An Attack on a Fully Homomorphic Encryption Scheme

Yupu Hu[1] and Fenghe Wang[2]

[1] Telecommunication School, Xidian University, 710071 Xi'an, China
[2] Department of Mathematics and Physics Shandong Jianzhu University, 250101,Jinan,China
yphu@mail.xidian.edu.cn
fenghe2166@163.com

Abstract. In this paper we present an attack on a fully homomorphic encryption scheme on PKC2010. We construct a modified secret key, a modified decryption algorithm and a subset of the ciphertext space. When the ciphertext is from the subset, we can correctly decrypt it by our modified secret key and modified decryption algorithm. We also discuss when our modified decryption algorithm is efficient, and when the subset is not negligible.
Keywords: Fully Homomorphic Encryption, lattice-based PKC, cloud-computation.

1 Introduction: Bootstrapping-based FHE and attacks

Fully-homomorphic-Encryption (FHE) is a novel technique applied to cloud-computation. It is for such procedure. The user needs to obtain the output of ring-operations in plaintext space, but wants to compute nothing but only decryption. The sever implements "homomorphic operations in ciphertext space" of these operations in plaintext space. That is, the server is given corresponding ciphertexts of input plaintexts of these ring-operations, and computes corresponding ciphertext of output plaintext of these ring-operations. Then the server sends this output ciphertext to the user. The user then decrypts this output ciphertext to obtain original output plaintext. To be fully-homomorphic, these ring-operations in plaintext space must be of any depth.

Lattice-based PKC can achieve homomorphic encryption function for shallow ring-operations in plaintext space. This homomorphic encryption function is called somewhat homomorphic encryption function. But lattice-based PKC belongs to the class of error correction type PKC, so that the ciphertext must include noise. Deep ring-operations in plaintext space means deep homomorphic operations in ciphertext space, and noise will be greatly enhanced, resulting in wrong decryption. Therefore the major problem to achieve fully-homomorphic-encryption is to decrease the size of the noise. The frontier work to solve this problem belongs to Gentry [1], who presented a novel technique named bootstrapping. After that many fully-homomorphic-encryption schemes were presented [2-21], with major efforts on decreasing the computation complexity. Among them is a fully-homomorphic-encryption scheme [6], presented by Smart and Vercauteren, and on PKC2010. We call this paper SV10. SV10 scheme is one of efforts changing the ciphertexts from polynomials to integers. It has relatively small key and ciphertext sizes.

In this paper we present an attack on this fully homomorphic encryption scheme. In fact, our attack only aims at its "somewhat homomorphic encryption algorithm". We construct a modified secret key, a modified decryption algorithm and a subset of the ciphertext space. Whenever the ciphertext is from the subset, we can correctly decrypt it by our modified secret key and modified decryption algorithm. We also discuss when our modified decryption algorithm is efficient, and when the subset is not negligible. Our attack implies that it should be careful for designing fully homomorphic encryption over the integers.

The paper is organized as following. In section 2 we state the fully-homomorphic-encryption scheme presented by SV10, including its correctness. In section 3 we present our major proposition, which is the basis of our attack. In section 4 we present our attack, including modified secret key, modified decryption algorithm, and a subset of the ciphertext space. In this section we also discuss when our modified decryption algorithm is efficient, and when the subset is not negligible. Section 5 is the conclusion.

2 The FHE scheme of SV10

2.1 The Somewhat Homomorphic Encryption Algorithm

$\mathbb{Q}[x]$ and $\mathbb{Z}[x]$ are respectively the set of polynomials with rational and integer coefficients. Given a polynomial $g(x) = \sum_{i=0}^{t} g_i x^i \in \mathbb{Q}[x]$, $||g(x)||_\infty = \max\limits_{t=0,\cdots,t} |g_i|$. For a positive value r, $B_{\infty,N}(r)$ is such set of polynomials with integer coefficients:

$$B_{\infty,N}(r) = \{g(x) = \sum_{i=0}^{N-1} g_i x^i, ||g(x)||_\infty \leq r\}.$$

The scheme is parameterized by three values (N, η, μ). A typical set of parameters would be $(N, \eta, \mu) = (N, 2^{\sqrt{N}}, \sqrt{N}), N = 2^n$.

