6.892 Computing on Encrypted Data	September 23, 2013
Lecture 3	
Lecturer: Vinod Vaikuntanathan	*Scribe: Chiraag Juvekar*

1 Introduction

In this lecture we will see a fuller development of the LWE based homomorphic encryption scheme. The multiplication operation in the naive LWE − HE scheme changes the form of the cipher text. In this lecture we see a dimensionality reduction trick to restore the form of the original ciphertext. Finally we conclude with reductions between LWE assumptions that relax the constraints on the probability distribution from which the random matrix for the encryptions is drawn.

2 LWE based Homomorphic Encryption

In this lecture we mainly focus on the LWE based private-key HE scheme. We note that it is possible to extend any compact private key homomorphic scheme into a public key homomorphic scheme that is just slightly less homomorphic [Rot11]. However since our development will eventually lead to both levelled and fully homomorphic schemes this slight reduction in homomorphic capability is not a very big concern.

2.1 Basic secret-key encryption scheme

We first discuss the basic LWE based scheme that we described in the last class. The security parameters for this scheme are (n, q, χ) which are the dimension, modulus and the error distribution respectively. The error distribution χ is B-bounded, that is $\Pr_{x \leftarrow \chi}[|x| \geq B] = \text{negl}(\cdot)$. The various algorithms are:

- $\mathsf{Keygen}(1^n)$: Pick $\bar{t} \leftarrow \mathbb{Z}_q^n$
- $\mathsf{Enc}_{sk}(\mu)$: $\left(\overline{a}, \langle \overline{a}, \bar{t} \rangle + e + \mu \left\lfloor \frac{q}{2} \right\rceil\right) = (\overline{a}, b) = \overline{c} \in \mathbb{Z}_q^{n+1}$ where $(a, e) \leftarrow (\mathbb{Z}_q^n, \chi)$ and $\mu \in \{0, 1\}$
- $\mathsf{Dec}_{sk}(\overline{a}, b)$: $\text{Round}_{\frac{q}{2}}\left(b - \langle \overline{a}, \bar{t} \rangle\right) = \text{Round}_{\frac{q}{2}}\left(e + \mu \left\lfloor \frac{q}{2} \right\rceil\right)$

Let $\overline{s} = (-\bar{t}, 1)$. Hence we have $\mathsf{Dec}_{sk}(\overline{a}, b) = \text{Round}_{\frac{q}{2}}\left(\langle \overline{c}, \bar{t} \rangle\right)$.

- **Correctness**: The scheme is correct as long as $|e| \leq \frac{q}{4}$
- **Security**: The security is based on the hardness of the LWE problem. For LWE as long as $\frac{q}{B} < 2^{n^{\epsilon}}$, the run time of the best attacks against LWE is $O\left(2^{n^{1-\epsilon}}\right)$

The Eval algorithms for add and mult are given as follows:

- add: $\overline{c}_{add} = \overline{c}_1 + \overline{c}_2 \mod q$.

  Correctness of additions follows because:

$$
\begin{aligned}
\langle \overline{c}_1, \overline{s} \rangle &= e_1 + \mu_1 \left\lfloor \frac{q}{2} \right\rceil \\
\langle \overline{c}_2, \overline{s} \rangle &= e_2 + \mu_2 \left\lfloor \frac{q}{2} \right\rceil \\
\langle \overline{c}_a dd, \overline{s} \rangle &= (e_1 + e_2) + (\mu_1 + \mu_2) \left\lfloor \frac{q}{2} \right\rceil \\
&= (e_{add}) + (\mu_1 \oplus \mu_2) \left\lfloor \frac{q}{2} \right\rceil
\end{aligned}
$$

Thus $e_{add} \leq (e_1 + e_2 + 1)$ and the add function evaluates the $\oplus$ operation. As long as $|e_{add}| < \frac{q}{4}$ we are good.

- mult: $\bar{c}_{mult} = 2 \cdot \bar{c}_1 \otimes \bar{c}_2 \mod q$.

