Notes on Two Fully Homomorphic Encryption Schemes Without Bootstrapping

Yongge Wang

KINDI Center for Computing Research
Qatar University
yongge.wang@qu.edu.qa

Abstract. Recently, IACR ePrint archive posted two fully homomorphic encryption schemes without bootstrapping. In this note, we show that these schemes are trivially insecure. Furthermore, we also show that the encryption schemes of Liu and Wang [6] in CCS 2012 and the encryption scheme of Liu, Bertino, and Xun [5] in ASIACCS 2014 are insecure either.

1 Introduction

Though it is a very challenging problem to design fully homomorphic encryption schemes without bootstrapping. We still see that quite a few researchers post candidate designs frequently. This note points out that the two schemes posted to IACR ePrint archive recently are trivially insecure: the scheme by Masahiro Yagisawa [7] on 2015-05-19 and the scheme by Dongxi Liu [4] on 2015-05-17.

2 Masahiro Yagisawa [7]'s Scheme

Octonion (see, e.g., Conway and Smith [3] or Baez [1]) is the largest of the four normed division algebra and is the only normed division algebra that is neither commutative nor associative. Each octonion number is a vector $\mathbf{a} = [a_0, \cdots, a_7] \in R^8$ where R is the real number. For each octonion number $\mathbf{a} = [a_0, \cdots, a_7]$, we define an associated 8×8 matrix

$$A_{\mathbf{a}} = \begin{pmatrix} a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 \\ -a_1 & a_0 & a_4 & a_7 & -a_2 & a_6 & -a_5 & -a_3 \\ -a_2 & -a_4 & a_0 & a_5 & a_1 & -a_3 & a_7 & -a_6 \\ -a_3 & -a_7 & -a_5 & a_0 & a_6 & a_2 & -a_4 & a_1 \\ -a_4 & a_2 & -a_1 & -a_6 & a_0 & a_7 & a_3 & -a_5 \\ -a_5 & -a_6 & a_3 & -a_2 & -a_7 & a_0 & a_1 & a_4 \\ -a_6 & a_5 & -a_7 & a_4 & -a_3 & -a_1 & a_0 & a_2 \\ -a_7 & a_3 & a_6 & -a_1 & a_5 & -a_4 & -a_2 & a_0 \end{pmatrix}$$

For two octonions $\mathbf{a} = [a_0, \cdots, a_7]$ and $\mathbf{b} = [b_0, \cdots, b_7]$, we can add them as $\mathbf{a} + \mathbf{b} = [a_0 + b_0, \cdots, a_7 + b_t]$ and multiply them as $\mathbf{ab} = \mathbf{b}A_{\mathbf{a}}$. The norm of an octonion $\mathbf{a} = [a_0, \cdots, a_7]$ is defined as $\|\mathbf{a}\| = \sqrt{a_0^2 + \cdots + a_7^2}$.

Let $\mathbf{1} = [1, 0, 0, 0, 0, 0, 0, 0]$. Using octonions over $GF(q)$, Yagisawa [7] introduced the following fully homomorphic encryption scheme.

Key Setup. Let $\mathbf{x}$ be a variable representing octonions. Choose random invertible octonions $\mathbf{k}_0, \cdots, \mathbf{k}_{t-1} \in GF(q)^8$. The private key is $\mathtt{key} = \{\mathbf{k}_0, \cdots, \mathbf{k}_{t-1}\}$.
Encryption. For a message $\mathbf{m} \in GF(q)^8$, let the cipher text

$$\mathbf{c_m}(\mathbf{x}) = \mathtt{O.Enc}(\mathtt{key}, \mathbf{m}) = \mathbf{k}_0(\cdots(\mathbf{k}_{t-1}(\mathbf{m}(\mathbf{k}_{t-1}^{-1}(\cdots(\mathbf{k}_0^{-1}\mathbf{x})\cdots))))\cdots).$$

