DiSK: A Deniable Split KEM from The MLWE Problem

Brian Goncalves, Atefeh Mashatan
Cybersecurity Research Lab, Ted Rogers School of Information Technology Management, Toronto Metropolitan University

Introduction / Motivation

Shor's Algorithm [1] represents an existential threat to any cryptosystem built from quantum weak assumptions such as RSA or ECC. Any digital communication that relies on these traditional cryptosystems risks one day becoming completely insecure should a sufficiently powerful quantum computer come into existence. As a result, there has been a global effort in developing Quantum Computing-Resistant (QCR) cryptographic algorithms to replace these now vulnerable schemes.

One of the **most important categories of cryptographic protocols that need to be updated are key exchange protocols**. These protocols are a cornerstone of asynchronous, end-to-end encrypted messaging apps. One of the most **famous key exchange protocols** used by a variety of messaging apps, such as Signal, Whatsapp, RCS, Messenger, **is the eXtended Triple Diffie-Hellman (X3DH)** protocol [2]. Traditional X3DH is **based on a problem known to be weak to Shor's algorithm.** As **society will eventually need to migrate completely away from X3DH**, **finding a quantum computing-resistant replacement for it and updating the countless protocols that are built from it,** in order to continue protecting digital communications and **prevent further exposure to harvest-now-and-decrypt-later attacks is vital.**

Split KEMs and Deniability [3, 4]

In order to address the imminent need to migrate away from (X3)DH-based protocols, Brendel *et al.* **defined a type of Key Encapsulation Mechanism (KEM) that mirrored the message flow of X3DH,** which they called a **split KEM** [3]. Split KEMs function by encapsulation using Alice's secret key and Bob's public key, while decapsulation requires Bob's private key and Alice's public key. **Part of the motivation for this** new primitive was **the fact that attempting to replicate X3DH with standard KEMs required losing the asynchronicity** that the X3DH protocol enables. Along with the definition of split KEMs Brendel *et al.* defined a notion of indistinguishability from random.

The **initial attempt to replace X3DH was incomplete**, as the **definition of split KEM does not include any notion of deniability. Deniability** allows a party to claim that they, plausibly, did not initiate contact with someone, **a useful property to protect whistleblowers**. **Collins *et al.* completed the vision of split KEMs** as a replacement for DH **by formalizing what it means for a split KEM to be deniable [4].**

