

1. OKCN-KEX (IND-CPA) 算法的推荐参数为 $m = 2, q = 7681, g = 4$. 这样根据 KC 参数要求的关系 $(2d + 1)m < q(1 - \frac{1}{g})$, 可以计算得 $d < 1440$ 即可保证 KC 算法的正确性. 说明文档中给出的 OKCN-KEX 协议如下 (其中 Algorithm 3 中应为 $E_\sigma \leftarrow S_\eta$):

Algorithm 1 (pk_1, sk_1) \leftarrow KeyGen-Alice(seed)	Algorithm 2 (pk_2, sk_2) \leftarrow KeyGen-Bob(seed)
1: $A := \text{Gen}(\text{seed})$	1: $A := \text{Gen}(\text{seed})$
2: $X_1, E_1 \leftarrow S_\eta^{t \times 1}$	2: $X_2, E_2 \leftarrow S_\eta^{t \times 1}$
3: $Y_1 := \lfloor (AX_1 + E_1) / 2^{t_1} \rfloor$	3: $Y_2 := \lfloor (A^T X_2 + E_2) / 2^{t_1} \rfloor$
4: return ($pk_1 := Y_1, sk_1 := X_1$)	4: return ($pk_2 := Y_2, sk_2 := X_2$)

Algorithm 3 (ss_2, signal) \leftarrow KDF-Bob(pk_1, sk_2)	Algorithm 4 $ss_1 \leftarrow$ KDF-Alice($pk_2, sk_1, \text{signal}$)
1: $E_\sigma \leftarrow S_\eta^{t \times 1}$	1: $\Sigma_1 := X_1^T (2^{t_2} Y_2)$
2: $\Sigma_2 := 2^{t_1} Y_1^T X_2 + E_\sigma$	2: $K_1 := \text{Rec}(\Sigma_1, V, \text{params})$
3: (K_2, V) \leftarrow Con(Σ_2, params)	3: return $ss_1 := K_1$
4: return ($ss_2 := K_2, \text{signal} := V$)	

算法使用的是分圆环 $Z[x] / x^{256} + 1$, 模的秩为 $l = 3$, MLWE 对应的误差分布为中心二项分布 S_2 . 选取的参数 $t = t_1 = t_2 = 3$. 注意到 OKCN-KEX 输入 KC 的 Σ_1 和 Σ_2 之间的误差可以用说明文档中的方法作如下分析: 记 $\varepsilon_1 = AX_1 + E_1 - 2^t \lfloor (AX_1 + E_1) / 2^t \rfloor$, $\varepsilon_2 = A^T X_2 + E_2 - 2^t \lfloor (A^T X_2 + E_2) / 2^t \rfloor$, 则有

$$\begin{aligned}
\Sigma_1 - \Sigma_2 &= X_1^T (2^t Y_2) - (2^t Y_1^T X_2 + E_\sigma) \\
&= 2^t X_1^T \lfloor (A^T X_2 + E_2) / 2^t \rfloor - ((2^t \lfloor (AX_1 + E_1) / 2^t \rfloor)^T X_2 + E_\sigma) \\
&= X_1^T (A^T X_2 + E_2 - \varepsilon_2) - ((AX_1 + E_1 - \varepsilon_1)^T X_2 + E_\sigma) \\
&= X_1^T (E_2 - \varepsilon_2) - (E_1 - \varepsilon_1)^T X_2 - E_\sigma
\end{aligned}$$

注意到 $X_1, X_2, E_1, E_2 \leftarrow S_2^3$, $E_\sigma \leftarrow S_2$, 且 $\varepsilon_1, \varepsilon_2$ 的无穷范数不超过 4. 所以, 我们可以计算 $\Sigma_1 - \Sigma_2$ 的无穷范数的绝对上界, 即 $\|\Sigma_1 - \Sigma_2\|_\infty \leq \|X_1^T \cdot E_2\|_\infty + \|X_1^T \cdot \varepsilon_2\|_\infty + \|E_1^T \cdot X_2\|_\infty + \|\varepsilon_1^T \cdot X_2\|_\infty + \|E_\sigma\|_\infty < 18436$. 注意到这是一个平凡的理论上界. 本竞选算法所选取的参数并不满足这个绝对上界, 因而 OKCN-KEX 会有一定的概率出错 (竞选算法的错误率分析一节, 采用的是理想情况下的分析, 即在 Nor-primal-D-MLWE(S_η) 假设情况下, 我们可以近似地将 $\varepsilon_1, \varepsilon_2$ 的系数分布看成为区间 $[-2^{t-1}, 2^{t-1}]$ 上的均匀分布 (更确切的说应为 $[-2^{t-1}, 2^{t-1})$ 上的均匀分布), 此时, $\Sigma_1 - \Sigma_2$ 的系数服从的分布的期望为 0,

方差为 $2 \cdot n \cdot l \cdot \left(\frac{\eta}{2}\right)^2 + 2 \cdot n \cdot l \cdot \frac{\eta}{2} \cdot \frac{1}{3} \cdot 2^{t-1} \cdot (2^{t-1} + 1) + \frac{\eta}{2} = 11777$). 这里, 我们可以使用类似于 Kyber 的方法, 使用脚本程序计算错误率的界 [BDK+17], 即计算 $\Pr[||\Sigma_1 - \Sigma_2||_\infty > 1440]$ 的上界. 我们采用 Kyber 开源的 python 脚本, 略作修改测试 OKCN-KEX, 得到的错误率结果约为 $2^{-142.4}$, 与文档声称的 $2^{-166.4}$ 的错误率大小不符, 但是仍比要求的 2^{-128} 要小得多. 我们的具体测试结果如下图:

```

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:099359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\Administrator\Desktop\Kyber\kyber---security-estimates-master\security-estimates-master\OKCN_Test.py
OKCN_MLWE_768_CCA:
-----
security:
Primal attacks uses block-size 560 and 685 samples
Primal & 685 & 560 & 163 & 148 & 116
Dual attacks uses block-size 560 and 610 samples
shortest vector used has length 1=10950.78, q=7681, \|kq\| = 0
log2(epsilon) = -57.88, log2 nvector per run 116.21
Dual & 610 & 560 & 163 & 148 & 116
params: {'n': 256, 'l': 3, 'ks': 2, 'ke': 2, 'q': 7681, 't': 3}
com costs: (992.0, 1024.0, 2016.0)
failure: 0.0 = 2^-142.4
>>>

```

2. 在认证密钥交换 (AKE) 的量子 random oracle (QROM) 方面, 算法说明文档没有给出详细的安全性证明, 目前现有的基于 QROM 模型下的安全性证明归约损失较大, 对参数要求较高. 但这也是目前大多数 AKE 算法均面临的问题. 为了解决此类问题, [BDK+17] 提出了一种新的归约思路, 给出了一种基于非标准假设的紧的归约, 然而假设的合理性仍需继续讨论. ([BDK+17] J.W. Bos et al.: CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM. ePrint 2017/634.)