KeyGen(.):

1. Set the plaintext space to be $\{0, 1\}$. Choose a monic irreducible polynomial $F(x) \in \mathbb{Z}[x]$ of degree N.Typically we can take $F(x) = x^N + 1 = x^{2^n} + 1$.
2. Repeat:
   • $S(x) \leftarrow_R B_{\infty,N}(\eta/2)$.
   • $G(x) \leftarrow 1 + 2S(x)$.
   • $p \leftarrow resultant(G(x), F(x))$.
   -Until p is prime.
3. $D(x) \leftarrow gcd(G(x), F(x))$ over $\mathbb{F}_p[x]$
   -$D(x) \leftarrow gcd(G(x), F(x))$ over $\mathbb{F}_p[x]$
   -Let $\alpha \in \mathbb{F}_p$ denote the unique root of $D(x)$.
4. Apply the XGCD algorithm over $\mathbb{Q}[x]$ to obtain $Z(x) = \sum_{i=0}^{N-1} z_i x^i \in \mathbb{Z}[x]$ such that

$$Z(x)G(x) = p \mathrm{mod} F(x).$$

5. $B \leftarrow z_0 \mathrm{mod} 2p$
   -The public key is (p, α). The secret key B.

Encrypt(M, p, α):

1. Let $R(x) \leftarrow_R B_{\infty,N}(\mu/2)$.
2. Let $C(x) \leftarrow M + 2R(x)$.
3. Let $c \leftarrow C(\alpha) \mathrm{mod} p$.
   -Output c.

Decrypt(c, B): $M \leftarrow c - \lceil cB/P \text{mod} 2$
-Output M.
Add(c_1, c_2, p, α): $c_3 \leftarrow c_1 + c_2 \text{mod} p$.
-Output c_3.
Mult(c_1, c_2, p, α): $c_3 \leftarrow c_1 c_2 \text{mod} p$.
-Output c_3.

2.2 Some Notes

It should be pointed out that, although $Z(x)$ is obtained by the XGCD algorithm over $\mathbb{Q}[x]$, $Z(x)\mathbb{Z}(x)$ because $p = resultant(G(x), F(x))$. According to typical parameters, the ciphertext can be correctly decrypted. Again according to typical parameters, the scheme allows about d homomorphic multiplications over the ciphertext space, where $d \approx \frac{\log \frac{2^{\sqrt{N}}}{2\sqrt{N}}}{\log N\sqrt{N}}$. In other words, the product of not more than d ciphertexts can be correctly decrypted. The value of d increases with the value of N. For $N = 256, d = 0$. For $(N = 512, d = 1)$. For $N = 65536, d = 10$.

The scheme of SV10 uses both the somewhat homomorphic encryption algorithm and the bootstrapping technique to achieve FHE. Our attack only aims at the somewhat homomorphic encryption algorithm, so that we don't introduce the bootstrapping.

3 Our Major Proposition

3.1 The Proposition

Take a positive integer δ such that $\delta < N$. Take $\beta_k = 2\alpha^k (\text{mod} q), k = 0, 1, \cdots, N-1$. Define $N+2$ sets $\{A_0, A_1, \cdots, A_{N-1}, A_*, A_{**}\}$ as follows. For $k = 0, 1, 2, \cdots, N-1$,

$$A_k = (0, p) \bigcap \{ \bigcup_{i=0}^{+\infty} (2i\frac{p}{\beta_k} - \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}, 2i\frac{p}{\beta_k} + \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}) \}$$

if $\beta_k (\text{mod} 2) = 0$, and

$$A_k = (0, p) \bigcap \{ \bigcup_{i=0}^{+\infty} ((2i+1)\frac{p}{\beta_k} - \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}, (2i+1)\frac{p}{\beta_k} + \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}) \}$$

if $\beta_k (\text{mod} 2) = 1$.

