Improved Parameters for the Ring-TESLA Digital Signature Scheme

Arjun Chopra*

Abstract

Akleylek *et al.* have proposed Ring-TESLA, a practical and efficient digital signature scheme based on the Ring Learning With Errors problem. However we have identified there are some problems with the *parameters* proposed for Ring-TESLA, as we believe they do not ensure the correct operation of the scheme and do not provide the targeted levels of security under either the provable Ring-TESLA reduction, or an assessment of practical modern attacks such as lattice sieving.

We recommend new Ring-TESLA parameters that target more security levels and provide for correct, secure, and efficient instantiation. We describe the necessary preliminaries, recap the Ring-TESLA scheme, and present our parameter recommendations, selection methodology, and analysis.

Keywords: Cryptography, Post-Quantum Cryptography, Lattice, Ring-LWE, Ring Learning With Errors, Digital Signature, Ring-TESLA

1 Introduction

Lattice-based cryptographic primitives are emerging as promising post-quantum alternatives for classical asymmetric public key cryptography. There are now mature proposals for efficient and secure Diffie Hellman-like key agreements such as [4, 6, 12, 13] which have strong security guarantees and have been shown to work within real-world protocols. However practical post-quantum alternatives to classical *digital signature* schemes are only now beginning to emerge, such as [2, 5, 7].

A strong candidate for a post-quantum digital signature scheme is the Ring-TESLA signature scheme by Akleylek *et al.* in [2], which is fast, compact, and provably secure with a tight security reduction to the Ring Learning With Errors (Ring-LWE) problem. Because it uses uniform samplying for signatures, Ring-TESLA is easier to implement than the BLISS proposal of Ducas *et al.* BLISS uses discrete Gaussian sampling for signatures and so requires careful

*`arjun.chopra.vsc@outlook.com`. `Associate Consultant, VS Communications`

precision analysis, such as in a paper by Saarinen in [11], to preserve the security. Ring-TESLA also avoids NTRU lattices, which may be vulnerable to emerging subfield lattice attacks described by Kirchner and Fouque in [8].

However we have identified there are problems with the *parameters* proposed in [2] to instantiate Ring-TESLA. We have observed empirical failure rates around one per-cent, where valid signatures are incorrectly rejected by the verification algorithm. We have observed the proposed parameters are also not fully secure as they violate constraints set out in [2] for the Ring-TESLA security reduction and because, in the case of the parameters targeting 128 bits of security, the practical security is insufficient against modern costing methodologies such in [6] by Ducas *et al.*

In this paper we shall recommend new parameters for use with Ring-TESLA to ensure that it is correct, secure, and efficient[1]. In Section 2 we shall give the necessary background, establishing cyclotomic rings, discrete Gaussian distributions and the Ring-LWE problem. In Section 3 we shall review the Ring-TESLA scheme, presenting the key generation, signing and signature verification stages, and comment on its correctness. In Section 4 we shall describe and justify our parameter recommendations. We suggest that our parameters are superior in terms of efficiency, security and correctness to those proposed in the original paper [2]. We shall conclude in Section 5.

We stress that although this paper highlights and resolves some problems with Ring-TESLA *parameter selection*, this should not be interpreted as a criticism of the overall scheme. On the contrary, when properly instantiated Ring-TESLA is an elegant, efficient, and secure digital signature scheme, and we hope that our contribution will support its use in practice.

2 Preliminaries

In this section we shall recap preliminaries necessary to understand the Ring-TESLA signature scheme.

2.1 Cyclotomic Rings

Let $R = \mathbb{Z}/\langle\Phi_m(x)\rangle = \mathbb{Z}[\zeta_m]$ be the mth cyclotomic ring, where $\Phi_m(x)$ is the m-th cyclotomic polynomial defined by $x^m - 1 = \prod_{d|m}\Phi_d(x)$ and ζ_m is a primitive m-th root of unity. The

[1] We note a very recent paper [3] by Barreto *et al.* describing independent work on TESLA♯, a new algorithm based on Ring-TESLA. There are several differences between our work and that in [3]. We address the Ring-TESLA correctness concerns by careful parameter selection while [3] addresses them by modifying the algorithm. We apply the more recent and conservative costing methodology in [6] to assess the security of our parameters whereas [3] uses the approach by Albrecht *et al.* in [1]. Finally, we consider cyclotomic rings of degree $p-1$ for prime p as well as the power of two case considered in [3]. In future work we intend to analyse parameter selection for TESLA♯.

degree n of R is the degree of Φ_m, which is given by the Euler totient function $\phi(m)$. In this paper m shall be either a power of two, or shall be prime.