  Correctness of multiplication follows because:

$$\begin{aligned}
\langle \bar{c}_1, \bar{s} \rangle &= e_1 + \mu_1 \left\lfloor \frac{q}{2} \right\rceil \\
\langle \bar{c}_2, \bar{s} \rangle &= e_2 + \mu_2 \left\lfloor \frac{q}{2} \right\rceil \\
2 \cdot \langle \bar{c}_1, \bar{s} \rangle \cdot \langle \bar{c}_1, \bar{s} \rangle &= 2 \cdot \left(e_1 + \mu_1 \left\lfloor \frac{q}{2} \right\rceil \right) \cdot \left(e_2 + \mu_2 \left\lfloor \frac{q}{2} \right\rceil \right) \\
&= 2e_1e_2 + (\mu_1 e_2 + \mu_2 e_1) + \mu_1\mu_2 \left\lfloor \frac{q}{2} \right\rceil \\
\text{But, } 2 \cdot \langle \bar{c}_1, \bar{s} \rangle \cdot \langle \bar{c}_1, \bar{s} \rangle &= 2 \cdot \langle \bar{c}_1 \otimes \bar{c}_2, \bar{s} \otimes \bar{s} \rangle \\
\text{Hence, } 2 \cdot \langle \bar{c}_1 \otimes \bar{c}_2, \bar{s} \otimes \bar{s} \rangle &= 2e_1e_2 + (\mu_1 e_2 + \mu_2 e_1) + \mu_1\mu_2 \left\lfloor \frac{q}{2} \right\rceil \\
&= e_{mult} + \mu_1\mu_2 \left\lfloor \frac{q}{2} \right\rceil
\end{aligned}$$

The above approach for mult has two major drawbacks:

- The dimension of the output of the multiplication has changed. $\bar{c}_1, \bar{c}_2 \leftarrow \mathbb{Z}_q^{n+1}$ but $\bar{c}_{mult} \leftarrow \mathbb{Z}_q^{(n+1)^2}$. Thus after evalutaing a circuit C of depth d the dimension of the output grows from $(n+1) \rightarrow (n+1)^{2^d}$. This is a huge and unreasonable penalty to pay for outsourcing computation since the decryption algorithm must deal with huge ciphertexts.

- e_{mult} grows as the square of the initial error. Thus when evaluating a circuit of depth d, the final error is $e_{init}^{2^d}$. We need to control the fast growth of this error.

3 Dimension Switching

To solve the first of the two problems mentioned above we use a technique called dimension switching. To gain some intuition about this procedure we first discuss an "error-less" version that is insecure. We will then convert to secure version using some error terms.

3.1 Error-less Dimension Switching

Consider that we have a vector $\bar{c}_{mult} \in \mathbb{Z}_q^{(n+1)^2}$ such that $\langle \bar{c}_{mult}, \bar{s} \otimes \bar{s} \rangle = \mu \left\lfloor \frac{q}{2} \right\rceil$. We want to find a new vector $\overline{c'}_{mult} \in \mathbb{Z}_q^{n+1}$ such that $\left\langle \overline{c'}_{mult}, \bar{s} \right\rangle = \mu \left\lfloor \frac{q}{2} \right\rceil$. To accomplish this assume that the secret key owner publishes a hint matrix D such that $D^T \bar{s} = \bar{s} \otimes \bar{s}$.

We can see that,

$$\langle D \cdot \bar{c}_{mult}, \bar{s} \rangle = \bar{s}^T \left(D \cdot \bar{c}_{mult} \right) = \left(\bar{s}^T \cdot D \right) \bar{c}_{mult} = \left(D^T \cdot \bar{s} \right)^T \cdot \bar{c}_{mult} = \left(\bar{s} \otimes \bar{s} \right)^T \cdot \bar{c}_{mult} = \langle \bar{c}_{mult}, \bar{s} \otimes \bar{s} \rangle$$

Thus, $\overline{c'}_{mult} = D \cdot \bar{c}_{mult}$. Unfortunately publishing such a D is completely insecure. Assume that ψ_{ij} is the ij^{th}-row of D^T. Then

$$\psi_{ij}[n] = \begin{cases} \bar{s}[i], \text{if } n = j. \\ 0, \text{otherwise.} \end{cases}$$

Thus given D we can simply read off $\bar{s}$.

