector Problem (Gap-SVP)

SVP Given a lattice, find the shortest Vector out of the origin.

Gap-SVP Given a lattice, find if the shortest Vector out of the origin is LONG or SHORT. If its neither, still give an answer, but it won't mean anything.

Gap-Shortest Vector Problem (Gap-SVP)

SVP Given a lattice, find the shortest Vector out of the origin.

Gap-SVP Given a lattice, find if the shortest Vector out of the origin is LONG or SHORT. If its neither, still give an answer, but it won't mean anything.

Want Gap-SVP $\leq$ LWE $\leq$ LWE-Public.

Gap-Shortest Vector Problem (Gap-SVP)

SVP Given a lattice, find the shortest Vector out of the origin.

Gap-SVP Given a lattice, find if the shortest Vector out of the origin is LONG or SHORT. If its neither, still give an answer, but it won't mean anything.

Want Gap-SVP $\le$ LWE $\le$ LWE-Public. We do have this! Sort of.

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \;\le\; \text{LWE} \;\le\; \text{LWE-Public}$$

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \;\leq\; \text{LWE} \;\leq\; \text{LWE-Public}$$

This is true. Sort of.

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \;\leq\; \text{LWE} \;\leq\; \text{LWE-Public}$$

This is true. Sort of.
$\text{Gap-SVP} \;\leq\; \text{LWE}$ is a **Quantum Reduction**
Quantum Reduction means the reduction works if you have a quantum computer.

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \le \text{LWE} \le \text{LWE-Public}$$

This is true. Sort of.

Gap-SVP $\le$ LWE is a **Quantum Reduction**

Quantum Reduction means the reduction works if you have a quantum computer.

Its a Win-Win!

QC means that Quantum Computing is Practical.

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \le \text{LWE} \le \text{LWE-Public}$$

This is true. Sort of.

Gap-SVP $\le$ LWE is a **Quantum Reduction**

Quantum Reduction means the reduction works if you have a quantum computer.

Its a Win-Win!

QC means that Quantum Computing is Practical.

1. $\neg QC$: RSA secure (against Quantum Factoring).

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \leq \text{LWE} \leq \text{LWE-Public}$$

This is true. Sort of.

$\text{Gap-SVP} \leq \text{LWE}$ is a **Quantum Reduction**

Quantum Reduction means the reduction works if you have a quantum computer.

Its a Win-Win!

QC means that Quantum Computing is Practical.

1. $\neg QC$: RSA secure (against Quantum Factoring).
2. QC: LWE-Public is secure (assuming GAP-SVP is hard).

LWE-Public. Hardness Assumption – A Caveat

Want:

$$\text{Gap-SVP} \ \le \ \text{LWE} \ \le \ \text{LWE-Public}$$

This is true. Sort of.

Gap-SVP $\le$ LWE is a **Quantum Reduction**

Quantum Reduction means the reduction works if you have a quantum computer.

Its a Win-Win!

QC means that Quantum Computing is Practical.

1. $\neg QC$: RSA secure (against Quantum Factoring).
2. QC: LWE-Public is secure (assuming GAP-SVP is hard).

Caveat Regev showed the quantum reduction in 2009. Peikert obtained a randomized reduction in 2014. The quantum reduction works for a wider range of parameters.

Is LWE-private Being Used?

NIST has initiated a process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptosystems:

Is LWE-private Being Used?

NIST has initiated a process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptosystems:

Many of the finalists are LWE or similar to LWE.

Is LWE-private Being Used?

NIST has initiated a process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptosystems:

Many of the finalists are LWE or similar to LWE.

Note that what I showed here were the IDEAS behind LWE-public.
Getting it to actually work requires many modifications.

BILL, STOP RECORDING LECTURE!!!!

BILL STOP RECORDING LECTURE!!!