In the case where m is a power of 2, the situation is $n = \phi(m) = m/2$ and $\Phi_m(x) = 1+x^n$. This case has the advantage of practical efficiencies and simplifications, but has the disadvantage of restricting to a narrow range of parameter sizes. A wider choice of parameter sizes is provided when instead m is prime, from which $n = \phi(m) = m - 1$, and $\Phi_m(x) = 1 + x + x^2 + \ldots + x^n$.

For any integer q we shall let R_q denote the quotient ring R/qR. Multiple bases for the rings R and R_q are considered in the literature and used for efficient implementations, however for ease of exposition we shall here consider only the natural basis for R and R_q given by $\{1, \zeta, \zeta^2, \ldots, \zeta^{n-1}\}$, also known as the power basis.

Elements of R_q are represented as $\sum_i x_i \cdot \zeta^i$ where the coefficients x_i are integers in $[0, q)$. For any integer d, we shall define the *rounding* function $\mathrm{round}_d : R_q \to R_q$ by

$$\mathrm{round}_d\left(\sum_i x_i \cdot \zeta^i\right) = \sum_i \left\lfloor \frac{x_i}{2^d} \right\rfloor \cdot \zeta^i.$$

We shall also consider coefficients modulo 2^d, which should be interpreted by considering the representation of the coefficient in $[0, q)$ and computing its value modulo 2^d in $[0, 2^d)$.

We shall also refer to coefficients that are negative mod q for consistency with the language in [2]. These shall be interpreted as the appropriate mod q representation in $[0, q)$, for example -3 as $4 \bmod q$ in the simple case where $q = 7$. We shall also refer to the absolute value of a coefficient, which shall mean the absolute value of its representation in $[(1 - q)/2, (q + 1)/2] \bmod q$.

The variation in notation may appear perhaps confusing, but clarifies possible ambiguities between the presentation in [2] and the accompanying software implementation; the paper [2] reports that mod q coefficients are represented in $[(1 - q)/2, (q + 1)/2]$ but the implementation makes clear that rounding and mod 2^d computations are always applied to the representation in $[0, q)$.

For any integer ω, we define S_ω the set of ω-*sparse* elements of R_q to be the $2^\omega\binom{n}{\omega}$ elements for which ω of the x_i are $\pm 1 \bmod q$, and the remainder are zero[2].

2.2 Discrete Gaussian Distributions

The discrete Gaussian distribution of standard deviation σ assigns to each $x \in \mathbb{Z}$ a probability proportional to $e^{-x^2/(2\sigma^2)}$, normalised by the factor $S = 1 + 2\sum_{k=1}^{\infty} e^{-k^2/(2\sigma^2)}$, given by

$$D_{\mathbb{Z},\sigma}(x) = \frac{1}{S} e^{-x^2/(2\sigma^2)}.$$

[2] The [2] presentation suggests non-zero coefficients are restricted to be 1 only. However, because of the enumeration given, we believe that non-zero coefficients are intended to be $\pm 1 \bmod q$.

The *Gaussian parameter* of the discrete Gaussian distribution $D_{\mathbb{Z},\sigma}$ is $\sqrt{2\pi}\sigma$.

2.3 The Ring-LWE Problem

We here recall the Ring-LWE distribution, and the associated decision and search problems:

Definition 1 (Ring-LWE Distribution)**.** For an $s \in R_q$ and a distribution χ over R, a sample from the Ring-LWE Distribution $A_{s,\chi}$ over $R_q \times R_q$ is generated by sampling a uniformly at random in R_q, sampling e from χ, and outputing $(a, as + e)$.

Definition 2 (Decision Ring-LWE)**.** The Decision Ring-LWE Problem is to distinguish with non-negligible advantage between independent samples from $A_{s,\chi}$ where s is chosen once and for all, and the same number of *uniformly random* and independent samples from $R_q \times R_q$.