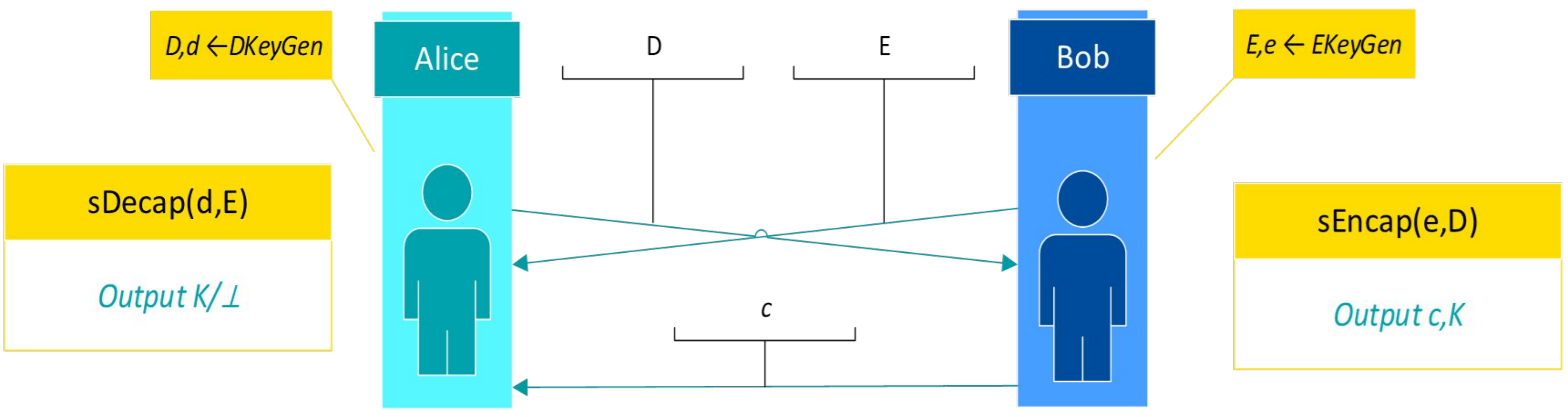

Figure 1: Structure of a Split KEM.

Module Learning With Errors [5, 6]

The **Module Learning With Errors (MLWE) problem**, first defined by Brakerski et al.[5] and then thoroughly studied by Langlois and Stehlé [6] is a generalization of the standard Learning With Errors problem, which is **known to be hard for quantum computers to solve.** The (decisional) MLWE problem is as follows:

Let $\boldsymbol{Z}$ be the set of integers, q a prime number, δ a power of 2, and χ be an error distribution.

- Let $\boldsymbol{Z}_q$ be the integer reduced mod q ie the set $\{0,1,\ldots,q-1\}$.
- Let $\boldsymbol{Z}_q[X]$ be the set of polynomials in X with coefficients from $\boldsymbol{Z}_q$.
- Let $R_q = \boldsymbol{Z}_q[X]/(X^{\delta}+1)$ be the set of polynomials from $\boldsymbol{Z}_q[X]$ reduced by X^d+1.
- Let $R_q^{n\times m}$ be the set of a matrices with entries from R_q.

A taken randomly from $R_q^{n\times m}$, **s, e** sampled from χ^m,χ^n, and **u** picked randomly from R_q^n.

Decide if you are given (A,As+e) **or** (A,**u**).

The **MLWE problem was proposed as a middle ground between traditional LWE and Ring LWE,** overcoming the failures of both. **Module lattices have a more complicated algebraic structures than ideal lattices,** but are **more structured than Euclidean lattices**. Consequently, **MLWE might be able to offer more security than Ring LWE and better performance than traditional LWE.**

NIKE SWOOSH [7]

Our construction is inspired by Gajland et al.'s Non-Interactive Key Exchange (NIKE) (passive) **SWOOSH**, itself based on MLWE [7]. The **basic idea behind SWOOSH is** that for a fixed, shared matrix A and error terms $\mathbf{s}_A, \mathbf{e}_A$ and $\mathbf{s}_B, \mathbf{e}_B$ **the following holds**:

$$(\mathbf{s}_0^T A + \mathbf{e}_0^T)\mathbf{s}_1 \simeq \mathbf{s}_0^T(A\mathbf{s}_1 + \mathbf{e}_1).$$

That is, the **terms above are approximately equal, with the disagreement due to the error terms**. This **difference**, however, **can be overcome through the use of a reconciliation function to round out the errors**. In addition to **adding a random shift, r, ensures a uniform distribution,** and that the **reconciliation process results in a random key**. In practice, **r** is the hash over the two parties IDs and public key.

SWOOSH Keys

- $sk_1 = (\mathbf{s}_{1,L}{}^T, \mathbf{s}_{1,R})$
- $pk_1 = (pk_{1,L} = \mathbf{s}_L^T A + \mathbf{e}_L^T,\ pk_{1,R} = A\mathbf{s}_R + \mathbf{e}_R)$

SWOOSH Key Derivation

- $\mathbf{r} = \mathbf{H}(ID_1, pk_1, ID_2, pk_2)$
- $K = \mathbf{Rec}(pk_{1,L}\mathbf{s}_{2,R} + \mathbf{r})$

