Migrating Protocols to the Post-Quantum Setting: The Case of Signal's Double Ratchet

Benedikt Auerbach[1], Yevgeniy Dodis[2], Daniel Jost[2], Shuichi Katsumata[1], Thomas Prest[1]*, and Rolfe Schmidt[3]

[1] PQShield
[2] New York University
[3] Signal Messenger

Abstract. Secure Messaging apps are used by billions of people daily. However, due to imminent threat of a "Harvest Now, Decrypt Later" attack, secure messaging providers must react now in order to make their protocols *hybrid-secure*: at least as secure as before, but now also post-quantum (PQ) secure. Since many of these apps are internally based on the famous Signal's Double-Ratchet (DR) protocol, making Signal hybrid-secure is of great importance.

In fact, Signal and Apple already deployed various Signal-based variants with varying levels of hybrid security: PQXDH (only on the initial handshake), and PQ3 (on the entire protocol), by adding a *PQ-ratchet* to the DR protocol. Unfortunately, due to the large communication overheads of the Kyber scheme used by PQ3, real-world PQ3 performs this PQ-ratchet approximately every 50 messages. As we observe, the effectiveness of this amortization, while reasonable in the best-case communication scenario, quickly deteriorates in other still realistic scenarios; causing *many consecutive* (rather than 1 in 50) re-transmissions of the same Kyber public keys and ciphertexts (of combined size 2272 bytes!).

In this work, we present a new Signal-based, hybrid-secure messaging protocol with improved complexity compared to PQ3: the "*Triple Ratchet*" protocol.

- First, Triple Ratchet uses *erasure codes* to make the communication inside the PQ-ratchet provably balanced. This results in much better *worst-case* communication guarantees compared to PQ3.
- Second, we design a novel variant of Kyber, called Katana, with significantly smaller combined length of ciphertext and public key (which is the relevant efficiency measure for "PQ-secure ratchets"). For 192 bits of security, Katana improves this key efficiency measure by over 37%: from 2272 to 1416 bytes. In doing so, we identify a critical security flaw in prior suggestions to optimize communication complexity of lattice-based PQ-ratchets, and fix this flaw using recent advances in lattice security proof techniques.

This protocol has been developed with the Signal team, and some ideas discussed in this work have been brought into production by Signal, as explained in their blog post: https://signal.org/blog/spqr/.

* Thomas Prest is the designated presenter for this talk.

1 Background

The Signal Protocol, used by Signal, WhatsApp, Google RCS, and Facebook Messenger to protect the communications of billions of people worldwide, has widely been considered to be the gold standard for secure messaging. At its core, the *Double Ratchet* protocol [MP16a] provides important security properties called forward secrecy (FS) and post-compromise security (PCS), which informally state that if a user is compromised at a given time t, the confidentiality of messages before (resp. after) that time is preserved. Signal and the Double Ratchet protocol have been widely deployed with heavily scrutinized open source implementations, and have been formally analyzed in [CCD+20,ACD19,BFG+22,KBB17,BBD+21].

Post-Quantum Security. While this gives us confidence in the protocol today, these security guarantees are contingent on Diffie-Hellman (DH) assumptions for elliptic curves that can be broken by a quantum computer using Shor's algorithm [Sho94]. This is not only a future threat, since protocol transcripts collected today can be recorded and saved until a quantum computer is available, then decrypted in a *Harvest Now, Decrypt Later* (HNDL) attack.

Motivated by these concerns, Alwen et al. [ACD19] showed how to generalize the Signal protocol to work with any key encapsulation mechanism (KEM). As a result, one could potentially replace the DH-based Signal with a post-quantum variant; for example, using recently standardized Kyber (i.e., ML-KEM) [SAB+22]. Unfortunately, the resulting protocol is not sufficient for practical use, for two reasons.

1. **Hybrid security.** We want to preserve Signal's classical DH-based security. Thus, practically relevant post-quantum extensions of Signal should provide what is called *hybrid* security, and meaningfully combine the DH-based Double Ratchet with some post-quantum variant.
2. **Communication cost.** Kyber has a noticeable impact on the communication complexity, making it often impractical in the real world.

PQXDH and PQ3 As a result, the industry's transition to post-quantum Signal has been somewhat slower. First, Signal Messenger recently deployed PQXDH [KS23], an update to the X3DH [MP16b] handshake component of the Signal Protocol, and formally verified that the updated protocol provides HNDL protection without removing any of the previous DH-based security guarantees [BJKS24]. Since this was only an update to the initial protocol handshake, it does not provide any post-quantum PCS, one of the key features of the original Double Ratchet protocol.