3.2 Real Dimension Switching

In order to secure the D, we simply publish a noisy version such that,

$$\psi_{ij} = \widetilde{\mathsf{Enc}}_{sk}(\overline{s}[i]\overline{s}[j]) = \left(\overline{a}, \left\langle \overline{a}, \overline{t} \right\rangle + e_{ij} + \overline{s}[i]\overline{s}[j]\right)$$

Note that ψ_{ij} is not a valid cipher text since $s[i]s[j] \in \mathbb{Z}_q^{n+1}$ and not $0, 1$. Thus ψ_{ij} does not decrpyt to a valid ciphertext. This is not an issue for us since we will never try to decrypt ψ_{ij}. What matters is that $\langle \psi_{ij}, s \rangle = \overline{s}[i]\overline{s}[j] + e_{ij}$.

When $sk \neq \overline{s}$, the security of the $\widetilde{\mathsf{Enc}}_{sk}$ operation is equivalent to the LWE-assumption. When $sk = \overline{s}$, this is equivalent to assuming that LWE is hard under a circular security assumption.

4 Homomorphic Multiplication with Dimension Switching

From the above discussion we can construct a new homomorphic multiplication algorithm as follows.

1. Tensoring: $\overline{c}_{mult} = 2 \cdot \overline{c}_1 \otimes \overline{c}_2 \in \mathbb{Z}_q^{(n+1)^2}$
2. Dimension Switching: $\overline{c'}_{mult} = D \cdot \overline{c}_{mult}$.

Thus we have,

$$\begin{aligned}
\left\langle \overline{c'}_{mult}, \overline{s} \right\rangle &= \langle D \cdot \overline{c}_{mult}, \overline{s} \rangle \\
&= \left\langle \sum_{i,j \in [n+1]} \overline{c}_{mult}[i,j] \cdot \psi_{ij}, \overline{s} \right\rangle \\
&= \sum \overline{c}_{mult}[i,j] \cdot \langle \psi_{ij}, \overline{s} \rangle \\
&= \sum \overline{c}_{mult}[i,j] \cdot (\overline{s}[i]\overline{s}[j] + e_{ij}) \\
&= \sum \overline{c}_{mult}[i,j]\overline{s}[i]\overline{s}[j] + \sum \overline{c}_{mult}[i,j] e_{ij} \\
&= \sum 2 \cdot \overline{c}_1[i]\overline{c}_2[j]\overline{s}[i]\overline{s}[j] + e_{dr} \\
&= 2 \cdot \langle \overline{c}_1 \otimes \overline{c}_2, \overline{s} \otimes \overline{s} \rangle + e_{dr} \\
&= \mu_1 \mu_2 \left\lfloor \frac{q}{2} \right\rceil + e_{mult} + e_{dr}
\end{aligned}$$

Thus dimension switching convert an $(n+1)^2$-dimension ciphertext back to $(n+1)$-dimension ciphertext. Unfortunately this incurs an extra error penalty term. This is e_{dr}, the error of performing dimension reduction. In general this error may be large because,

$$\begin{aligned}
e_{dr} &= \sum_{i,j \in [n+1]} \overline{c}_{mult}[i,j] e_{ij} \\
&\leq \sum_{i,j \in [n+1]} q \cdot |B| \\
&\leq (n+1)^2 q |B|
\end{aligned}$$

Thus e_{dr} may quite easily be greater than $\frac{q}{4}$ and the final result may not decode correctly. In order to reduce the magnitude of e_{dr} we use a further trick involving the binary representations.

4.1 Binary Representation Trick

Instead of using a packed representation for $\overline{c}_{mult}[i, j]$, we describe it using an expanded bit-representation. Thus we can write,

$$\overline{c}_{mult}[i, j] = \sum_{\tau=0}^{\lfloor \log q \rfloor} \overline{c}_{mult}[i, j, \tau] \cdot 2^{\tau} \text{where } \overline{c}_{mult}[i, j, \tau] \in \{0, 1\}$$

