POSTER: On Post-Quantum Identification in 5G

Mohamed Taoufiq Damir
Department of Computer Science
University of Helsinki
Helsinki, Finland
mohamed.damir@helsinki.fi

Valtteri Niemi
Department of Computer Science
University of Helsinki
Helsinki, Finland
valtteri.niemi@helsinki.fi

ABSTRACT

We introduce PQ 5G AKA, a prototype for possible extensions of the current 5G authentication and key agreement protocol to the post-quantum setting, we further analyse the computational and communication complexities of our prototype using potential post-quantum KEMs.

CCS CONCEPTS

• **Security and privacy** → *Cryptography*; • **Network security** → **Mobile and wireless security**.

KEYWORDS

5G AKA, post-quantum cryptography, privacy, IMSI catchers.

ACM Reference Format:
Mohamed Taoufiq Damir and Valtteri Niemi. 2022. On Post-Quantum Identification in 5G. In *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '22), May 16–19, 2022, San Antonio, TX, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3507657.3529657

1 INTRODUCTION

Compared to previous mobile telephony generations, the Fifth Generation (5G) mobile standards offer a development platform for other technologies, e.g., connected cars, industrial automation, smart homes and smart large event venues. Clearly, such a system cannot be complete without considering security and user's privacy challenges. The 3^{rd} Generation Partnership Project (3GPP), the organism responsible of developing mobile communication standards, released `TS 33.501` which is the technical specification for security architecture and procedures for 5G system. As in the precedent mobile technology generations, 5G security protocols laid mainly on symmetric cryptographic primitives, i.e., SNOW, AES. Moreover, the security design of 5G is enforced by a developed key hierarchy and by the possibility of considering 256−bit keys (instead of just 128 bits). The use of public key cryptography (PKC) is extended in the 5G security procedures. Elliptic curve cryptography (ECC) was introduced to protect the user's identity, and the internet protocol TLS is used to secure the interaction between multiple services operating on the network.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
WiSec '22, May 16–19, 2022, San Antonio, TX, USA.
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9216-7/22/05.
https://doi.org/10.1145/3507657.3529657

Coincidentally, the introduction of PKC in 5G befell while quantum computers started to be considered as a realistic threat to existing PKC. Public-key cryptography schemes inherit their security from the hardness of some well-studied mathematical problems. For example, RSA is based on the difficulty of integer factoring and ECC is based on the difficulty of calculating discrete logarithms on elliptic curves. In [5], Shor showed that these two problems can be efficiently solved by a sufficiently large-scale quantum computer. On the other hand, Grover [1] proposed a quantum algorithm allowing brute-forcing an n-bit key in $O(2^{\frac{n}{2}})$ time instead of the $O(2^n)$ time required by classical computers. Consequently, primitives[1] providing 128-bits of (classical) security, need to have a key length of at least 256 bits to offer the same hypothetical security level against quantum adversaries, which is technically feasible in the case of 5G as it supports 256-bits keys.

For the above reasons, efforts have focused on public-key cryptography. In December 2016, *the National Institute of Standards and Technology* (NIST) in United States initiated a standardization process for quantum secure asymmetric primitives, that is by announcing a call for proposals for post-quantum public-key encryptions and signatures schemes. This lead us to the main motivation behind the present note, where we carry out a comparative analysis of migrating various PQC schemes in 5G.

2 IDENTITY PRIVACY IN MOBILE NETWORKS

Mobile networks architecture consists of three entities. The *User Equipment* (UE), the *Home Network* (HN) and the *Serving Network* (SN). The UE includes the *Mobile Equipment* (ME) and the *Universal Subscriber Identity Module* (USIM) which can be illustrated respectively as a phone and a SIM card.

In 5G mobile systems, networks assign to every USIM a unique identifier known as the *Subscription Permanent Identifier* (SUPI), that is the *International Mobile Subscriber Identity* (IMSI) in the case of 4G. This globally unique identifier is used to identify the UE during the communication, but it is also used to locate and track the user. In this setting, intercepting the SUPI/IMSI, , and especially the MSIN by an attacker, known as an IMSI catcher, is a serious privacy threat.

To avoid IMSI catching, the serving network assigns to the USIM a *Globally Unique Temporary User Equipment Identity* (GUTI), which is a temporary frequently changing identifier. In 4G, there are circumstances where the user is required to send the permanent identifier as a plain text. For example, when the network is unable to derive the IMSI/SUPI from the GUTI, or when a temporary identifier (GUTI) is not yet assigned, which might accrue during the first contact to a new network in the case of roaming. This issue is

[1] Especially symmetric cryptography primitives

prevented in 5G by sending a concealed SUPI called the *Subscription Concealed Identifier* (SUCI) [see 3GPP TS.33.501]. The SUCI is generated by the UE using an *Elliptic Curve Integrated Encryption Scheme* (ECIES) as defined in 3GPP TS 33.501 Annex C.