Definition 3 (Search Ring-LWE)**.** The Search Ring-LWE Problem is to recover s with non-negligible advantage from samples from $A_{s,\chi}$ where s is chosen once and for all.

We shall consider only the case where χ is the co-ordinate-wise discrete Gaussian distribution $(D_{\mathbb{Z},\sigma})^n$. If properly instantiated, the decision and search Ring-LWE Problems are known to be hard so long as the Shortest Vector Problem in ideal lattices is hard. They benefit from a reduction from average to worst cases [9], namely that if *any* instance of the problem is hard then *typical* instances are hard as well.

For these hardness guarantees, the ring R_q must be instantiated so that $q \equiv 1 \mod m$. When as here χ is discrete Gaussian, Peikert in [10] explains that it must also be spherical *in the dual space* with Gaussian parameter greater than or equal to two. Practically, this means that the Gaussian parameter $\sigma\sqrt{2\pi}$ must be greater than or equal to two if m is a power of two, and greater than or equal to $2\sqrt{n}$ if m is prime.

The Ring-LWE problem is a special ideal-lattice case of the general Learning With Errors (LWE) problem defined over lattices:

Definition 4 (LWE Distribution)**.** For an $\mathbf{s} \in \mathbb{Z}_q^n$ and a distribution χ over $\mathbb{Z}$, a sample from the LWE Distribution $B_{\mathbf{s},\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is generated by sampling $\mathbf{b}$ uniformly at random in $\mathbb{Z}_q^n$, sampling e from χ, and outputing $(\mathbf{b}, \mathbf{b} \cdot \mathbf{s} + e \bmod q)$.

Definition 5 (Decision LWE)**.** The Decision LWE Problem is to distinguish with non-negligible advantage between independent samples from $B_{\mathbf{s},\chi}$ where $\mathbf{s}$ is chosen once and for all, and the same number of *uniformly random* and independent samples from $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Definition 6 (Search LWE)**.** The Search LWE Problem is to recover $\mathbf{s}$ with non-negligible advantage from samples from $B_{\mathbf{s},\chi}$ where $\mathbf{s}$ is chosen once and for all.

A single Ring-LWE sample $(a, as + e)$ corresponds to n LWE samples:

$$(\mathbf{b}_1, \mathbf{b}_1 \cdot \mathbf{s} + e_1 \bmod q), \ldots, (\mathbf{b}_n, \mathbf{b}_n \cdot \mathbf{s} + e_n \bmod q).$$

Each $\mathbf{b}_i$ is the vector of coefficients for the polynomial $\zeta^i a$, $\mathbf{s}$ is vector of coefficients from s, and the e_i are the coefficients of e. Therefore the bit-level hardness of Ring-LWE problems can be calculated from the bit-level hardness of LWE problems.

Concrete assessments of LWE security are given by Albrecht *et al.* in [1], and most recently and conservatively by Ducas *et al.* in [6], which analyses lattice sieving in conjunction with the BKZ 2.0 algorithm.

3 The Ring-TESLA Signature Scheme

In this section we shall describe the Ring-TESLA signature scheme as presented in [2] and analyse its correctness. The components of Ring-TESLA are Key Generation, Sign and Verify. In addition to the parameters m, n, σ, q, d in Section 2, there are integer parameters L, ω, κ, B, U, and auxiliary functions H and F. To benefit from the supporting security proof, these must satisfy constraints described in [2], which we shall now describe and place in context.

Ring-TESLA requires a κ-bit hash function $H : \{0,1\}^* \to \{0,1\}^\kappa$ and an *encoding* function $F : \{0,1\}^\kappa \to S_\omega$ from the output of H to the sparse elements of R_q. This can be instantiated, for example, using a pseudo-random number generator (PRNG) such as AES in counter mode, seeded with the hash output, and setting coefficients from the output.

The output size κ of the hash function H must have at least the required security level λ of the signature scheme, for instance κ could be 256 for up to 128 bits of security and H instantiated as SHA256. It must be chosen so that the output space of H is larger than the number of ω-sparse elements of R_q. The encoding function F must be chosen so that the probability of mapping two hash outputs to the same sparse element is less than one in 2^λ. This will be satisfied if both the output space of the PRNG and the number of ω-sparse elements of R_q are larger than 2^λ.