$$A_* = (0, p) \bigcap \{ \bigcup_{i=0}^{+\infty} (2ip - \frac{p}{2(\delta+2)}, 2ip + \frac{p}{2(\delta+2)}) \}.$$

$$A_{**} = (0, p) \bigcap \{ \bigcup_{i=0}^{+\infty} (2i + 1 - \frac{1}{\mu\delta(\delta+2)}, 2i + 1 + \frac{1}{\mu\delta(\delta+2)}) \}.$$

It is clear that $A_* = (0, \frac{p}{2(\delta+2)})$, $A_0 = (0, \frac{p}{\mu(\delta+2)})$. For δ integers $k_1, k_2, \cdots, k_\delta$, such that $0 \leq k_1 < k_2 < \cdots < k_\delta \leq N-1$, define the intersection set

$$A(k_1, k_2, \cdots, k_\delta) = A_{k_1} \bigcap A_{k_2} \bigcap \cdots \bigcap A_{k_\delta} \bigcap A_* \bigcap A_{**},$$

and define the subset of the ciphertext space

$$C(k_1, k_2, \cdots, k_\delta) = \left\{C(\alpha)(\mathrm{mod}p) | C(x) = M + r_{k_1}x^{k_1} + r_{k_2}x^{k_2} + \cdots + r_{k_\delta}x^{k_\delta}, r_{k_i} even, |r_{k_i}| \le \mu, i = 1, \cdots, \delta\right\}$$

Proposition 1. *Take a real number B from the set $A(k_1, k_2, \cdots, k_\delta)$. Then any ciphertext c from $C(k_1, k_2, \cdots, k_\delta)$ can be correctly decrypted by the algorithm $c - \lfloor cB/p \rceil$.*

Proof. $c = M + (\frac{r_{k_1}}{2}\beta_{k_1} + \frac{r_{k_2}}{2}\beta_{k_2} + \cdots + \frac{r_{k_\delta}}{2}\beta_{k_\delta}) - up$, where $\frac{r_{k_i}}{2}$ is an integer, $|\frac{r_{k_2}}{2}| \le \frac{\mu}{2}, i = 1, 2, \cdots, \delta$. It is immediate that $0 \le u \le \frac{\mu}{2}\delta$.

Notice the definition of $A(k_1, k_2, \cdots, k_\delta)$. $0 \le \frac{MB}{p} \le \frac{1}{2(\delta+2)}$. $\frac{r_{k_i}\beta_{k_i}B}{2p}$ is in $(-\frac{1}{2(\delta+2)}, \frac{1}{2(\delta+2)})$ neighboring field of an integer, and that integer is modular 2 congruent with $r_{k_i}\beta_{k_i}uB$ is in $(-\frac{1}{2(\delta+2)}, \frac{1}{2(\delta+2)})$ neighboring field of an integer, and that integer is modular 2 congruent with up. From all of the above,

$$\begin{aligned} &c - \lfloor cB/p \rceil (\mathrm{mod}2) \\ &= M - \lfloor MB/p \rceil + r_{k_1}\beta_{k_1} - \lfloor r_{k_1}\beta_{k_1}B/p \rceil + r_{k_1}\beta_{k_1} - \lfloor r_{k_2}\beta_{k_2}B/p \rceil + \cdots + r_{k_1}\beta_{k_1} - \lfloor r_{k_\delta}\beta_{k_\delta}B/p \rceil \\ &- up + \lfloor uB \rceil (\mathrm{mod}2) = M. \end{aligned}$$

3.2 Time Cost for Finding a B from $A(k_1, k_2, \cdots, k_\delta)$

Suppose $\beta_{k_\delta} = 0(\mathrm{mod}2)$ ($\beta_{k_\delta} = 1(\mathrm{mod}2)$), for $i = 0, 1, \cdots$ check whether the interval $(2i\frac{p}{\beta_k} - \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}, 2i\frac{p}{\beta_k} + \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k})$ (the interval $((2i+1)\frac{p}{\beta_k} - \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k}, (2i+1)\frac{p}{\beta_k} + \frac{1}{\mu(\delta+2)}\frac{p}{\beta_k})$ intersects the set $\{A_0, A_1, \cdots, A_{\delta-1}, A_*, A_{**}\}$ until it intersects all of them. Then take any real number B from such intersection set.