To address this issue, Apple recently deployed PQ3 [App24], — a protocol similar to Signal, — that continuously adds Kyber-768 freshly shared secrets to the "root secrets" of the Double Ratchet protocol. Simplifications of the resulting PQ3 protocol have been analyzed by [Ste24] and machine verified by [LSB24], but they do not fully capture what is done in the real world. Concretely, [Ste24] only models Kyber public keys and ciphertexts as being sent with *every* asymmetric ratchet message. As we mentioned above, this is quite expensive, and Apple decided to perform a post-quantum ratchet approximately every 50 messages (or whenever they have not sent a fresh Kyber public key within a week), in order to amortize the large communication cost of Kyber keys and ciphertexts [Jac24]. Heuristically (and somewhat oversimplifying), this means that users have 50 "cheap" epochs (which do not help with post-quantum PCS), followed by 1 "expensive" epoch (which gives post-quantum PCS, but at a much slower rate than DH-based PCS).[4]

Communication Efficiency of PQ3. While the deployment of PQ3 was an amazing, and greatly celebrated advance of post-quantum cryptography in the real-world, there are at least two avenues where it can be substantially improved in terms of its communication efficiency. (And we address these deficiencies in this work, as our main contribution.)

First, while PQ3's "amortization trick" might provide a reasonable trade-off in the best-case scenario, when the communication pattern between the users is roughly balanced, the effectiveness of this amortization quickly deteriorates in less balanced, but *still realistic* real-world scenarios. This is because each of Signal's sending epochs lasts roughly until the peer responds (and advances the public ratchet). So it might be possible — and certainly happens from time to time — that the "expensive epoch" happens exactly when one of the users is offline for an extended period of time,[5] resulting in *many consecutive re-transmissions repeating the same (long!) Kyber public keys and ciphertexts.*

Second, we already mentioned that Kyber's public key and ciphertext (and each "expensive epoch" message in PQ3 sends both) is much larger than the single DH group element sent by classical Signal. Concretely, (1088+1184=2272) bytes compared to 32 bytes, which is 71 times longer! Thus, any concrete efficiency improvement over using the generic (post-quantum) KEM advocated by [ACD19] will likely result in much faster PCS. For example, it allows reduction of the number 50 in PQ3's heuris-

[4] This heuristics is related to "on-demand" ratcheting suggested by [CDV21].
[5] E.g., when using devices which are periodically turned off.

tic amortization, while maintaining similar communication complexity. In that regard, [ACD19,DG19,LKS23] already described lattice-based protocols (either directly for Kyber, or equivalent variants over other rings) which seemingly achieve this goal. Unfortunately, the protocol of [DG19] achieves almost no saving (less than 2%, as noticed by the authors) as compared to using the generic Kyber, while the protocols of [ACD19,LKS23] contain a critical subtle security flaw (which we found in this work) invalidating these analyses. Thus, prior to this work we did not have optimized variants of Kyber which would significantly reduce the communication complexity of post-quantum Signal or its variants.

2 Out contributions

In this work, we provide a practical hybrid-secure Double Ratchet protocol called the *Triple Ratchet* protocol.[6] Our name is taken from the fact that we use (i) a post-quantum public ratchet, (ii) a classical public ratchet, and (iii) symmetric ratchet. Compared to PQ3, it addresses both of the communication deficiencies mentioned above. An overview of our result is given in Fig. 1. At a high level, our work consists of two technical contributions.

Erasure codes. We use *erasure codes* to evenly distribute the communication inside the "post-quantum" ratchet (i.e., PQ CKA protocol in Fig. 1), without any amortization heuristics. This is illustrated in Fig. 1. At a high level, instead of sending one long message every 50 epochs, we encode the resulting message using an erasure code, and send a fresh chunk of this encoding with every message. For example, we could set parameters so that the long message will be decoded from *any* 50 chunks. Then, in a fully balanced setting we would still achieve PCS in 50 epochs and same communication as PQ3, but without any amortization. However, we start getting big savings in the unbalanced cases, when some epochs are long-lasting. For such epochs, PQ3's strategy could be viewed as using a hugely inefficient *repetition code*, leading to a big communication penalty; e.g., a factor of up to 50 in our "PQ3-inspired" example. We detail this and give an overview of some of the technical challenges we resolved in our presentation.

A new CKA*:* Katana*.* We design a novel *Continuous Key Agreement* (CKA) protocol based on Kyber, which we call Katana-CKA, which could be used

[6] This should not be confused with the protocol by [BFG+22] with the same name.

inside our Triple Ratchet protocol. This is illustrated in Fig. 1. Recall, CKA was a generic building block used by [ACD19] to abstract out the design of the Double Ratchet Protocol. [ACD19] then presented a generic KEM-based CKA, where every message contained a KEM public key and ciphertext. When applied to Kyber at security level 192 bits, this gives CKA *messages of size 2272 bytes*. In contrast, for the same security level Katana-CKA uses *messages of size 1416 bytes*, saving over 37% over the generic construction.

