A Comment on Gu Map-1*

Yupu Hu and Huiwen Jia

ISN Laboratory, Xidian University, 710071 Xi'an, China
yphu@mail.xidian.edu.cn

Abstract. Gu map-1 is a modified version of GGH map. It uses same ideal lattices for constructing the trapdoors, while the novelty is that no encodings of zero are given. In this short paper we show that Gu map-1 cannot be used for the instance of witness encryption (WE) based on the hardness of 3-exact cover problem. That is, if Gu map-1 is used for such instance, we can break it by soving a combined 3-exact cover problem. The reason is just that no encodings of zero are given.

Keywords: Multilinear maps, GGH map, Gu map-1, Multi-party key exchange (MPKE), Witness encryption (WE), Lattice based cryptography.

1 Introduction: Background and Our Comment

Multilinear map is a novel primitive. It is the solution of a long-standing open problem [1], and has many novel cryptographic applications, such as multi-party key exchange (MPKE) [2], witness encryption (WE) [3–9], obfuscation [7–10], and so on. It also has several advantages in traditional cryptographic area [3], such as IBE, ABE, broadcasting Encryption, and so on. The first and the major candidate of multilinear map is GGH map [2, 11]. We presented efficient attack on GGH map for given encodings of zero [12], therefore we broke GGH multi-party key exchange (MPKE) and GGH instance of witness encryption (WE) based on the hardness of 3-exact cover problem. Gu map-1 [13] is a modified version of GGH map. It uses same ideal lattices for constructing the trapdoors, while the novelty is that no encodings of zero are given. This novelty makes it successfully form MPKE scheme, and avoid our attack.

In this short paper we show that Gu map-1 cannot be used for the instance of witness encryption (WE) based on the hardness of 3-exact cover problem. That is, if Gu map-1 is used for such instance, we can break it by solving a combined 3-exact cover problem. The reason is just that no encodings of zero are given. In other words, the reason is that there is no randomizer. The instance of WE based on the hardness of 3-exact cover problem is a strong application of multilinear map, and it has strong requirement.

* The work was supported in part by Natural Science Foundation of China under Grant 60833008

2 Gu Map-1

2.1 Setting Parameters

We define the integers by $\mathbb{Z}$. We specify that n-dimensional vectors of $\mathbb{Z}^n$ are row vectors. We consider the $2n$'th cyclotomic polynomial ring $R=\mathbb{Z}[X]/(X^n+1)$, and identify an element $u\in R$ with the coefficient vector of the degree-$(n-1)$ integer polynomial that represents u. In this way, R is identified with the integer lattice $\mathbb{Z}^n$. We also consider the ring $R_q=R/qR=\mathbb{Z}_q[X]/(X^n+1)$ for a (large enough) integer q. Obviously, addition in these rings is done component-wise in their coefficients, and multiplication is polynomial multiplication modulo the ring polynomial X^n+1. For $u\in R$, we denote $Rot(u)=\begin{bmatrix} u_0 & u_1 & \cdots & u_{n-1} \\ -u_{n-1} & u_0 & \cdots & u_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ -u_1 & -u_2 & \cdots & u_0 \end{bmatrix}\in\mathbb{Z}^{n\times n}$.

Because Gu map-1 scheme uses the GGH construction [2, 11] as the basic component, parameter setting is set as that of GGH to conveniently describe and compare. Let λ be the security parameter, K the multilinearity level, n the dimension of elements of R. Concrete parameters are set as $\sigma=\sqrt{\lambda n}$, $\sigma'=\lambda n^{1.5}$, $\sigma^*=2^\lambda$, $q\geq 2^{8K\lambda}n^{O(K)}$, $n\geq\widetilde{O}(K\lambda^2)$, $\tau=O(n^2)$.

2.2 Instance Generation

(1) Choose a prime $q\geq 2^{8K\lambda}n^{O(K)}$.

(2) Choose an element $g\leftarrow D_{\mathbb{Z}^n,\sigma}$ in R so that $\|g^{-1}\|\leq n^2$. In other words, g is "very small".

(3) Choose elements $a_i,e_i\leftarrow D_{\mathbb{Z}^n,\sigma}$, $b_i\leftarrow D_{\mathbb{Z}^n,\sqrt{q}}$, $i=1,\cdots,\tau$ in R. In other words, a_i,e_i are "very small", while b_i is "somewhat small".

(4) Choose a random element $z\in R_q$. In other words, $z\in R_q$ is never small.

(5) Choose two matrices $T,S\leftarrow D_{\mathbb{Z}^{n\times n},\sigma}$. In other words, T and S are "very small".

(6) Set $Y_i=\left[TRot(\frac{a_ig+e_i}{z})T^{-1}\right]_q$, $P_{zt,i}=\left[TRot(\frac{z^K(b_ig+e_i)}{g})S\right]_q$, $i=1,\cdots,\tau$.

(7) Output the public parameters $\{q,\{Y_i,P_{zt,i}\},i=1,\cdots,\tau\}$.

