# The Importance of the Public Global Parameter On Ring-LWE problem-based Key Encapsulation Mechanisms

**Reynaldo C. Villena & Routo Terada**

Institute of Mathematics and Statistics – University of São Paulo – SP – Brazil

reynaldo@ime.usp.br, rt@ime.usp.br

## Abstract

*There are cryptographic systems that are secure against attacks by both quantum and classical computers. Some of these cryptographic systems are the Key Encapsulation Mechanisms (KEM) based on Ring-LWE problem. Some Ring-LWE problem-based KEMs include a public global parameter that is random and uniformly chosen. This parameter is used to generate a public key using in the process one secret key.*
*In this work, we analyze some values of the public global parameter that leak information about the secret key.*

## 1. Preliminaries

### 1.1 Mathematical Notations

For an integer $q \geq 1$, let $\mathbb{Z}_q$ be the residue class ring modulo $q$ and $\mathbb{Z}_q = \{0, ..., q-1\}$. Let $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n+1)$ denote the polynomial ring modulo $x^n+1$ where the coefficients are in $\mathbb{Z}_q$. The operations (addition and multiplication) of the elements in $\mathcal{R}_q$ are according to those of polynomials.
The integer $\lfloor x \rceil$ is defined as $\lfloor x + \frac{1}{2} \rfloor \in \mathbb{Z}$.
**Centered Binomial Distribution** We define centered binomial distribution $\psi_\eta$ as follows: sample$(a_1, ..., a_\eta, b_1, ..., b_\eta) \leftarrow \{0,1\}^{2\eta}$ and output $\sum_{i=1}^{\eta}(a_i - b_i)$.
The samples are in the interval $[-\eta, \eta]$.
**Discrete Gaussian Distribution** It is defined by $(\chi_\sigma)$ where is assigned a weight proportional $\exp\left(\frac{-x^2}{2\sigma^2}\right)$ to all integer $x$ where $\sigma \in \mathbb{R}$ is a standard deviation.

### 1.2 The Ring-LWE Problem

The Ring-LWE problem fixes a power of two $n$ and modulus $q$. For $\mathbf{s} \in \mathcal{R}_q$ called as secret, the Ring-LWE distribution $A_{\mathbf{s},\chi}$ over $\mathcal{R}_q \times \mathcal{R}_q$ is sampled by choosing $\mathbf{a} \in \mathcal{R}_q$ uniformly at random, choosing $\mathbf{e} \rightarrow \chi_\sigma^n$, and outputting $(\mathbf{a}, \mathbf{a}.\mathbf{s} + \mathbf{e})$. One version of the Ring-LWE problem is the Search Ring-LWE.
**Search Ring-LWE**$_{q,\chi,k}$**:** Given $k$ independent samples $(\mathbf{a}_i, \mathbf{b}_i) \in \mathcal{R}_q \times \mathcal{R}_q$ drawn from $A_{s,\chi}$ for a uniformly random $\mathbf{s} \in \mathcal{R}_q$ (fixed for all samples), find $\mathbf{s}$.
Next, we explain two key encapsulation mechanisms to understand the importance of the public global parameter.

### 1.3 Key Encapsulation Mechanisms based on Ring-LWE

#### 1.3.1 Ring-LWE KEM

Common parameter: $\mathbf{a} \xleftarrow{\$} \mathcal{R}_q$

| **Alice** | | **Bob** |
|---|---|---|
| $\mathbf{s}_A, \mathbf{e}_A \xleftarrow{\$} \chi_\sigma^n$ | | |
| $\mathbf{p}_A \leftarrow \mathbf{a}.\mathbf{s}_A + \mathbf{e}_A$ | $\xrightarrow{\mathbf{p}_A}$ | $\mathbf{s}_B, \mathbf{e}_B, \mathbf{e}'_B \xleftarrow{\$} \chi_\sigma^n$ |
| | | $\mathbf{p}_B \leftarrow \mathbf{a}.\mathbf{s}_B + \mathbf{e}_B$ |
| | | $\mathbf{s}_{K_B} \xleftarrow{\$} \{0,1\}^n$ |
| | | $\mathbf{k} \leftarrow \lfloor \frac{q}{2} \rceil \mathbf{s}_{K_B}$ |
| $\mathbf{c}' \leftarrow \mathbf{c} - \mathbf{p}_B.\mathbf{s}_A$ | $\xleftarrow{(\mathbf{p}_B, \mathbf{c})}$ | $\mathbf{c} \leftarrow \mathbf{p}_A.\mathbf{s}_B + \mathbf{e}'_B + \mathbf{k}$ |
| $\mathbf{s}_{K_A} \leftarrow \left\lceil \mathbf{c}'.\frac{2}{q} \right\rfloor \mod 2$ | | |