The parameter L is used in checks during both the Key Generation and Sign stages. The parameters B and U are used in uniform sampling and rejection sampling during the Sign process. For secure sampling that does not leak private information through signatures, the parameter B must be less than 2^d, and greater than or equal to $14(n-1)\sqrt{\omega}$, and the parameter U must be $\lceil 14\sqrt{\omega}\sigma \rfloor$. The number 14 chosen here by the authors of [2] corresponds to a 'large number' of Gaussian standard deviations, and could plausibly be tuned to the exact security level required although we do not do so here. The parameters must be picked so that q will be greater than or equal to $4B$ and $2^{2d+1+\kappa/n}/B$.

As per Section 2, we shall require $q \equiv 1 \bmod m$, and shall require the Gaussian parameter $\sigma\sqrt{2\pi}$ to be greater than or equal to two if m is a power of two, or greater than or equal to $2\sqrt{n}$ if m is prime.

We now describe the operation of Ring-TESLA.

```
Key Generation
Input: Uniformly-sampled public parameters[3] $a_1, a_2 \in R_q$.
Output: Private/public key pair (sk,pk).

1. Sample $s, e_1, e_2 \in R_q$ with coefficients from $D_{\mathbb{Z},\sigma}$ mod $q$.
2. If checkE($e_1$) or checkE($e_2$) returns Reject then restart.
3. $t_1 \leftarrow a_1 s + e_1$
4. $t_2 \leftarrow a_2 s + e_2$
5. sk $\leftarrow (s, e_1, e_2)$
6. pk $\leftarrow (t_1, t_2)$
7. Return (sk,pk)
```

```
CheckE
Input: $e \in R_q$.
Output: Accept or Reject.

1. If the $\omega$ largest absolute values of coefficients from $e$ sum to more than $L$ then return Reject.
2. Return Accept.
```

For efficient key generation the parameters L, ω, σ must be picked so that e_1, e_2 are accepted by CheckE with high probability. In the original Ring-TESLA paper [2] this is at least 0.5; in our Section 4 parameter recommendations we target at least 0.8 and in most cases achieve 0.99.

[3] These may be globally specified or chosen by each user.

Sign

Input: Message m, private key (s, e_1, e_2), public parameters a_1, a_2.

Output: Signature (z, c').

```
1. Sample $y \in R_q$ with coefficients uniform in $[-B, B] \bmod q$.
2. $v_1 \leftarrow a_1 y$
3. $v_2 \leftarrow a_2 y$
4. $c' \leftarrow H\left(\mathrm{round}_d(v_1), \mathrm{round}_d(v_2), m\right)$
5. $c \leftarrow F(c')$
6. $z \leftarrow y + sc$
7. $w_1 \leftarrow v_1 - e_1 c$
8. $w_2 \leftarrow v_2 - e_2 c$
9. If any coefficient $w_1$ or $w_2$ is within $L$ of a multiple of $2^d$ then restart.
10. If any coefficient of $z$ is not in $[-(B-U), B-U] \bmod q$ then restart.
11. Return $(z, c')$
```

Restarting in steps 9 or 10 will reduce performance by extending the signing process. The independent per-coefficient probability of restarting is $L/2^{d-1}$ at step 9 and $2U/(2B+1)$ at step 10, so the overall probability of passing through these steps without restarting is $\left(1 - \frac{L}{2^{d-1}}\right)^{2n} \left(\frac{2(B-U)+1}{2B+1}\right)^{n}$.

Verify

Input: Message m, public key (t_1, t_2), public parameters a_1, a_2, signature (z, c').
Output: Accept or Reject.

```
1. If any coefficient of $z$ is not in $[-(B-U), B-U] \bmod q$
   then return Reject.
2. $c \leftarrow F(c')$
3. $w_1 \leftarrow a_1 z - t_1 c$
4. $w_2 \leftarrow a_2 z - t_2 c$
5. $c'' \leftarrow H\left(\text{round}_d(w_1), \text{round}_d(w_2), m\right)$
6. If $c'' \leftarrow c'$ then return Accept, otherwise return Reject.
```

We found it difficult to implement the parameter sets provided in the original Ring-TESLA paper [2] because despite a valid key pair, Verify would sometimes reject signatures produced by Sign. We observed this empirically with approximately one per-cent of signatures using the 'Ring-TESLA-2' parameters from [2]. This is detrimental to the smooth operation of the scheme, and could allow an active attacker to recover private key information by causing and analysing rejections, although we stress that we do not know of any such practical attack. We now explain the reason for these failures, and how we overcame it in our own parameter selection.