Checking whether an interval intersects the set $\{A_{k_2}, A_{k_3}, \cdots, A_\delta, A_*, A_{**}\}$ is a quite simple task. On the other hand, $A_* = (0, \frac{p}{2(\delta+2)})$, and p is extremely large. The length of the set A_k is about $\frac{p}{\mu(\delta+2)}$. The length of the set A_{**} is about $\frac{p}{\mu\delta(\delta+2)}$. Then the length of the intersection set $A_{k_1} \bigcap A_{k_2} \bigcap \cdots \bigcap A_{k_\delta} \bigcap A_{**}$ is about $\frac{p}{\mu^{\delta+1}\delta(\delta+2)^{\delta+1}}$. This means that the length of the intersection set $A(k_1, k_2, \cdots, k_\delta) = A_{k_1} \bigcap A_{k_2} \bigcap \cdots \bigcap A_{k_\delta} \bigcap A_* \bigcap A_{**}$ is about $\frac{p}{2\mu^{\delta+1}\delta(\delta+2)^{\delta+2}}$, and $A(k_1, k_2, \cdots, k_\delta) \subset (0, \frac{p}{2(\delta+2)})$.

If $k_1 > 0$, the point of $A(k_1, k_2, \cdots, k_\delta)$ are randomly distributed within $(0, \frac{p}{2(\delta+2)})$.

If $k_1 = 0$, $A(k_1, k_2, \cdots, k_\delta) \subset (0, \frac{p}{\mu(\delta+2)})$, and the points of $A(k_1, k_2, \cdots, k_\delta)$ are randomly distributed within $(0, \frac{p}{\mu(\delta+2)})$. Therefore there should be a point of $A(k_1, k_2, \cdots, k_\delta)$ which is smaller than $O(\mu^{\delta+1}\delta(\delta+2)^{\delta+1})$, and checking at most $O(\mu^{\delta+1}\delta(\delta+2)^{\delta+1})$ intervals should find a real number B of $A(k_1, k_2, \cdots, k_\delta)$.

Suppose $N = 256$, then $\mu = \sqrt{N} = 16$. Take $\delta = 3$, $O(\mu^{\delta+1}\delta(\delta+2)^{\delta+1}) = O(16^4 \times 3 \times 5^4) \approx O(2^{27})$. Take $\delta = 9$, $O(\mu^{\delta+1}\delta(\delta+2)^{\delta+1}) = O(16^{10} \times 9 \times 11^{10}) \approx O(2^{78})$. Table 1 lists $n(\delta)$ the appropriate number of intervals for checking, with $N = 256$.

4 Our Attack

4.1 The Modified Decryption Algorithm

Suppose that for any δ integers $k_1, k_2, \cdots, k_\delta$ such that $0 \le k_1 < k_2 < \cdots < k_\delta \le N - 1$, we have chosen a real number $B(k_1, k_2, \cdots, k_\delta)$ from the set $A(k_1, k_2, \cdots, k_\delta)$. Our modified secret key is

Table 1. he Appropriate Number of Intervals for Checking (N=256)

δ	1	2	3	4	5	6
$n(\delta)$	$16^2 \times 3^2$	$16^3 \times 2 \times 4^3$	$16^4 \times 3 \times 5^4$	$16^5 \times 4 \times 6^5$	$16^6 \times 5 \times 7^6$	$16^7 \times 6 \times 8^7$
δ	7	8	9	10	11	12
$n(\delta)$	$16^8 \times 7 \times 9^8$	$16^9 \times 8 \times 10^9$	$16^{10} \times 9 \times 11^{10}$	$16^{11} \times 10 \times 12^{11}$	$16^{12} \times 11 \times 13^{12}$	$16^{13} \times 12 \times 14^{13}$