The SUPI concealment starts by generating a pair of ephemeral public/private keys at the UE. The 3GPP specification allows two ECIES profiles, Profile A which is based on *Curve25519* [7] and Profile B which uses the curve *Secp256r1* [4]. Next a shared key K_s is obtained by both the UE and the HN using a Deffie-Hellman key exchange procedure using the HN public key, pk_{HN}, stored at the USIM. A derived key K_D is obtained from K_s using the *Key Derivation Function* (KDF) ANSI-X9.63-KDF with the hash SHA-256 [3GPP TS 33.220]. Then the SUPI is encrypted using K_D and the AES128 algorithm, where the first 16 bytes of the derived key are used as an encryption key, the following 8 bytes are used as an AES nonce and the next 8 bytes are used as an initial value for the AES encryption in counter mode. Finally, a mac key is given by the remaining 32 bytes. Note that the above ECIES scheme can be seen as a *key encapsulation mechanism* (KEM) and *data encapsulation mechanism* (DEM), see [6].

3 PQ 5G AKA

The authentication and key agreement procedure in 5G supports two protocols, namely, the 5G-*Authentication and Key Agreement* and the *Extensible Authentication Protocol* denoted 5G AKA and EAP-AKA' respectively. In fact, the two protocols are similar in our context, thus, is enough to consider the 5G AKA [3GPP TS 33.501]. We distinguish two phases in 5G AKA. The identification phase which consists of the ECIES KEM/DEM, and the authentication phase which is based on symmetric cryptography primitives. In 5G AKA, the identification phase can be initiated either by the UE or the SN, while the starting of authentication is always triggered by the SN. In the identification phase, the UE sends its concealed identity to the SN, then the SN forwards the information to the HN over a channel that is secured by means independent of UE. Next, the HN starts the authentication phase by generating a challenge and an authentication token (based on symmetric primitives)[see 3GPP TS 33.501]. Our aim is to extend such a protocol to a quantum resistant framework.

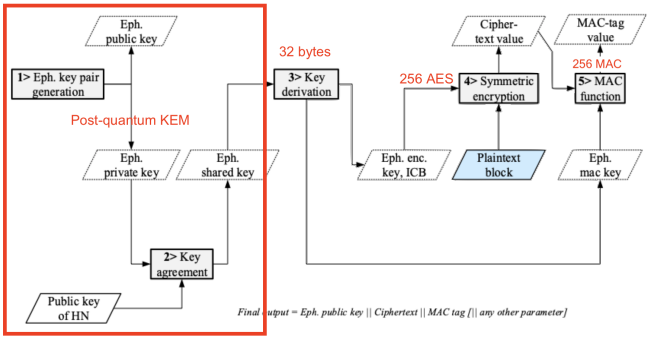


Figure 1: PQ encryption at UE

For this end, we define the *Post-Quantum 5G AKA* (PQ 5G AKA) protocol to be a copy of 5G AKA, but with a post-quantum asymmetric KEM (instead of Diffie-Hellmann in ECIES) and symmetric primitives using 256 keys instead of 128 bits. Figure 1 illustrates the above operations at the UE.

3.1 Performance Analysis

In this section we compare the performance of the SUPI encryption part in PQ 5G AKA using NIST Round 3 KEM finalists. To stay consistent with the 3GPP 33.501 security requirements, i.e., 128-bits security level, we only compare KEMs affording an equivalent (hypothetical) quantum security level and producing a 32 bytes shared key [see Table 1].

As stated previously, the UE sends the SUCI, the ciphertext c and the Mac. As the sizes of the Mac and SUCI are basically the same using any of the above KEMs, then the communication performance is measured by the size of the ciphertext. Clearly, Classic McEliece outperforms lattice based schemes with respect to this metric. However, recall that pk_{HN} is stored at the USIM, while sk_{HN} is stored at the HN. Thus, from a storage point of view, Classic McEliece will require a considerably larger space at the SIM card compared to lattice based KEMs as depicted in Table 1.

Algorithm	sk_{HN}	pk_{HN}	Ciphertext	Shared secret
Classic-McEliece-348864	6452	261120	128	32
Kyber 512	1632	800	768	32
NTRU-HPS-2048-509	935	699	699	32
LightSaber-KEM	1568	672	736	32

Table 1: Round 3 finalists (communication cost in bytes)

	Iterations	Mean (μs)	stdev
Classic-McEliece-348864			
Keygen	12	87410.333	15607.684
Encaps	83460	11.982	2.526
Decaps	24979	40.034	4.392
Kyber512			
Keygen	115904	8.628	1.736
Encaps	91823	10.891	1.844
Decaps	127009	7.874	1.568
NTRU-HPS-2048-677			
Keygen	22517	44.412	9.377
Encaps	124147	8.055	1.5551
Decaps	89097	11.224	1.1599
LightSaber-KEM			
Keygen	77098	12.971	2.603
Encaps	68478	14.603	2.540
Decaps	75635	13.222	3.932

Table 2: Round 3 finalists (each computation run for 1 second)

We emphasize that pk_H and the cipher text c (sent over the radio channel) are independent of the used ME, that is also the case for the decapsulation process which is performed at the HN. Hence, in our

context, carrying out the experiments on a mobile device instead of a workstation is only relevant to rank the mentioned algorithms with respect to the encapsulation part. The choice of the device won't affect the ranking/classification of the used encapsulation algorithms. Thus, and to illustrate the performance of the used KEMs, we use a workstation and we leave leading experiments on mobile devices to a future work.