DiSK - Idea

We call our split KEM **DiSK (Deniable Split KEM)**. By their very creation, **split KEMs are designed to mimic key exchange protocols.** More importantly, **in the nn-IND-CCA-security experiment, split KEMs behave even more like a passive NIKE** as the adversary does not have any oracle powers to engage with the challenge secret keys. **One potential issue with naively using SWOOSH directly as a split KEM is the fact that the keys are computed from fixed public keys**, and thus **all keys would be** deterministically computed and **constant between any two fixed parties**. Thus, **to overcome this issue in DiSK, sEncap is designed to run an ephemeral instance of SWOOSH to randomize the ciphertext and key** even when evoked on the same two parties, along with a SWOOSH instance on the public key of the decapsulators.
The random oracle is used to add a random shift to ensure that the intermediate keys k_0 are, in fact, uniformly distributed in R_q, and results in unconditional correctness in any ring.

DiSK - Construction

```
𝕯.EKeyGen(1^λ):
    1.  $A_E \leftarrow\$ GL_N(R_q)$
    2.  $\mathbf{s}_E \leftarrow\$ \chi^N$
    3.  $\mathbf{e}_E \leftarrow\$ \chi^N$
    4.  $E = (A_E, \mathbf{s}_E^T A_E + \mathbf{e}_E^T)$
    5.  $e = \mathbf{s}_E$

𝕯.sEncaps(D, e):
    6.  $\mathbf{s}_c \leftarrow\$ \chi^N$
    7.  $\mathbf{e}_0 \leftarrow\$ \chi^N$
    8.  $\mathbf{e}_1 \leftarrow\$ \chi^N$
    9.  Parse $D = (D_0, D_1)$
    10. $\mathbf{c}_0 = \mathbf{s}_c^T D_0 + \mathbf{e}_0^T$
    11. $\mathbf{c}_1 = e^T D_0 + \mathbf{e}_1^T$
    12. $k_0 = \mathbf{s}_c^T D_1 + \mathbf{H}(E, D, \mathbf{c}_0, \mathbf{c}_1)$
    13. $k_1 = e^T D_1 + \mathbf{H}(E, D, \mathbf{c}_0, \mathbf{c}_1)$
    14. $K = \mathbf{Rec}(k_0) \oplus \mathbf{Rec}(k_1)$
    15. Output $(c = (\mathbf{c}_0, \mathbf{c}_1), K)$

𝕯.DKeyGen(1^λ):
    1.  $A_D \leftarrow\$ GL_N(R_q)$
    2.  $\mathbf{s}_D \leftarrow\$ \chi^N$
    3.  $\mathbf{e}_D \leftarrow\$ \chi^N$
    4.  $D = (A_D, A_D\mathbf{s}_D + \mathbf{e}_D)$
    5.  $d = \mathbf{s}_D$

𝕯.sDecaps(E, d):
    6.  Parse $c = (\mathbf{c}_0, \mathbf{c}_1)$
    7.  $k^*_0 = \mathbf{c}_0 d + \mathbf{H}(E, D\ \mathbf{c}_0, \mathbf{c}_1)$
    8.  $k^*_1 = \mathbf{c}_1 d + \mathbf{H}(E, D, \mathbf{c}_0, \mathbf{c}_1)$
    9.  $K^* = \mathbf{Rec}(k^*_0) \oplus \mathbf{Rec}(k^*_1)$
    10. Output $K^*$
```

Figure 2: DiSK a Deniable nn-IND-CCA-secure split KEM
Public parameters:
$q, \delta, N, R_q = Z_q[X]/(X^{\delta}+1)$,
and
$H{:}(R_q^{N\times N}\times R_q)\times(R_q^{N\times N}\times R_q)\times R_q\times R_q \rightarrow R_q$

Security Theorems

Assuming the **quantum hardness of the MLWE$_{N, N, q, \delta, \chi}$ problem** and **that B is a bound on the maximum absolute value of the support of χ,** then for any **QPT adversary A, making an arbitrary number of queries to the random oracle H (that may be in superposition)**, then the split KEM **DiSK**, as defined above, is a ***nn-IND-CCA-secure*** split KEM that is $16B^2\delta^2/q$-correct and ***Deniable.***