(8) Generating level-1 encodings. A user generates his secret $d\leftarrow D_{\mathbb{Z}^n,\sigma^*}$ in R, then publishes $U=\left[\sum_{i=1}^{\tau}d_iY_i\right]_q=\left[TRot(\frac{\sum_{i=1}^{\tau}d_i(a_ig+e_i)}{z})T^{-1}\right]_q$. U is level-1 encoding of the secret d.

(9) Generating level-K decoding factors. After the user generating his secret d, he secretly computes $V=\left[\sum_{i=1}^{\tau}d_iP_{zt,i}\right]_q=\left[TRot(\frac{z^K\sum_{i=1}^{\tau}d_i(b_ig+e_i)}{g})S\right]_q$. V is level-K decoding factor of the secret d.

2.3 An Application: Multi-party Key Exchange (MPKE)

Suppose that $K+1$ users want to generate a common shared key by public discussion. To do so, each user k generates his secret $d_k\leftarrow D_{\mathbb{Z}^\tau,\sigma^*}$ in R, publishes level-1 encoding $U_k=\left[\sum_{i=1}^{\tau}d_{k,i}Y_i\right]_q=\left[TRot(\frac{\sum_{i=1}^{\tau}d_{k,i}(a_ig+e_i)}{z})T^{-1}\right]_q$, and secretly computes level-K decoding factor $V_k = \left[\sum_{i=1}^{\tau} d_{k,i} P_{zt,i}\right]_q = \left[TRot(\frac{z^K \sum_{i=1}^{\tau} d_{k,i}(b_i g+e_i)}{g}) S\right]_q$, $k = 1, \cdots, K+1$. Then each user k can compute common shared key, which is high-order bits of $\left[V_k \prod_{j \neq k} U_j\right]_q$.

3 Another Application: the Instance of Witness Encryption (WE) Based on the Hardness of 3-Exact Cover Problem

3.1 3-Exact Cover Problem

Let K be a multiple of 3. A subset of $\{1, \cdots, K\}$ including 3 elements is called a piece. $K/3$ pieces without intersection are called a 3-exact cover of $\{1, \cdots, K\}$. 3-exact cover problem is, for given $O(K^2)$ pieces, to find a 3-exact cover of $\{1, \cdots, K\}$.

3.2 Gu Instance of WE Based on the Hardness of 3-Exact Cover Problem

Let K be a multiple of 3. The public key includes the public parameters $\{q, \{Y_i, P_{zt,i}\}, i = 1, \cdots, \tau\}$ and a group of $O(K^2)$ pieces, while the secret key is EC, a 3-exact cover of $\{1, \cdots, K\}$ hidden into this group.

Encryption

(1) The encrypter generates the $d_k \leftarrow D_{\mathbb{Z}^\tau, \sigma^*}$ in R. Then he computes level-1 encoding of d_k: $U_k \left[\sum_{i=1}^{\tau} d_{k,i} Y_i\right]_q = \left[TRot(\frac{\sum_{i=1}^{\tau} d_{k,i}(a_i g+e_i)}{z}) T^{-1}\right]_q$, $k = 1, \cdots, K+1$. Then he can compute encryption key, which is high-order bits of $\left[\prod_{k=1}^{K} U_k P_{zt,1}\right]_q$.

(2) The encrypter uses this encryption key and any encryption algorithm, to encrypt his plaintext M into ciphertext C.

(3) The encrypter hides encryption key into pieces. For each piece $\{i_1, i_2, i_3\}$, he computes level-3 encoding of $\{d_{i_1}, d_{i_2}, d_{i_3}\}$, which we denote $U_{\{i_1,i_2,i_3\}}$.

(4) The encrypter publishes all level-3 encodings. Therefore the final ciphertext is C and all $U_{\{i_1,i_2,i_3\}}$.

Decryption

The decrypter is anyone who knows the hidden 3-exact cover of $\{1, \cdots, K\}$, EC. On receiving C and all $U_{\{i_1,i_2,i_3\}}$, he computes $\left[\prod_{\{i_1,i_2,i_3\} \in EC} U_{\{i_1,i_2,i_3\}} P_{zt,1}\right]_q$. Then encryption key is its high-order bits, and he can decrypt C into the plaintext M.

3.3 A Note

For GGH map, $U_{\{i_1,i_2,i_3\}} = \Big[U_{i_1}U_{i_2}U_{i_3}+e_{\{i_1,i_2,i_3\}}\Big]_q$, where $e_{\{i_1,i_2,i_3\}}$ is a radomizer, which is an encoding of zero. However, for Gu map-1 we cannot obtain such radomizer, so that we can only have $U_{\{i_1,i_2,i_3\}} = \Big[U_{i_1}U_{i_2}U_{i_3}\Big]_q$.