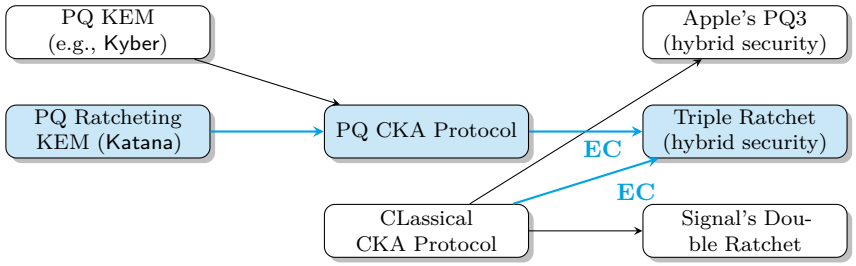

Fig. 1: Different types of PQ KEM can be compiled into a PQ CKA protocol with different security and efficiency profiles. A classical CKA protocol leads to Signal's Double Ratchet protocol. This classical CKA and a PQ CKA based on Kyber, combined in a natural manner, leads to Apples's PQ3. Instead, using a PQ CKA based on Katana and performing erasure encoding for the hybrid composition leads to our Triple Ratchet protocol. The blue boxes and arrows indicate our construction. ECC = "Erasure codes".

A security flaw and a fix. We notice that Katana-CKA is closely related to what previous works called "optimized" lattice-based CKA [ACD19,LKS23], but instantiated with a carefully chosen variant of Kyber. As we mentioned, however, we identify a critical flaw in the previous analyses of this "optimized" KEM, and non-trivially fix them with a novel proof relying on the recently introduced Hint-MLWE assumption [KLSS23].

In more detail, we first generalize the KEM-based CKA from [ACD19] to work with what we call a *Ratcheted KEM* (RKEM). On a high level, RKEM abstracts KEM properties in a way which allows a freshly sampled ciphertext also be used as "part" of a different KEM public key. In essence, this is precisely why the original DH-based CKA of Signal saved a factor of 2 in communication, when compared to the generic KEM-based DH construction. And this is why RKEM is precisely fitted for the use inside a CKA. Once we define RKEM and show that it generically implies CKA, it allows us to focus on a cleaner RKEM primitive, which we then construct from the Hint-MLWE assumption. We call the resulting RKEM Katana, which explains the name Katana-CKA for our new CKA.

Comparison with PQ3. We wrap up by providing an efficiency analysis of our Triple Ratchet protocol by comparing it with Apple's PQ3 and a variant of our Triple Ratchet instantiated with Kyber (i.e., we use the standard PQ KEM to construct the PQ CKA protocol in Fig. 1). The latter variant illustrates the effectiveness of only relying on erasure codes. The efficiency comparison is found in Table 1. Here, we assume a simple model of unbalanced communication where every sender has a probability p of sending another message before receiving all incoming messages, independent of previous events. In row one we use $p = 0$ to capture perfectly balanced communication. In row two we use $p = 0.5$ to conservatively approximate the sending behavior of two online parties using typing indicators and read receipts, and we see that at this point both Triple Ratchet instantiations have an advantage over PQ3. Finally, in row 3, we use $p = 0.9$ to approximate the behavior of a device that is offline for hours at a time, where PQ3 is more than 4 times as expensive as Triple Ratchet with Katana. The talk will include more discussion on our efficiency analysis and the tradeoff between security.

	PQ3	TR with Kyber-768	TR with Katana ($\lambda = 192$)
$p = 0$	8 144	11 270	8 722
$p = 0.5$	12 760	11 615	8 989
$p = 0.9$	49 688	14 375	11 125

Table 1: Expected communication cost in bytes to attain PCS for PQ3 and Triple Ratchet. See text for the parameter p. PQ3 is assumed to send two Kyber-768 encapsulation keys and ciphertexts every 50 messages. Triple Ratchet with Kyber-768 (resp. Katana) uses a post-quantum CKA based on Kyber-768 (resp. Katana with $\lambda = 192$). This includes base message cost of 36B for PQ3 and 46B for Triple Ratchet to account for the overhead of sending counters and DH keys but excludes the 64B signature used by PQ3 for fair comparison.

Original work. This submission is based entirely on *Triple Ratchet: A Bandwidth Efficient Hybrid-Secure Signal Protocol* [DJK+25], an article by the same set of authors which has been published at IACR EUROCRYPT 2025 and has also been presented at IACR Real World Crypto 2025.