Further let, $\psi_{ij\tau} = \overline{s}[i]\overline{s}[j] \cdot 2^{\tau} + e_{ij\tau}$. If we are to publish the extended D' matrix with the columns as $\psi_{ij\tau}$, we now have,

$$\begin{aligned}
\left\langle \overline{c'}_{mult}, \overline{s} \right\rangle &= \langle D' \cdot \overline{c}_{mult}, \overline{s} \rangle \\
&= \left\langle \sum_{\substack{i,j \in [n+1] \\ \tau \in [\lfloor \log q \rfloor]}} \overline{c}_{mult}[i, j, \tau] \cdot \psi_{ij\tau}, \overline{s} \right\rangle \\
&= \sum \overline{c}_{mult}[i, j, \tau] \cdot \langle \psi_{ij\tau}, \overline{s} \rangle \\
&= \sum \overline{c}_{mult}[i, j, \tau] \cdot (\overline{s}[i]\overline{s}[j] \cdot 2^{\tau} + e_{ij}) \\
&= \sum \overline{c}_{mult}[i, j, \tau] \cdot 2^{\tau} \overline{s}[i]\overline{s}[j] + \sum \overline{c}_{mult}[i, j, \tau] e_{ij} \\
&= \sum \left(\sum \overline{c}_{mult}[i, j, \tau] 2^{\tau} \right) \overline{s}[i]\overline{s}[j] + e_{dr} \\
&= \sum \overline{c}_{mult}[i, j] \overline{s}[i]\overline{s}[j] + e_{dr} \\
&= \mu_1 \mu_2 \left\lfloor \frac{q}{2} \right\rceil + e_{mult} + e_{dr}
\end{aligned}$$

Although this analysis is very similar to the packed representation cases, since $\overline{c}_{mult}[i, j, \tau] \in \{0, 1\}$ we now have a tighter bound on e_{dr}. Infact we can show that $e_{dr} \leq (n+1)^2 \lfloor \log q \rfloor |B|$.

4.2 Somewhat Homomorphic Encryption with Dimension Switching

Now that we have an efficient procedure for dimension switching we will look at an L-level $\mathsf{LWE} - \mathsf{SH}$ scheme. The scheme as described does not make any assumptions on circular security of the LWE problem but can be made more efficient using that assumption.[BV11][BGV12]

- $\mathsf{Keygen}(1^n)$: $\forall i \in \{0, 1, \ldots, L\}$ pick $L+1$-independent $\overline{t}_i \leftarrow \mathbb{Z}_q^n$. Let $\overline{s}_i = (-\overline{t}_i, 1)$.
- $\mathsf{Enc}_{sk}(\mu)$: $\left(\overline{a}, \left\langle \overline{a}, \overline{t}_0 \right\rangle + e + \mu \left\lfloor \frac{q}{2} \right\rceil \right) = (\overline{a}, b) = \overline{c} \in \mathbb{Z}_q^{n+1}$ where $(a, e) \leftarrow (\mathbb{Z}_q^n, \chi)$ and $\mu \in \{0, 1\}$
- $\mathsf{Dec}_{sk}(\overline{c})$: $\text{Round}_{\frac{q}{2}} \left(\langle \overline{c}, \overline{s}_L \rangle \right) \text{Round}_{\frac{q}{2}} \left(e + \mu \left\lfloor \frac{q}{2} \right\rceil \right)$

In addition, to aid dimension switching we publish a set of L-evaluation keys $evk = \{evk_1, ..., evk_L\}$ such that,

$$evk_l = \widetilde{\mathsf{Enc}}_{s_l}(s_{l-1}[i]s_{l-1}[j] \cdot 2^{\tau}) \text{where } i, j \in [n+1], \tau \in [\lfloor \log q \rfloor]$$

- **Correctness**: The scheme is correct as long as the L-level ciphertexts are decodable. Thus $B^{2^L} \leq \frac{q}{4}$
- **Security**: The security is based on the hardness of the LWE problem. Hence $\frac{q}{B} < 2^{n^{\epsilon}}$.

The above two inequalities tells us that $L \approx \epsilon \log n$

5 Reductions for the LWE problem

After reducing the error in the dimension reduction term we focus on the e_{mult}. In order to reduce this error we first prove some result regarding the hardness of the *lwe* problem when the secret is chosen from the error distribution χ.