Decryption. Let $g_0(\mathbf{x}) = \mathbf{k}_{t-1}^{-1}(\cdots(\mathbf{k}_0^{-1}\mathbf{x})\cdots)$ and $g_1(\mathbf{x}) = \mathbf{k}_0(\cdots(\mathbf{k}_{t-1}\mathbf{x})\cdots)$. For a received ciphertext $\mathbf{c_m}(\mathbf{x})$, compute $\mathbf{m} = \mathtt{O.Dec}(\mathtt{key},(\mathbf{c_m}(\mathbf{x})) = g_0(\mathbf{c_m}(g_1(\mathbf{1})))$.
Ciphertext addition. The addition of two ciphertexts $\mathbf{c}_{\mathbf{m}_0}(\mathbf{x})$ and $\mathbf{c}_{\mathbf{m}_1}(\mathbf{x})$ is defined as the component wise addition $\mathbf{c}_{\mathbf{m}_0+\mathbf{m}_1}(\mathbf{x}) = \mathbf{c}_{\mathbf{m}_0}(\mathbf{x}) + \mathbf{c}_{\mathbf{m}_1}(\mathbf{x})$. That is, this is just the octonion addition.
Ciphertext multiplication. The multiplication of two ciphertexts $\mathbf{c}_{\mathbf{m}_0}(\mathbf{x})$ and $\mathbf{c}_{\mathbf{m}_1}(\mathbf{x})$ is defined as $\mathbf{c}_{\mathbf{m}_1\mathbf{m}_0} = \mathbf{c}_{\mathbf{m}_1}(\mathbf{c}_{\mathbf{m}_0}(\mathbf{x}))$.

It should be noted that the above scheme is not fully homomorphic over $GF(q)$. Indeed, it is only fully homomorphic over octonion numbers over $GF(q)$ since the multiplication of ciphertexts is the ciphertext of an octonion number which is a multiplication over the octonions. Since the multiplication in octonions is neither associative nor commutative, the above scheme cannot be fully homomorphic scheme over $GF(q)$ using the above scheme. In order to achieve FHE over $GF(q)$, Yagisawa's [7] uses the following message coding technique.

Let $\mathbf{z} \in GF(q)^8$ be a random octonion with $\|\mathbf{z}\| = 0$ and $z_0 \neq 0$. For a message $m \in GF(q)$, choose a random $r \in GF(q)$ and encode the message m as an octonion number $\mathbf{m} = m\mathbf{1} + r\mathbf{z} \in GF^{8\times 8}$. Using this encoding approach, Yagisawa showed that his scheme is fully homomorphic over $GF(q)$. The details could be found in [7].

It is straightforward to observe that for the above scheme with message encoding, the message 0 is encrypted to a ciphertext $\mathbf{c}_0(\mathbf{x}) = \mathbf{c}_{r\mathbf{z}}(\mathbf{x})$ such that $\|\mathbf{c}_{r\mathbf{z}}(\mathbf{1})\| = 0$. In other words, we can easily distinguish ciphertexts of m and $-m$ since $\|(\mathbf{c}_m + \mathbf{c}_{-m})(\mathbf{1})\| = 0$.

2.1 A variant of Yagisawa's [7] encryption scheme

Yagisawa's [7] encryption scheme is defined in terms of a sequence of octonions. It will be interesting to have a simplified definition of Yagisawa's scheme in terms of matrix operations. In the following, we use matrix operations to define a variant of Yagisawa's scheme. Then we show that the scheme is equivalent to Yagisawa's scheme when only encoded messages over $GF(q)$ are encrypted. However, the variant scheme is not equivalent to Yagisawa's scheme when messages are plain octonions. Let $GF(q)$ be the underlying finite field that we will work with. Then the protocol works as follows:
Key Setup. Choose a random invertible 8×8 matrix $K \in GF(q)^{8\times 8}$ as the private key.
Encryption. For a message $\mathbf{m} \in GF(q)^8$, the cipher text $C_{\mathbf{m}} = \mathtt{M.Enc}(K, \mathbf{m}) = K^{-1}A_{\mathbf{m}}K \in GF^{8\times 8}$ where $A_{\mathbf{m}}$ is the associated matrix for $\mathbf{m}$ when $\mathbf{m}$ is considered as an octonion number.
Decryption. For a received ciphertext $C_{\mathbf{m}}$, compute

$$\mathbf{m} = \mathtt{M.Dec}(K, C_{\mathbf{m}}) = \mathbf{1}(KC_{\mathbf{m}}K^{-1}) = \mathbf{1}A_{\mathbf{m}}.$$

Ciphertext addition. The addition of two ciphertexts $C_{\mathbf{m}_0}$ and $C_{\mathbf{m}_1}$ is defined as the component wise addition $C_{\mathbf{m}_0+\mathbf{m}_1} = C_{\mathbf{m}_0} + C_{\mathbf{m}_1}$. That is, this is just the regular matrix addition.