SSTIC presentation outline. If accepted at SSTIC, our presentation will be in French. While the original submission was aimed at cryptographers, we aim to make this presentation more geared towards an audience of security practitioners. We will first present the threat model and design rationale of the original Double Ratchet protocol. We will then highlight how this design breaks when trying to port it naively to the post-quantum setting. Finally, we will present our improved protocol and explain how it is tailored to the quirks and unique properties of post-quantum cryptographic primitives.

References

ACD19. Joël Alwen, Sandro Coretti, and Yevgeniy Dodis. The double ratchet: Security notions, proofs, and modularization for the Signal protocol. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part I*, volume 11476 of *LNCS*, pages 129–158. Springer, Cham, May 2019.

App24. Apple Security Engineering and Architecture (SEAR). iMessage with PQ3: The new state of the art in quantum-secure messaging at scale, 2 2024. Available at https://security.apple.com/blog/imessage-pq3/.

BBD+21. Karthikeyan Bhargavan, Abhishek Bichhawat, Quoc Huy Do, Pedram Hosseyni, Ralf Küsters, Guido Schmitz, and Tim Würtele. DY*: A modular symbolic verification framework for executable cryptographic protocol code. In *2021 IEEE European Symposium on Security and Privacy*, pages 523–542. IEEE Computer Society Press, September 2021.

BFG+22. Alexander Bienstock, Jaiden Fairoze, Sanjam Garg, Pratyay Mukherjee, and Srinivasan Raghuraman. A more complete analysis of the Signal double ratchet algorithm. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 784–813. Springer, Cham, August 2022.

BJKS24. Karthikeyan Bhargavan, Charlie Jacomme, Franziskus Kiefer, and Rolfe Schmidt. Formal verification of the PQXDH post-quantum key agreement protocol for end-to-end secure messaging. In Davide Balzarotti and Wenyuan Xu, editors, *USENIX Security 2024*. USENIX Association, August 2024.

CCD+20. Katriel Cohn-Gordon, Cas Cremers, Benjamin Dowling, Luke Garratt, and Douglas Stebila. A formal security analysis of the Signal messaging protocol. *Journal of Cryptology*, 33(4):1914–1983, October 2020.

CDV21. Andrea Caforio, F. Betül Durak, and Serge Vaudenay. Beyond security and efficiency: On-demand ratcheting with security awareness. In Juan Garay, editor, *PKC 2021, Part II*, volume 12711 of *LNCS*, pages 649–677. Springer, Cham, May 2021.

DG19. Nir Drucker and Shay Gueron. Continuous key agreement with reduced bandwidth. In *International Symposium on Cyber Security Cryptography and Machine Learning*, pages 33–46. Springer, 2019.

DJK+25. Yevgeniy Dodis, Daniel Jost, Shuichi Katsumata, Thomas Prest, and Rolfe Schmidt. Triple ratchet: A bandwidth efficient hybrid-secure signal protocol. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology -*

EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques Madrid, Spain, May 4-8, 2025, Proceedings, Part VIII, volume 15608 of *Lecture Notes in Computer Science*, pages 302–331. Springer, 2025.

Jac24. Frederic Jacobs. Designing iMessage PQ3: Quantum-secure messaging at scale. Invited talk at the Real World Crypto Symposium 2025, 2024.

KBB17. Nadim Kobeissi, Karthikeyan Bhargavan, and Bruno Blanchet. Automated verification for secure messaging protocols and their implementations: A symbolic and computational approach. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017*, pages 435–450. IEEE, 2017.

KLSS23. Duhyeong Kim, Dongwon Lee, Jinyeong Seo, and Yongsoo Song. Toward practical lattice-based proof of knowledge from hint-MLWE. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 549–580. Springer, Cham, August 2023.

KS23. Ehren Kret and Rolfe Schmidt. The pqxdh key agreement protocol, 2023. Available at https://signal.org/docs/specifications/pqxdh/.

LKS23. Joohee Lee, Jihoon Kwon, and Ji Sun Shin. Efficient continuous key agreement with reduced bandwidth from a decomposable kem. *IEEE Access*, 11:33224–33235, 2023.

LSB24. Felix Linker, Ralf Sasse, and David Basin. A formal analysis of apple's iMessage PQ3 protocol. Cryptology ePrint Archive, Paper 2024/1395, 2024.

MP16a. Moxie Marlinspike and Trevor Perrin. The double ratchet algorithm, 2016. Available at https://signal.org/docs/specifications/doubleratchet/.

MP16b. Moxie Marlinspike and Trevor Perrin. The x3dh key agreement protocol, 2016. Available at https://signal.org/docs/specifications/x3dh/.

SAB+22. Peter Schwabe, Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Gregor Seiler, Damien Stehlé, and Jintai Ding. CRYSTALS-KYBER. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

Sho94. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994.

Ste24. Douglas Stebila. Security analysis of the iMessage PQ3 protocol. Cryptology ePrint Archive, Report 2024/357, 2024.