For Verify to accept a signature, $\text{round}_d(v_i)$ must equal $\text{round}_d(w_i)$ for $i = 1, 2$. These are related[4] by $w_i = v_i - e_i c$. Step 9 of Verify ensures the coefficients of w_1, w_2 are not close to multiples of 2^d. CheckE from Key Generation ensures the coefficients of $e_i c$ have small absolute value. Therefore the outputs of round_d are not affected by $e_i c$ when *integer* arithmetic is considered.

Addition of coefficients is however *modulo* q so round_d may still give different outputs, which we observed using the parameters in [2]. To prevent this, we shall impose an additional requirement beyond [2] and require that $q \bmod 2^d$ is no greater than L, which we have found ensures correct signature validation. For the mathematical reader we shall now give a proof.

Theorem 1. Provided that $q \bmod 2^d$ is in $[0, L]$, Verify will return Accept when using a valid key pair.

Proof. Verify will Accept if $\text{round}_d(v_i)$ equals $\text{round}_d(w_i)$ for $i = 1, 2$. They are related by $w_i = x_i + e_i c$.

[4] This equation is equivalent to the alternate equation $w_i = a_i z - t_1 c$ used in Verify.

Coefficients of $e_i c$ are in $[-L, L]$ mod q because they are the sum of no more than ω coefficients from e_i. The sum of any ω absolute values of e_i is no more than L as CheckE accepted e_i in Key Generation.

Let x_{ij} be the coefficients of v_i and y_{ij} be the coefficients of w_i, so that $y_{ij} = x_{ij} + r_{ij} \bmod q$ with r_{ij} in $[-L, L]$. Because w_i was accepted by step 9 of Sign, y_{ij} is at least L away from a multiple of 2^d. Therefore by the Lemma below, y_{ij} is in $[L, q-L)$. As r_{ij} is in $[-L, L]$, it follows that $y_{ij} = x_{ij} + r_{ij}$ holds as an *integer* equation as well as modulo q.

Because y_{ij} is at least L from a multiple of 2^d and because r_{ij} is in $[-L, L]$, it follows that every $\left\lfloor \frac{x_{ij}}{2^d} \right\rfloor$ equals $\left\lfloor \frac{y_{ij}}{2^d} \right\rfloor$, so $\mathrm{round}_d(v_i)$ equals $\mathrm{round}_d(w_i)$.

□

Lemma 1. If y is an integer at least L from any multiple of 2^d, and that lies in $[0, q)$ for an integer q such that $q \bmod 2^d < L$, then it must follow that y is in $[L, q-L)$.

Proof. y is at least L away from any multiple of 2^d. As 0 is such a multiple, therefore y is greater than or equal to L.

$q = k2^d + s$ for integer k and integer s in $[0, L)$. As y is at least L away from any multiple of 2^d, it follows that either y is greater than or equal to $k2^d + L$, or that y is less than or equal to $k2^d - L$. It cannot be the first, because $k2^d + L$ equals $q + L - s$, which must be greater than q, therefore it must be the second. Because $k2^d - L$ is equal to $q - s - L$ and s is non-negative, y is less than $q - L$.

□

4 Parameters

In this section we recommend several parameter sets for Ring-TESLA, targeting a variety of security levels and performance characteristics.

4.1 Methodology

We select our parameters to satisfy the constraints given in [2] and documented in Section 2 and Section 3. However we use different methodology from [2], which we have found gives more efficient parameters, for example signatures that are 0.5kB smaller than we would have found when targeting 128 bits of security.

We now describe our methodology. For a target security level λ and choice of m we take $n = \phi(m)$ and $\kappa = 2\lambda$. We pick $\sigma_{\min}$ so that $\sigma_{\min}\sqrt{2\pi}$ is at least 2 if m is a power of two and at least $2\sqrt{n}$ if m is prime. We choose ω so that the number $2^{\omega}\binom{n}{\omega}$ of ω-sparse polynomials is greater than 2^{λ} and less that 2^{κ}.