**Figure 1:** *Ring-LWE KEM*

## 2. Bad Values for the Public Global Parameter

**Case 1:** (Trivial Case) Let $\mathbf{a}$ be a polynomial with degree 0 (an integer) with value $m$. This case can happen but with a negligible probability.
Alice generates her public key $\mathbf{p}_A = \mathbf{s}_A.\mathbf{a} + \mathbf{e}_A$. The integer $m$ multiplies each coefficient of $\mathbf{s}_A$ and its respective error $\mathbf{e}_A$ is added. We have:

$$\mathbf{p}_A[i] = \mathbf{s}_A[i].m + \mathbf{e}_A[i] \qquad \text{for } 0 \leq i \leq n-1$$

Eve can notice the integer value of $\mathbf{a} = m$ and recover $\mathbf{s}_A$ applying $\lceil . \rfloor$ on $\frac{\mathbf{p}_A}{m}$.

$$\left\lceil \frac{\mathbf{p}_A[i]}{m} \right\rfloor = \left\lceil \mathbf{s}_A[i] + \frac{\mathbf{e}_A[i]}{m} \right\rfloor = \mathbf{s}_A[i] + \left\lceil \frac{\mathbf{e}_A[i]}{m} \right\rfloor$$

Where the $i$-th coefficient of $s_A$ can be retrieved with no error if $-\frac{1}{2} < \frac{\mathbf{e}_A[i]}{m} < \frac{1}{2}$.
On NewHope, the value $\mathbf{e}_A \in \psi_8^n$ ($-8 \leq \mathbf{e}_A[i] \leq 8$), therefore $m$ should be greater and equal to 17 ($m \geq 17$) because $-\frac{1}{2} < \frac{\mathbf{e}_A[i]}{m} < \frac{1}{2}$.
**Note:** The value of $\mathbf{a}$ is a polynomial where each coefficient is selected uniformly at random in $\mathbb{Z}_q$, therefore the value of $\mathbf{a}$ being an integer would be suspicious for the participants. Alice and Bob can deny to share a secret using this suspect value of $\mathbf{a}$.
**Case 2:** Let $\mathbf{a}$ be the public global parameter and we define a polynomial $\mathbf{c} \in \mathcal{R}_q$ such that $\mathbf{a}.\mathbf{c} = m$ where $m$ is the integer mentioned before.
Alice generates her public key $\mathbf{p}_A = \mathbf{s}_A.\mathbf{a} + \mathbf{e}_A$ and sends it to Bob. Eve takes $\mathbf{p}_A$ and multiplies by $\mathbf{c}$. The integer $m$ multiplies each coefficient of $\mathbf{s}_A$.

$$\mathbf{p}_A.\mathbf{c} = \mathbf{s}_A.\mathbf{a}.\mathbf{c} + \mathbf{e}_A.\mathbf{c} = \mathbf{s}_A.m + \mathbf{e}_A.\mathbf{c} \qquad \text{because } \mathbf{a}.\mathbf{c} = m$$

Eve applies $\lceil . \rfloor$ to $\frac{\mathbf{p}_A.\mathbf{c}}{m}$

$$\left\lceil \frac{(\mathbf{p}_A.\mathbf{c})[i]}{m} \right\rfloor = \mathbf{s}_A[i] + \left\lceil \frac{(\mathbf{e}_A.\mathbf{c})[i]}{m} \right\rfloor \qquad \text{for } 0 \leq i \leq n-1$$