Ciphertext multiplication. The multiplication of two ciphertexts $C_{\mathbf{m}_0}$ and $C_{\mathbf{m}_1}$ is defined as the regular matrix multiplication

$$C_{\mathbf{m}_0\mathbf{m}_1} = C_{\mathbf{m}_1}C_{\mathbf{m}_0} = KA_{\mathbf{m}_1}K^{-1}KA_{\mathbf{m}_0}K^{-1} = KA_{\mathbf{m}_1}A_{\mathbf{m}_0}K^{-1}.$$

It is straightforward that the above encryption scheme can have ciphertext addition homomorphically for unlimited times. However, the ciphertext multiplication can only be used for one time by the following observation. Let $\mathbf{m}_0, \mathbf{m}_1, \mathbf{m}_2$ be octonions such that $\mathbf{m}_0(\mathbf{m}_1\mathbf{m}_2) \neq (\mathbf{m}_0\mathbf{m}_1)\mathbf{m}_2$. By definition, we have

$$\begin{aligned} \texttt{M.Dec}(K, C_{\mathbf{m}_2}C_{\mathbf{m}_0\mathbf{m}_1}) &= \texttt{M.Dec}(K, K^{-1}A_{\mathbf{m}_2}A_{\mathbf{m}_1}A_{\mathbf{m}_0}K) \\ &= \mathbf{1}A_{\mathbf{m}_2}A_{\mathbf{m}_1}A_{\mathbf{m}_0} \\ &= \mathbf{m}_0(\mathbf{m}_1\mathbf{m}_2) \\ &\neq (\mathbf{m}_0\mathbf{m}_1)\mathbf{m}_2 \end{aligned} \tag{1}$$

It follows that $C_{(\mathbf{m}_0\mathbf{m}_1)\mathbf{m}_2} \neq C_{\mathbf{m}_2}C_{\mathbf{m}_0\mathbf{m}_1}$.

In the above scheme, the ciphertext multiplication could not be used for more than one time due to the fact that that $A_{\mathbf{m}_0\mathbf{m}_1} \neq A_{\mathbf{m}_0}A_{\mathbf{m}_1}$. The scheme could be revised by redefining the associated matrix (then the scheme will not be based on octonions). For example, for a plain text message $m \in GF(q)$ one may define an associated matrix $A_m = \begin{pmatrix} m & 0 \\ \mathbf{b}^T & B \end{pmatrix} \in GF(q)^{8\times 8}$ with uniformly at random chosen $\mathbf{b} \in GF(q)^7$ and $B \in GF(q)^{7\times 7}$. Then it is straightforward that $A_{m_0m_1} \neq A_{m_0}A_{m_1}$ for all $m_0, m_1 \in GF(q)$. It follows that the resulting scheme is fully homomorphic with unlimited ciphertext additions and multiplications.

Alternatively, we may also use Yagisawa's [7] message encoding technique to make M.Enc fully homomorphic. That is, a message $m \in GF(q)$ is mapped to an octonion $m\mathbf{1} + r\mathbf{z}$ where $\mathbf{z}$ is a fixed octonion number with $\|\mathbf{z}\| = 0$ and $r \in GF(q)$ is randomly chosen. In other words, a message $m \in GF(q)$ is encrypted to $C_m = \texttt{M.Enc}(K, m) = K^{-1}A_{m\mathbf{1}+r\mathbf{z}}K \in GF^{8\times 8}$ where $A_{m\mathbf{1}+r\mathbf{z}}$ is the associated matrix for $m\mathbf{1} + r\mathbf{z}$ when $m\mathbf{1} + r\mathbf{z}$ is considered as an octonion number. For a received ciphertext C_m, one can first compute $A_{m\mathbf{1}+r\mathbf{z}} = \texttt{M.Dec}(K, C_m) = KC_mK^{-1}$. The plaintext message m can then be recovered by finding an octonion $\mathbf{u}$ such that $\|\mathbf{u}\| = 0$ and $\mathbf{1}A_{m\mathbf{1}+r\mathbf{z}} = m\mathbf{1} + \mathbf{u}$. Ciphertext addition and multiplication are carried out in the same way by using matrix addition and multiplication. In the following, we show that this revised scheme achieves fully multiplication homomorphism.

For $m_0, m_1, m_2, r_0, r_1, r_2 \in GF(q)$, it is straightforward to show that (for a proof, see Yagisawa [7])

$$(m_0\mathbf{1} + r_0\mathbf{z})(m_0\mathbf{1} + r_0\mathbf{z}) = m_0m_1\mathbf{1} + r_3\mathbf{z}$$

for some $r_3 \in GF(q)$. It follows that

$$\mathbf{m}_0(\mathbf{m}_1\mathbf{m}_2) = (\mathbf{m}_0\mathbf{m}_1)\mathbf{m}_2 = m_0m_1m_2\mathbf{1} + r_4\mathbf{z}$$