For each σ increasing from $\sigma_{\min}$ we set $U = \lceil 14\sigma\sqrt{\omega}\rceil$ and $d_{\min} = \lceil \log_2 (14(n-1)\sigma\sqrt{\omega})\rceil$, and loop through a range of $b, d \geq d_{\min}$ with $b \leq d$. We set B_{temp} to be $2^b - 1$, and try to pick L maximal subject to obtaining desired Key Generation and Sign acceptance rates if we were to choose B to equal B_{temp}. For any (b, d) where we succeed, we pick the least prime q that is greater than both $4B_{\text{temp}}$ and $2^{2d+1+\kappa/n}/B_{\text{temp}}$, and which satisfies $q \bmod 2^d$ no greater than L. We pick finally $B \leq B_{\text{temp}}$ smallest subject to a desired Sign acceptance rate and subject to exceeding $14(n-1)\sigma\sqrt{\omega}$ and $q/2^{2d+1+\kappa/n}$.

We output $(m, n, \sigma, L, \omega, B, U, d, q, \kappa)$ as a candidate parameter set and assess its security. The various parameter constraints described in Section 3 are satisfied, therefore the security proof in [2] means that forging a Ring-TESLA signature is at least as hard as solving the Decision Ring-LWE problem for R_q with distribution $D_{\mathbb{Z},\sigma}$. We assess the security of the Ring-LWE problem afforded by the parameters n, q, σ, and retain the candidate parameter set if this exceeds the target level.

In practice the Decision Ring-LWE problem is solved by solving the Search Ring-LWE problem, hence our security assessments are based on the bit-level hardness of this problem, using the conservative methodology proposed in [6]. From the secure candidate parameter sets, we select those with desirable characteristics such as small signature or key sizes or good Sign and Verify performance.

4.2 Recommendations

We provide three sets of parameters 'A', 'B' and 'C' targeting respectively 80, 128, and 192 bits of security against classical attacks. The 'A' parameters do not target a particular level of security against a quantum attack, and the 'B' and 'C' sets target respectively 80 and 128 bits of post-quantum security.

In each collection, a '1' variant is optimised for computational efficiency, a '3' variant is optimised for compact signatures, and a '2' variant targets a balance. For this reason, the '1' variants use m that are powers of two, because the Fast Fourier Transform used for multiplication in R_q is more efficient in this case, and the '2' and '3' variants use prime m for greater flexibility. We include the parameter sets 'Ring-TESLA-1' and 'Ring-TESLA-2' proposed in [2] for comparison.

Our recommendations are:

	m	n	σ	L	ω	B	U	d	q	κ
A1	1024	512	59	3046	11	4192013	2741	23	50332673	160
A2	479	478	54	1733	11	4189552	2508	22	20972537	160
A3	449	448	34	948	11	1048088	1580	20	5243423	160
B1	2048	1024	2	109	16	509644	113	19	5767169	256
B2	827	826	23	1035	17	4190594	1329	22	33554699	256
B3	709	708	28	1297	17	2094584	1617	21	10486111	256
C1	2048	1024	162	14780	26	67103941	11566	26	335546369	384
C2	983	982	220	14482	26	67102911	15706	26	335549017	384
C3	929	928	212	16998	26	33550234	15135	25	167779258	384
Ring-TESLA-1	1024	512	30	814	11	2097151	993	21	8399873	160
Ring-TESLA-2	1024	512	52	2766	19	4194303	3173	23	39960577	256

We suggest that our recommendations have security superior to the proposals in [2]. They satisfy all of the conditions specified in [2] for the Ring-TESLA security proof to hold, achieve the target security bit levels, and satisfy the constraints described at the end of Section 3 to ensure correct validation of signatures, and so are secure against an active attacker who might cause and analyse validation failures.