and can retrieve the $i$-th coefficient of $\mathbf{s}_A$ if $-\frac{1}{2} < \frac{(\mathbf{e}_A.\mathbf{c})[i]}{m} < \frac{1}{2}$. The result $\mathbf{e}_A.\mathbf{c}$ should be a small polynomial where each coefficient of $\mathbf{e}_A.\mathbf{c}$ divided by $m$ should be between -0.5 and 0.5. One way to ensure this is to make the polynomial $\mathbf{c}$ belong to Gaussian or Centered Binomial Distribution. Because a multiplication between two polynomials which belong to Gaussian or Centered Binomial Distribution results in a small polynomial where its coefficients have an expected value equal to 0.
On NewHope, the value $\mathbf{e}_A \in \psi_8^n$ and $\mathbf{e}_A.\mathbf{c}$ should be small therefore $\mathbf{c}$ should belong to $\psi_\mu^n$ where $\mu$ is a small integer. Therefore the parameter $\mathbf{a}$ that leaks information about secret keys can be generated using the formula $\mathbf{a} = m(\psi_\mu^n)^{-1}$.
**Note:** Alice and Bob can calculate $\mathbf{a}^{-1}$ and multiply by all integers in $\mathbb{Z}_q$ (brute force to determine the value of $m$). If the result is a small polynomial then it is possible that $\mathbf{a}$ leaks information because $\mathbf{c} \in \psi_\mu^n$ and $\mathbf{c} = \mathbf{a}^{-1}.m$.
**Case 3**: (Adding error to Case 2)
Let $\mathbf{a}$ be the public global parameter and we define a polynomial $\mathbf{c} \in \mathcal{R}_q$ such that $\mathbf{a}.\mathbf{c} = m + \psi_\nu^n$ where $\psi_\nu^n$ is a small polynomial and $\nu$ is a small integer.
Alice generates her public key $\mathbf{p}_A = \mathbf{s}_A.\mathbf{a} + \mathbf{e}_A$ and sends it to Bob. Eve takes $\mathbf{p}_A$ and multiplies by $\mathbf{c}$. The integer $m$ multiplies each coefficient of $s_A$.

$$\begin{aligned}\mathbf{p}_A.\mathbf{c} &= = \mathbf{s}_A.(m + \psi_\nu^n) + \mathbf{e}_A.\mathbf{c} \qquad \text{because } \mathbf{a}.\mathbf{c} = m + \psi_\nu^n \\ &= \mathbf{s}_A.m + \mathbf{s}_A.\psi_\nu^n + \mathbf{e}_A.\mathbf{c}\end{aligned}$$

Eve applies $\lceil . \rfloor$ to $\frac{\mathbf{p}_A.\mathbf{c}}{m}$

$$\left\lceil \frac{(\mathbf{p}_A.\mathbf{c})[i]}{m} \right\rfloor = \mathbf{s}_A[i] + \left\lceil \frac{(\mathbf{s}_A.\psi_\nu^n + \mathbf{e}_A.\mathbf{c})[i]}{m} \right\rfloor \qquad \text{for } 0 \leq i \leq n-1$$

and can retrieve the $i$-th coefficient of $\mathbf{s}_A$ if $-\frac{1}{2} < \frac{(\mathbf{s}_A.\psi_\nu^n + \mathbf{e}_A.\mathbf{c})[i]}{m} < \frac{1}{2}$.
Note that the result $\mathbf{s}_A.\psi_\nu^n + \mathbf{e}_A.\mathbf{c}$ should be a small polynomial therefore the polynomial $\mathbf{c}$ should belong to Gaussian or Centered Binomial Distribution.
On NewHope, the values $\mathbf{s}_A, \mathbf{e}_A \in \psi_8^n$. By definition $\mathbf{a}.\mathbf{c} = m + \psi_\nu^n$, a public global parameter $\mathbf{a}$ can be generated using the formula $\mathbf{a} = (m + \psi_\nu^n)\mathbf{c}^{-1}$ where $\mathbf{c}$ should belong to the Centered Binomial Distribution ($\mathbf{c} \in \psi_\mu^n$).
**Note:** In this case, we have $m = \mathbf{a}.\mathbf{c} - \psi_\nu^n$ where $\mathbf{a}$ and $m$ are public (for $m$ we can use brute force). The value $\psi_\nu^n$ is an error (small) polynomial and the value $\mathbf{c}$ is unknown. It looks like the Search Ring-LWE problem where $\mathbf{c}$ is the secret. Therefore Alice and Bob do not have knowledge about $\mathbf{a}$ (being generated by $\mathbf{c}$) and its possibility of leaking information.
The same process can be applied to retrieve Bob's secret $\mathbf{s}_B$ too in all cases.