$B(k_1, k_2, \cdots, k_\delta) | 0 \le k_1 < k_2 < \cdots < k_\delta \le N-1$. For any $\delta - 1$ integers $l_1, l_2, \cdots, l_{\delta-1}$ such that $0 \le l_1 < l_2 < \cdots < l_{\delta-1} \le N-1$, define the set

$$S(l_1, l_2, \cdots, l_{\delta-1}) = \{B(k_1, k_2, \cdots, l_\delta) \supset \{l_1, l_2, \cdots, l_{\delta-1}\}\}.$$

Notice that $|S(l_1, l_2, \cdots, l_{\delta-1})| = N - \delta + 1$.

The modified decryption algorithm is as the follow. For a ciphertext c, compute

$$\{c - \lfloor cB(k_1, k_2, \cdots, k_\delta)/p \rceil (\mathrm{mod} 2) | 0 \le k_1 < k_2 < \cdots < k_\delta \le N-1\}.$$

If there is some $S(l_1, l_2, \cdots, l_{\delta-1})$, such that all values

$$\{c - \lfloor cB(k_1, k_2, \cdots, k_\delta)/p \rceil (\mathrm{mod} 2) | B(k_1, k_2, \cdots, k_\delta) \in S(l_1, l_2, \cdots, l_{\delta-1})\}$$

are same, take the plaintext as such same value. If there is no such $S(l_1, l_2, \cdots, l_{\delta-1})$, the decryption fails.

4.2 The Correctness

For a ciphertext c, the corresponding polynomial is $C(x) = M + 2R(x)$. If $R(x)$ has m non-zero coefficients, we say that the Hamming weight of c is m. Define $C(m)$ as the set of all ciphertexts with the Hamming weights not more than m. It is clear that $C(m) = \bigcup_{0 \le l_1 < l_2 < \cdots < l_m \le N-1}$, and that $C(0) \subset C(1) \subset \cdots \subset C(N)$. $C(N)$ is the ciphertext space.

Now we explain that our algorithm makes wrong decryption with a negligible probability for the ciphertexts from $C(\delta - 1)$.

Take any ciphertext $c \in C(\delta - 1)$, any $C(l_1, l_2, \cdots, l_{\delta-1}) \subset C(\delta - 1)$. If $c \in C(l_1, l_2, \cdots, l_{\delta-1})$, all value $\{c - \lfloor cB(k_1, k_2, \cdots, k_\delta)/p \rceil (\mathrm{mod} q) | B(k_1, k_2, \cdots, k_\delta) \in S(l_1, l_2, \cdots, l_{\delta-1})\}$ are the plaintext of c. If $c \notin C(l_1, l_2, \cdots, l_{\delta-1})$, value $\{c - \lfloor cB(k_1, k_2, \cdots, k_\delta)/p \rceil (\mathrm{mod} q) | B(k_1, k_2, \cdots, k_\delta) \in S(l_1, l_2, \cdots, l_{\delta-1})\}$ are random, therefore the probability they are same as the wrong value is $\frac{1}{2^{N-\delta+1}}$. There are at most $C_N^{\delta-1} - 1$ different $C(l_1, l_2, \cdots, l_{\delta-1})$ such that $c \notin C(l_1, l_2, \cdots, l_{\delta-1})$, therefore the probability c is decrypted as wrong value is not larger than $\frac{C_N^{\delta-1}-1}{2^{N-\delta+1}}$. For $N \ge 256, \delta \le 17$, this probability is negligible.

4.3 The Efficiency

Our modified secret key includes C_N^δ real numbers. Our modified decryption algorithm includes C_N^δ ordinary decryptions, and a judgment.

4.4 The Impact

Our modified decryption algorithm can correctly decrypt all ciphertexts in $C(\delta - 1)$. If in the encryption procedure, the step $R(x) \leftarrow_R B_{\infty,N}(\mu/2)$ is under uniform distribution, $C(\delta - 1)$ is negligible in the ciphertext space. However, the ciphertext with small Hamming weight will have great advantage in somewhat homomorphic operations, especially in multiplication homomorphic operations. If the step $R(x) \leftarrow_R B_{\infty,N}(\mu/2)$ is partial to the smaller Hamming weights $C(\delta - 1)$ is not negligible, and our attack is effective.