Table 2 is generated using liboqs [3] which is an open source C library for quantum-safe cryptographic algorithms, where the implementations are directly from NIST submissions. The computations are performed on a 3.5 GHz Core ($i7-7567U$). We observe that in terms of encapsulations, lattice based schemes and in particular, Kyber and NTRU slightly outperform Classic McEliece. For decapsulation, lattice based cryptography seems to out perform code based cryptography. In summary, Classic McEliece (code based cryptography) is the best choice in terms of optimising the size of the cipher text sent over the radio channel, while lattice based cryptography is advantageous with respect to the required storage space for pk_H at the USIM. Unfortunately, quantum attacks are not the only issues in 5G AKA. Indeed, such a protocol is also vulnerable to the so-called linkability attacks. Thus, we argue that an implementation of PQC in 5G should also take in account such attacks and further security properties, i.e., backward and forward secrecy, we leave studying these issues to a future work.

As mentioned previously, 3GPP 33.501 specifies that the SUPI is concealed using a 128 bits AES. In the model above (PQ 5G), and to resist Grover's search algorithm, we assumed that public key PQC is implemented alongside symmetric primitives of a double key size, that is 256 bits in our case. Indeed, various reports, see for instance [2], state that quantum attacks will not affect symmetric cryptography in the near future. Thus, it is also conjectured that implementing public key PQC on the top of the currently used symmetric ciphers[2] won't affect the security of relatively "short term" infrastructures, which is more or less the case for mobile technologies. To conclude our study, we led a comparative study of the SUPI encryption/decryption at the UE and the HN, where we implemented both, the actual ECIES profiles and NIST PQ KEMs finalists, that is by using the current symmetric primitives recommended by 3GPP 33.501, i.e., 128 bits AES, SHA-256 and HMAC–SHA-256. For our implementation, we used OpenSSL and liboqs via a Python wrapper.

Algorithm	At UE (μs)	At HN (μs)
ECIES Curve25519	49.221	48.110
ECIES Secp256r1	131.201	131.001
Classic-McEliece-348864	16.167	49.319
LightSaber-KEM	17.039	16.563
NTRU-HPS-2048-509	14.722	20.015
Kyber512	14.351	10.101

Table 3: Running time of the operations at the UE/HN (each computation run for 1s)

[2]Without doubling the key size

In Table 3, we illustrate the running time of the SUPI encryption (resp. decryption) at the UE (resp. HN). For the key sharing between the UE and the HN, we use NIST PQC finalist KEMs and the ECIES algorithms standardised by 3GPP, namely, the profiles given by Curve25519 and Secp256r1. Moreover, and for the actual SUPI encryption, we use the symmetric primitives defined in 3GPP 33.501. We report that computationally, PQC clearly outperforms ECIES. On the other hand, ECIES Curve25519 and Secp256r1 both have public key and encapsulation ciphertext each of size 32 bytes, which is way shorter than those of PQC KEMs as depicted in Table 1. We recall that the public key is stored at the USIM and the encapsulation ciphertext is sent over the radio channel. Therefore, we conclude that ECIES offers better storage and communication cost compared to PQC.

ACKNOWLEDGMENT

This work was supported by the Business Finland Consortium Project "Post-Quantum Cryptography" under Grant 754/31/2020.

REFERENCES

[1] Lov K Grover. 1996. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 212–219.
[2] Erik Thormarker John Preus Mattsson, Ben Smeets. 2021. *Quantum technology and its impact on security in mobile networks.* Technical Report. Ericsson (December 2021).
[3] liboqs. 2019. liboqs. https://github.com/open-quantum-safe/liboqs
[4] Minghua Qu. 1999. Sec 2: Recommended elliptic curve domain parameters. *Certicom Res., Mississauga, ON, Canada, Tech. Rep. SEC2-Ver-0.6* (1999).
[5] Peter W Shor. 1994. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*. Ieee, 124–134.
[6] Victor Shoup. 2001. A proposal for an ISO standard for public key encryption (version 2.1). *IACR e-Print Archive* 112 (2001).
[7] Sean Turner, Adam Langley, and Mike Hamburg. 2016. *Elliptic curves for security.* Technical Report. IETF RFC 7748 (January 2016).