for some $r_4 \in GF(q)$. In a summary, we have

$$\begin{aligned}\texttt{M.Dec}(K, C_{m_2} C_{m_0 m_1}) &= \texttt{M.Dec}(K, K^{-1} A_{m_2 \mathbf{1} + r_2 \mathbf{z}} A_{m_1 \mathbf{1} + r_1 \mathbf{z}} A_{m_0 \mathbf{1} + r_0 \mathbf{z}} K) \\ &= \mathbf{1} A_{m_2 \mathbf{1} + r_2 \mathbf{z}} A_{m_1 \mathbf{1} + r_1 \mathbf{z}} A_{m_0 \mathbf{1} + r_0 \mathbf{z}} \\ &= m_0 m_1 m_2 \mathbf{1} + r_4 \mathbf{z} \\ &= \texttt{M.Dec}(K, C_{m_2 (m_0 m_1)})\end{aligned} \tag{2}$$

That is, for the revised encryption scheme, ciphertexts could be multiplied for unlimited times.

3 Dongxi Liu [4]'s Scheme

Liu [4] proposed a candidate fully homomorphic encryption scheme using linear algebra over $GF(q)$. Though the design in [4] is very complicated, we give a simple (equivalent) description of the protocol in [4]. From the simplified description, it is straightforward that the public evaluation keys leak all of the private key.

Let l, n be given numbers with $l \leq n-2$. It is recommended to use $n = 5$ and $l = 3$ in [4]. The protocol works as follows.

Key Setup.

- Choose random vectors $\mathbf{k} = [k_0, \cdots, k_n] \in GF(q)^{n+1}$ and $\Theta = [\theta_0, \cdots, \theta_{l-1}] \in GF(q)^l$.
- For each $m \in GF(q)$, let $\mathbf{c}_m = \texttt{ENC}(\mathbf{k}, m) = [c_0, \cdots, c_n] \in GF(q)^{n+1}$ such that $m = \mathbf{k} \cdot \mathbf{c}_m$ where $\cdot$ is the inner product of $\mathbf{k}$ and $\mathbf{c}_m$. That is, $\mathbf{k} \cdot \mathbf{c}_m = c_0 k_0 + c_1 k_1 + \cdots + c_n k_n$.
- Let $\Phi = [\texttt{ENC}(\mathbf{k}, \theta_0), \cdots, \texttt{ENC}(\mathbf{k}, \theta_{l-1}), \texttt{ENC}(\mathbf{k}, 1)]$.
- The private key is $\mathbf{k}$ and Θ.
- The public evaluation key is $\mathbf{pek} = \{\mathbf{p}_{i,j} = \texttt{ENC}(\mathbf{k}, k_i k_j) : 0 \leq i, j \leq n\}$

Encryption. For a message $\mathbf{m} \in GF(q)$, choose random $r_0, \cdots, r_l \in GF(q)$ with $m = r_0 \oplus r_1 \oplus \cdots \oplus r_l$. The ciphertext of m is $\mathbf{c}_m = (r_0 \cdot \texttt{ENC}(\mathbf{k}, \theta_0)) \oplus \cdots \oplus (r_{l-1} \cdot \texttt{ENC}(\mathbf{k}, \theta_{l-1})) \oplus (r_l \cdot \texttt{ENC}(\mathbf{k}, 1))$.

Decryption. For a received ciphertext $\mathbf{c}_m$, compute $m = \mathbf{k} \cdot \mathbf{c}_m$.

Ciphertext addition. The addition of two ciphertexts $\mathbf{c}_{m_0}$ and $\mathbf{c}_{m_1}$ is defined as the component wise addition $\mathbf{c}_{m_0 + m_1} = \mathbf{c}_{m_0} + \mathbf{c}_{m_1}$. That is, this is just the regular component wise vector addition.

Ciphertext multiplication. The multiplication of two ciphertexts $\mathbf{c}_{m_0} = [c_0, \cdots, c_n]$ and $\mathbf{c}_{m_1} = [c'_0, \cdots, c'_n]$ is defined as $\mathbf{c}_{m_0 m_1} = \sum_{i,j=0}^{n} c_i c_j \mathbf{p}_{i,j}$.