## 3. Experiments

For our experiments, we work with parameters $n = 1024, q = 12289$ (parameters of NewHope) and set $m = \frac{q}{17} = 722$ that allows a maximum margin or error. For cases 2 and 3, it was generated 100 values of $\mathbf{a} = m.(\psi_\mu^n)^{-1}$ for $\mu = \{1, 2, 4, 8, 16\}$ and $\mathbf{a} = (m + \psi_\nu^n).(\psi_\mu^n)^{-1}$ where $\mu = \nu$ for $\mu = \{1, 2, 4, 8, 16\}$, respectively. And for each value $\mathbf{a}$, 100 public keys were generated making a total of 100000 experiments. Each public key was multiplied by its respective $\mathbf{c}$ and divided by $m$. Applying rounding function $\lceil . \rfloor$ to this result, we retrieve some coefficients of the secret key. Each experiment takes at most 0.3 seconds.

| | Case 2 | | | | | Case 3 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Value of $\mu$ | 1 | 2 | 4 | 8 | 16 | 1 | 2 | 4 | 8 | 16 |
| Recovered complete keys | 10000 | 10000 | 9284 | 144 | 0 | 10000 | 9372 | 9 | 0 | 0 |
| Avg. recovered coefficients (%) | 100.0 | 100.0 | 99.9 | 99.5 | 95.5 | 100.0 | 99.9 | 99.5 | 95.4 | 84.2 |
| Max. # of wrong coefficients | 0 | 0 | 2 | 19 | 90 | 0 | 3 | 17 | 88 | 218 |

**Table 1:** *Results of experiments*

In both cases for $\mu = 1, 2$ we retrieve 39372 of 40000 secret keys with zero coefficient errors (giving a success of 98.4 %). And that in the remaining 628 experiments there was an error in at most 3 of 1024 coefficients of the secret key.

## 4. Concluding remarks

We exposed some values of the public global parameter $\mathbf{a}$ that leak information about secret keys. Thus, there is a big responsibility how the public global parameter $\mathbf{a}$ is generated. If $\mathbf{a}$ has a value that leaks information (selected deliberately or not), then the secrets are exposed. Therefore the great and open question is " How to know when the value of the public global parameter $\mathbf{a}$ may or may not leak information about the secret?". In our experiments for cases 2 and 3, the generated values $\mathbf{a}$ that leak information always have at least 10 repeated coefficients. Therefore, it is recommended to be careful with values of parameter $\mathbf{a}$ that have repeated coefficients.
**Acknowledgement** This paper was partially funded by the project INCT of the Future Internet for Smart Cities: FAPESP proc. 2014/50937-1 / CNPq proc. 465446/2014-0.

## References

[Alkim et al. 2017] Alkim, E., Avanzi, R. M., Bos, J. W., Ducas, L., de la Piedra, A., Pöppelmann, T., and Schwabe, P. (2017). Newhope algorithm specifications and supporting documentation.

[Barreto et al. 2016] Barreto, P. S., Longa, P., Naehrig, M., Ricardini, J. E., and Zanon, G. (2016). Sharper ring-lwe signatures. *Cryptology ePrint Archive*.

[Bos et al. 2015] Bos, J. W., Costello, C., Naehrig, M., and Stebila, D. (2015). Post-quantum key exchange for the tls protocol from the ring learning with errors problem. In *2015 IEEE Symposium on Security and Privacy*, pages 553–570. IEEE.

[de Clercq et al. 2015] de Clercq, R., Roy, S. S., Vercauteren, F., and Verbauwhede, I. (2015). Efficient software implementation of ring-lwe encryption. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, DATE '15, page 339–344, San Jose, CA, USA. EDA Consortium.

[Fan and Vercauteren 2012] Fan, J. and Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption. *Cryptology ePrint Archive*.

[Lindner and Peikert 2011] Lindner, R. and Peikert, C. (2011). Better key sizes (and attacks) for lwe-based encryption. In Kiayias, A., editor, *Topics in Cryptology – CT-RSA 2011*, pages 319–339, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Lyubashevsky et al. 2013] Lyubashevsky, V., Peikert, C., and Regev, O. (2013). On ideal lattices and learning with errors over rings. *J. ACM*, 60(6).

[Peikert 2009] Peikert, C. (2009). Public-key cryptosystems from the worst-case shortest vector problem: Extended abstract. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, page 333–342, New York, NY, USA. Association for Computing Machinery.

[Regev 2009] Regev, O. (2009). On lattices, learning with errors, random linear codes, and cryptography. volume 56, New York, NY, USA. Association for Computing Machinery.

[Roy et al. 2016] Roy, S. S., Karmakar, A., and Verbauwhede, I. (2016). Ring-lwe: applications to cryptography and their efficient realization. In *International conference on security, privacy, and applied cryptography engineering*, pages 323–331. Springer.

[Wu et al. 2012] Wu, Y., Huang, Z., Zhang, J., and Wen, Q. (2012). A lattice-based digital signature from the ring-lwe. In *2012 3rd IEEE International Conference on Network Infrastructure and Digital Content*, pages 646–651.