4 Breaking the Instance by Solving a Combined 3-Exact Cover Problem

Suppose $\{i_1, i_2, i_3\}$ is not from the group of $O(K^2)$ public pieces. If $\{i_1, i_2, i_3\} = \{j_1, j_2, j_3\} \cup \{k_1, k_2, k_3\} - \{l_1, l_2, l_3\}$, where $\{j_1, j_2, j_3\}$, $\{k_1, k_2, k_3\}$ and $\{l_1, l_2, l_3\}$ are from the group of $O(K^2)$ public pieces, we call $\{i_1, i_2, i_3\}$ a combined piece. In this case we can compute $U_{\{i_1,i_2,i_3\}} = \Big[(U_{j_1}U_{j_2}U_{j_3})(U_{k_1}U_{k_2}U_{k_3})(U_{l_1}U_{l_2}U_{l_3})^{-1}\Big]_q$. If $\{i_1, i_2, i_3\} = \{j_1, j_2, j_3\} \cup \{k_1, k_2, k_3\} - \{l_1, l_2, l_3\}$ where $\{j_1, j_2, j_3\}$, $\{k_1, k_2, k_3\}$ and $\{l_1, l_2, l_3\}$ are combined pieces, we call $\{i_1, i_2, i_3\}$ a second order combined pieces. In this case we can also compute $U_{\{i_1,i_2,i_3\}}$. According such procedure, we can compute $U_{\{i_1,i_2,i_3\}}$ for any subset $\{i_1, i_2, i_3\}$. We can randomly construct a "3-exact cover", EC$'$, which is composed of pieces and combined pieces and second order combined pieces an so on. So that encryption key is high order bits of $\Big[\prod_{\{i_1,i_2,i_3\}\in EC} U_{\{i_1,i_2,i_3\}} P_{zt,1}\Big]_q$.

5 Our Comment

It may be argued that Gu map-1 uses hidden randomizers. But we hope that, for $1 \leq l \leq K$, a level-l encoding of $\{d_1, \cdots, d_l\}$ is the sum of a randomizer and the product of level-1 encodings of $d_1, \cdots, d_l$. This is a necessary condition of several important applications of multilinear map. However Gu map-1 does not satisfy this condition. Besides, Gu map-2 [14] does not either.

References

1. Boneh, D., Silverberg, A.: Applications of Multilinear Forms to Cryptography. Contemporary Mathematics. 324, 71–90 (2003)
2. Garg, S., Gentry, C., Halevi, S.: Candidate Multilinear Maps from Ideal Lattices. In: Johansson, T., Nguyen, P.Q. (ed.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 181–184. Springer, Heidelberg (2013)
3. Garg, S., Gentry, C., Sahai, A., Waters, B.: Witness Encryption and its Applications. In: STOC (2013)
4. Gentry, C., Lewko, A., Waters, B.: Witness Encryption from Instance Independent Assumptions. In: Garay, J.A., Gennaro, R. (ed.) CRYPTO 2014. LNCS, vol. 8616, pp. 426–443. Springer, Heidelberg (2014)

5. Arita, S., Handa, S.: Two Applications of Multilinear Maps: Group Key Exchange and Witness Encryption. In: Proceedings of the 2nd ACM workshop on ASIA public-key cryptography(ASIAPKC '14). ACM, New York, NY, USA, pp. 13–22 (2014)
6. Bellare, M., Hoang V.T.: Adaptive Witness Encryption and Asymmetric Password-Based Cryptography. Cryptology ePrint Archive, Report 2013/704 (2013)
7. Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate Indistinguishability Obfuscation and Functional Encryption for all Circuits. In: FOCS (2013)
8. Goldwasser, S., Kalai, Y.T., Popa, R.A., Vaikuntanathan, V., Zeldovich, N.: How to Run Turing Machines on Encrypted Data. In: Canetti, R., Garay, J.A. (ed.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 536–553. Springer, Heidelberg (2013)
9. Garg, S., Gentry, C., Halevi, S., Wichs, D.: On the Implausibility of Differing-Inputs Obfuscation and Extractable Witness Encryption with Auxiliary Input. In: Garay, J.A., Gennaro, R. (ed.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 518–535. Springer, Heidelberg (2014)
10. Boyle, E., Chung, K.-M., Pass, R.: On Extractability (a.k.a. Differing-Input) Obfuscation. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 52–73. Springer, Heidelberg (2014)
11. Langlois, A., Stehlé, D., Steinfeld, R.: GGHLiteMore Efficient Multilinear Maps from Ideal Lattices. In: Nguyen, P.Q., Oswald, E. (ed.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 239–256. Springer, Heidelberg (2014)
12. Hu, Y., Jia, H.: Cryptanalysis of GGH Map. Cryptology ePrint Archive, Report 2015/301 (2015)
13. Gu, C.: Multilinear Maps Using Ideal Lattices without Encodings of Zero. Cryptology ePrint Archive, Report 2015/023 (2015)
14. Gu, C.: Ideal Multilinear Maps Based on Ideal Lattices. Cryptology ePrint Archive, Report 2015/269 (2015)