5 Conclusions

SV10 scheme is not as strong as some other fully homomorphic encryption schemes. Our attack at least threatens the distribution of the step $R(x) \leftarrow_R B_{\infty,N}(\mu/2)$ of the encryption procedure. Our attack implies that it should be careful for designing fully homomorphic encryption over the integers.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant Nos. 60970119, 61173151) and Science and Technology on Communication Security Laboratory (9140C110201110C1102). This work was also supported by Huawei Co. (YBCB2012026).

References

1. C. Gentry. Fully homomorphic encryption using ideal lattices. In STOC '09, pp. 169-178,2009.
2. Z. Brakerski and V. Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In FOCS 2011, pp. 97-106.
3. D. Stehle and R. Steinfeld. Faster fully homomorphic encryption. In Asiacrypt'10, pp. 377-394.
4. M. v. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In ASIACRYPT'10, pp.24-43.
5. Z. Brakerski and V. Vaikuntanathan. Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In CRYPTO'2011,pp. 505-524.
6. N. P. Smart, F. Vercauteren. Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In PKC 2010, 420-443, 2010.
7. C. Gentry, S. Halevi. Implementing Gentry's full homomorphic encryption scheme. In EUROCRYPT 2011, 129-148, 2011.
8. C. Gentry. Toward basing fully homomorphic encryption on worst-case hardness. In Crypto 2010, 116-137. 2010.
9. C. Gentry, S. Halevi. Fully homomorphic encryption without squashing using depth-3 arithmetic circuits. In FOCS 2011, 107-109.
10. Z. Brakerski, C. Gentry, V. Vaikuntanathan. Fully homomorphic without Bootstrapping. http://eprint.iacr.org/2011/277.
11. Jean-Sbastien Coron, Avradip Mandal, David Naccache, Mehdi Tibouchi. Fully Homomorphic Encryption over the Integers with Shorter Public Keys. In Crypto 2011. 487-504. 2011.
12. Yuanmi Chen, Phong Q. Nguyen. Faster Algorithms for Approximate Common Divisors: Breaking Fully-Homomorphic-Encryption Challenges over the Integers. In Eurocrypto 2012, 502-519. (http://eprint.iacr.org/2011/436)
13. Craig Gentry, Shai Halevi and Vinod Vaikuntanathan. i-Hop Homomorphic Encryption and Rerandomizable Yao Circuits. In Crypto 2011.155-172.
14. Zvika Brakerski. Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. http://eprint.iacr.org/2012/078.
15. Junfeng Fan and Frederik Vercauteren. Somewhat Practical Fully Homomorphic Encryption. http://eprint.iacr.org/2012/114.
16. Craig Gentry and Shai Halevi and Nigel P. Smart. Fully Homomorphic Encryption with Polylog Overhead. In Eurocrypto 2012, 465-482. (http://eprint.iacr.org/2011/566).

17. P. Scholl and N.P. Smart. Improved Key Generation For Gentry's Fully Homomorphic Encryption Scheme. http://eprint.iacr.org/2011/471
18. Steven Myers and Mona Sergi and abhi shelat. Threshold Fully Homomorphic Encryption and Secure Computation. http://eprint.iacr.org/2011/454.
19. Loftus and A. May and N.P. Smart and F. Vercauteren. On CCA-Secure Fully Homomorphic Encryption. http://eprint.iacr.org/2010/560.
20. Craig Gentry and Shai Halevi and Nigel P. Smart. Homomorphic Evaluation of the AES Circuit. http://eprint.iacr.org/2012/099.
21. Jean-Sbastien Coron, David Naccache, Mehdi Tibouchi. Public-key Compression and Modulus Switching for Fully Homomorphic Encryption over the Integers. In Eurocrypto 2012446-464 2012.