	Security Target		Security Assessment		Constraints
	Classical	Quantum	Classical	Quantum	Satisfied?
A1	80	-	80	73	Yes
A2	80	-	80	72	Yes
A3	80	-	80	72	Yes
B1	128	80	135	123	Yes
B2	128	80	132	120	Yes
B3	128	80	128	116	Yes
C1	192	128	192	174	Yes
C2	192	128	192	174	Yes
C3	192	128	192	174	Yes
Ring-TESLA-1	80	-	87	79	No because $U < 14\sqrt{\omega}\sigma$.
Ring-TESLA-2	128	-	80	73	No because $q < \frac{2^{2d+1+\frac{\kappa}{n}}}{B}$

Our recommendations also achieve competetive signature and key sizes:

	Size (kB)			Accept. Prob.	
	Signature	Public Key	Private Key	Key. Gen.	Sign
A1	1.4	3.3	1.9	0.99	0.34
A2	1.4	2.9	1.8	0.99	0.34
A3	1.2	2.5	1.7	0.82	0.10
B1	2.5	6.5	1.8	0.99	0.34
B2	2.3	5.2	2.0	0.96	0.34
B3	1.9	4.5	2.3	0.99	0.10
C1	3.5	7.3	4.3	0.99	0.34
C2	3.3	7.0	4.3	0.99	0.34
C3	3.1	6.4	4.0	0.99	0.10
Ring-TESLA-1	1.4	3.1	1.7	0.50	0.35
Ring-TESLA-2	1.5	3.3	1.8	0.99	0.34

5 Conclusion

Lattice-based cryptography and the Ring-LWE problem are promising post-quantum alternatives to classical public key cryptography. The Ring-TESLA algorithm is a competitive digital signature scheme based on Ring-LWE. We have examined Ring-TESLA and explained how careful parameter choices are necessary for correct and secure operation of the scheme. We have recommended new practical Ring-TESLA parameter choices which we suggest improve upon previous proposals because they ensure its correct, secure and efficient instantiation. In future work we intend to further explore enhancements to Ring-TESLA and other digital signature algorithms.

Acknowledgement We are most grateful to our colleague Vikram Singh for his guidance and support in this research.

References

[1] M.R. Albrecht, R. Player, and S. Scott. On the concrete hardness of Learning with Errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.

[2] S. Akleylek, N. Bindel, J. Buchmann, J. Krämer, and G. A. Marson. An Efficient Lattice-Based Signature Scheme with Provably Secure Instantiation. http://eprint.iacr.org/2016/030.

[3] P. S. L. M. Barreto, P. Longa, M. Naehring, J. E. Ricardini, and G. Zanon. Sharper Ring-LWE Signatures. http://eprint.iacr.org/2016/1026.

[4] J. W. Bos, C. Costello, M. Naehrig, and D. Stebila. Post-quantum key exchange for the TLS protocol from the ring learning with errors problem. http://eprint.iacr.org/2014/599.

[5] L. Ducas, A. Durmas, T. Lepoint, and V. Lyubashevsky. Lattice signatures and Bimodal Gaussians. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013*, Part I, volume 8042 of LNCS, pages 40–56, Santa Barbara, CA, USA, Aug. 18–22, 2013. Springer, Heidelberg, Germany.

[6] L. Ducas, T Pöppelmann and P. Schwabe. Post-quantum key exchange — a new hope. http://eprint.iacr.org/1092/2015.

[7] T. Guneysu, V. Lyubashevsky, and T. Pöppelmann. Practical Lattice-Based Cryptography: A Signature Scheme for Embedded Systems. In E. Prouff and P. Schaumont, editors, *Cryptographic Hardware and Embedded Systems CHES 2012*, volume 7428 of Lecture Notes in Computer Science, pages 530–547. Springer, 2012.

[8] P. Kirchner and P.-A. Fouque, Comparison between Subfield and Straightforward Attacks on NTRU. http://eprint.iacr.org/2016/717.

[9] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In H. Gilbert, editor *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23, French Riviera, May 30 – June 3, 2010, Springer, Heidelberg, Germany.

[10] C. Peikert. How (Not) to Instantiate Ring-LWE. http://eprint.iacr.org/2016/351.

[11] M. J. Saarinen. Gaussian Sampling Precision in Lattice Cryptography. http://eprint.iacr.org/2015/953.

[12] V. Singh. A Practical Key Exchange for the Internet using Lattice Cryptography. http://eprint.iacr.org/2015/138.

[13] V. Singh and A. Chopra. Even More Practical Key Exchanges for the Internet Using Lattice Cryptography http://eprint.iacr.org/2015/1120.