The correctness of the protocol could be easily verified (for details, it is referred to the original paper [4]. However, the protocol cannot be secure since the private key $\mathbf{k}$ could be trivially derived from the public evaluation key $\mathbf{pek}$. As an example, we can assume that $\mathbf{p}_{i,j} = [p_{i,j,0}, \cdots, p_{i,j,n}]$. Then we have the equations

$$\begin{aligned} & k_0 k_0 = p_{0,0,0} k_0 + \cdots + p_{0,0,n} k_n \\ & \cdots \\ & k_i k_j = p_{i,j,0} k_0 + \cdots + p_{i,j,n} k_n \\ & \cdots \\ & k_n k_n = p_{n,n,0} k_0 + \cdots + p_{nn,,n} k_n \end{aligned} \tag{3}$$

Using equation (3), one can easily obtain the private key $\mathbf{k}$ by constructing polynomial equations $f(k_i) = 0$ in one variable and then using the Euclidean algorithm to compute $\gcd(f(x), x^q - x)$ (or use Berlekamp's algorithm). For example, from the first equation, one can obtain an expression of k_n in terms of $k_0, \cdots, k_{n-1}$. By substituting this k_n into all remaining equations, one eliminates the occurrence of k_n from all remaining equations.

4 A scheme from ASIACCS 2014

Liu, Bertino, and Xun [5] introduced a fully homomorphic encryption scheme to carry out privacy preserving outsourced k-means clustering. In this section, we show that the FHE scheme in [5] is insecure either. The scheme in [5] works as follows.

Key Setup. Let $n \geq 3$. For $i < n$, choose random tuples $(k_i, s_i, t_i) \in GF(q)^3$ such that $k_0 \cdots k_{n-1} \neq 0$, $k_{n-1} + s_{n-1} + t_{n-1} \neq 0$, and there exists only one i_0 such that $t_{i_0} \neq 0$. The private key is $\mathtt{key} = \{(k_i, s_i, t_i) : 0 \leq i < n\}$.

Encryption. For a message $\mathbf{m} \in GF(q)$, choose random $r_0, \cdots, r_{n-1} \in GF(q)$ and compute the ciphertext $(c_0, \cdots, c_{n-1})$ as follows.

- Let $c_0 = k_0 t_0 m + s_0 r_{n-1} + k_0(r_0 - r_{n-1})$.
- For $1 \leq i \leq n-1$, let $c_i = k_i t_i v + s_i r_{n-1} + k_i(r_i - r_{i-1})$
- Let $c_{n-1} = (k_{n-1} + s_{n-1} + t_{n-1}) r_{n-1}$

Decryption. For a received ciphertext $(c_0, \cdots, c_{n-1})$, compute

$$m = \left(\sum_{i=0}^{n-2} (c_i - S s_i)/k_i \right) /T$$

where $T = \sum_{i=0}^{n-2} t_i$ and $S = c_{n-1}/(k_{n-1} + s_{n-1} + t_{n-1})$.

The above scheme is equivalent to the inner product encryption scheme. For example, we could set

$$\begin{aligned} \tau_i &= 1/k_i T \text{ for } 0 \leq i \leq n-2 \\ \tau_{n-1} &= -\sum_{i=0}^{n-2} \frac{s_i}{k_i T (k_{n-1} + s_{n-1} + t_{n-1})} \end{aligned} \tag{4}$$

Then the decryption circuit is just an inner product: $m = c_0\tau_0 + \cdots c_{n-1}\tau_{n-1}$. Since inner product function is trivially polynomially learnable, Brakerski [2]'s result shows that this scheme could not be fully homomorphic and secure at the same time.

5 A scheme from CCS 2012

Liu and Wang [6] introduced an additive homomorphic encryption scheme to design practical encrypted database query. The scheme is very similar to the schemes in Sections 3 and 4. It could easily be converted to an equivalent inner product encryption scheme as in Section 4. Thus the scheme could not be secure either.

Acknowledgment

I would like to thank Masahiro Yagisawa for some discussion on the protocols in [7].

References

1. John Baez. The octonions. *Bulletin of the American Mathematical Society*, 39(2):145–205, 2002.
2. Z. Brakerski. When homomorphism becomes a liability. In *Theory of Cryptography*, pages 143–161. Springer, 2013.
3. John H Conway and Derek A Smith. On quaternions and octonions. *AMC*, 10:12, 2003.
4. Dongxi Liu. Practical fully homomorphic encryption without noise reduction. Technical report, Cryptology ePrint Archive, Report 2015/468. http://eprint.iacr.org/2015/468, 2015.
5. Dongxi Liu, Elisa Bertino, and Xun Yi. Privacy of outsourced k-means clustering. In *Proceedings of the 9th ACM symposium on Information, computer and communications security*, pages 123–134. ACM, 2014.
6. Dongxi Liu and Shenlu Wang. Query encrypted databases practically. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 1049–1051. ACM, 2012.
7. Masahiro Yagisawa. Fully homomorphic encryption without bootstrapping. Technical report, Cryptology ePrint Archive, Report 2015/474. http://eprint.iacr.org/2015/474, 2015.