In particular we will look at the following two results:

Lemma 1. *[ACPS09] The* LWE *with secret* $\bar{t} \leftarrow \chi^n$ *is as hard as* LWE *with secret* $\bar{t} \leftarrow \mathbb{Z}_q^n$*.*

Proof. Let $\mathsf{LWE} \sim \mathsf{LWE}_{n,m,q,\chi,U}$: $\bar{t} \leftarrow \mathbb{Z}_q^n, e \leftarrow \chi$ and let $\mathsf{LWE}' \sim \mathsf{LWE}_{n,m,q,\chi,\chi}$: $\bar{t} \leftarrow \chi^n, e \leftarrow \chi$.

Assume that we have an oracle that solves LWE. We wish to use the oracle to find the secret $\bar{t} \leftarrow \chi^n$ when given $(A, \langle A, \bar{t} \rangle + e)$. Pick $\bar{s} \leftarrow \mathbb{Z}_q^n$. Hence $\bar{s} + \bar{t} \leftarrow \mathbb{Z}_q^n$. Now feed the oracle $(A, \langle A, \bar{t} + \bar{s} \rangle + e)$ which it can now solve. Thus knowing $\bar{s}$ we can now recover the original $\bar{t}$. Thus $\mathsf{LWE}' leq \mathsf{LWE}$

Assume that we have an oracle that solves LWE'. We wish to use the oracle to find the secret $\bar{t} \leftarrow \mathbb{Z}_q^n$ when given $(A, \langle A, \bar{t} \rangle + e)$. We rewrite the above as,

$$\left(\begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \cdot \bar{t} + \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \right) \tag{1}$$

where $A_1 \leftarrow \mathbb{Z}_q^{n^2}, A_2 \leftarrow \mathbb{Z}_q^{mn}$.

Now $b_1 = A_1 \cdot \bar{t} + e_1$. With very high probability A_1 is an invertible matrix. Hence $\bar{t} = A_1^{-1} \cdot (b_1 - e_1)$. Thus we see that the error and secret in LWE are in some sense interchangeable. More precisely,

$$\begin{aligned} b_2 &= A_2 \cdot \bar{t} + e_2 \\ &= A_2 A_1^{-1}(b_1 - e_1) + e_2 \\ &= A_2 A_1^{-1} b_1 - A_2 A_1^{-1} e_1 + e2 \end{aligned}$$

Thus if we feed our oracle $\big((-A_2 A_1^{-1}, -A_2 A_1^{-1} e_1 + e2)\big)$, it will solve it since $e_1 \leftarrow \chi^n$. But once we get the value of e_1 we can find $\bar{t}$ and thus solve our orginal LWE instance. Thus $\mathsf{LWE} \leq \mathsf{LWE}'$

Hence $\mathsf{LWE} = \mathsf{LWE}'$ □

Infact an even stronger result than the one shown above holds.

Lemma 2. *[GKPV08] The* LWE *with secret* $\bar{t} \leftarrow \mathcal{D}^n$ *is as hard as* LWE *with secret* $\bar{t} \leftarrow \mathbb{Z}_q^n$ *where* $\mathcal{D}$ *is any distribution with large enough min-entropy.*

References

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai, *Fast cryptographic primitives and circular-secure encryption based on hard learning problems*, Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Lecture Notes in Computer Science, vol. 5677, Springer, 2009, pp. 595–618.

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan, *(leveled) fully homomorphic encryption without bootstrapping*, Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (New York, NY, USA), ITCS '12, ACM, 2012, pp. 309–325.

[BV11] Z. Brakerski and V. Vaikuntanathan, *Efficient fully homomorphic encryption from (standard) lwe*, Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on, 2011, pp. 97–106.

[GKPV08] Shafi Goldwasser, Yael Kalai, Chris Peikert, and Vinod Vaikuntanathan, *Robustness of the learning with errors assumption*, In ICS. 2010. [GPV08] [GRS08, 2008.

[Rot11] Ron Rothblum, *Homomorphic encryption: From private-key to public-key*, Theory of Cryptography (Yuval Ishai, ed.), Lecture Notes in Computer Science, vol. 6597, Springer Berlin Heidelberg, 2011, pp. 219–234.