More precisely, for any efficient QPT adversary A, against the ***nn-IND-CCA-security*** and ***Deniability*** of the split-KEM **DiSK**, of arbitrary query depth and total query term, there exists efficient QPT adversaries B_1, B_2, B_3, B_4, and B_5 such that

$$\mathbf{Adv}^{nn\text{-IND-CCA}}(\lambda,\mathfrak{D},A) \leq \mathbf{Adv}^{\mathrm{MLWE}}(N,N,q,\delta,\chi,B_1) + 8B\delta/q + \mathbf{Adv}^{\mathrm{MLWE}}(2N,N,q,\delta,\chi,B_2) + \mathbf{Adv}^{\mathrm{MLWE}}(N,N,q,\delta,\chi,B_3).$$

$$\mathbf{Adv}^{\mathrm{DENY}}(\lambda,\mathfrak{D},A) \leq 4B\delta/q + \mathbf{Adv}^{\mathrm{MLWE}}(2N,N,q,\delta,\chi,B_4) + \mathbf{Adv}^{\mathrm{MLWE}}(N,N,q,\delta,\chi,B_5).$$

Future Work

The **immediate direction of future research** is to investigate how to **construct a Deniable Authenticated Key Exchange (DAKE) from DiSK** and its performance against previous works studying DAKEs.
Furthermore, **another avenue of research is to determine if DiSK possesses the UNF-1KCA and IND-1BatchCCA security notions** for split KEMs as defined by Collins *et al.* [4]. Importantly, **Collins *et al.* required the use of a QROM transform applied to their split KEM to obtain them and demonstrate the security of their DAKE, K-Waay[4].** As with the major of KEM transforms, this incurs a quadratic loss in the QROM model, thus proving that DiSK, or a variant, has such properties, which means that it can be used directly in K-Waay without such losses.

References

1. P.W. Shor. 1994. Algorithms for quantum computation: discrete logarithms and factoring. In Proceedings 35th Annual Symposium on Foundations of Computer Science, 124–134.
2. Moxie Marlinspike and Trevor Perrin. 2016. The x3dh key agreement protocol. Open Whisper Systems 283 (2016), 10.
3. Jacqueline Brendel, Marc Fischlin, Felix Günther, Christian Janson, and Douglas Stebila. 2020. Towards post-quantum security for Signal's X3DH handshake. In Proc. 27th Conference on Selected Areas in Cryptography (SAC) 2020 (LNCS), Michael J. Jacobson Jr., Orr Dunkelman, and Colin O'Flynn (Eds.). Springer.
4. Daniel Collins, Loïs Huguenin-Dumittan, Ngoc Khanh Nguyen, Nicolas Rolin, and Serge Vaudenay. 2025. K-waay: fast and deniable post-quantum X3DH without ring signatures. In Proceedings of the 33rd USENIX Conference on Security Symposium (Philadelphia, PA, USA) (SEC '24). USENIX Association, USA, Article 25, 18 pages.
5. Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. 2012. (Leveled) fully homomorphic encryption without bootstrapping. In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (Cambridge, Massachusetts) (ITCS '12). Association for Computing Machinery, New York, NY, USA, 309–325.
6. Adeline Langlois and Damien Stehlé. 2015. Worst-case to average-case reductions for module lattices. Des. Codes Cryptography 75, 3 (June 2015), 565–599
7. Phillip Gajland, Bor de Kock, Miguel Quaresma, Giulio Malavolta, and Peter Schwabe. 2024. SWOOSH: Efficient Lattice-Based Non-Interactive Key Exchange. In 33rd USENIX Security Symposium, USENIX Security 2024, Philadelphia, PA, USA, August 14-16, 2024, Davide Balzarotti and Wenyuan Xu (Eds.). USENIX Association.

www.torontomu.ca/crl