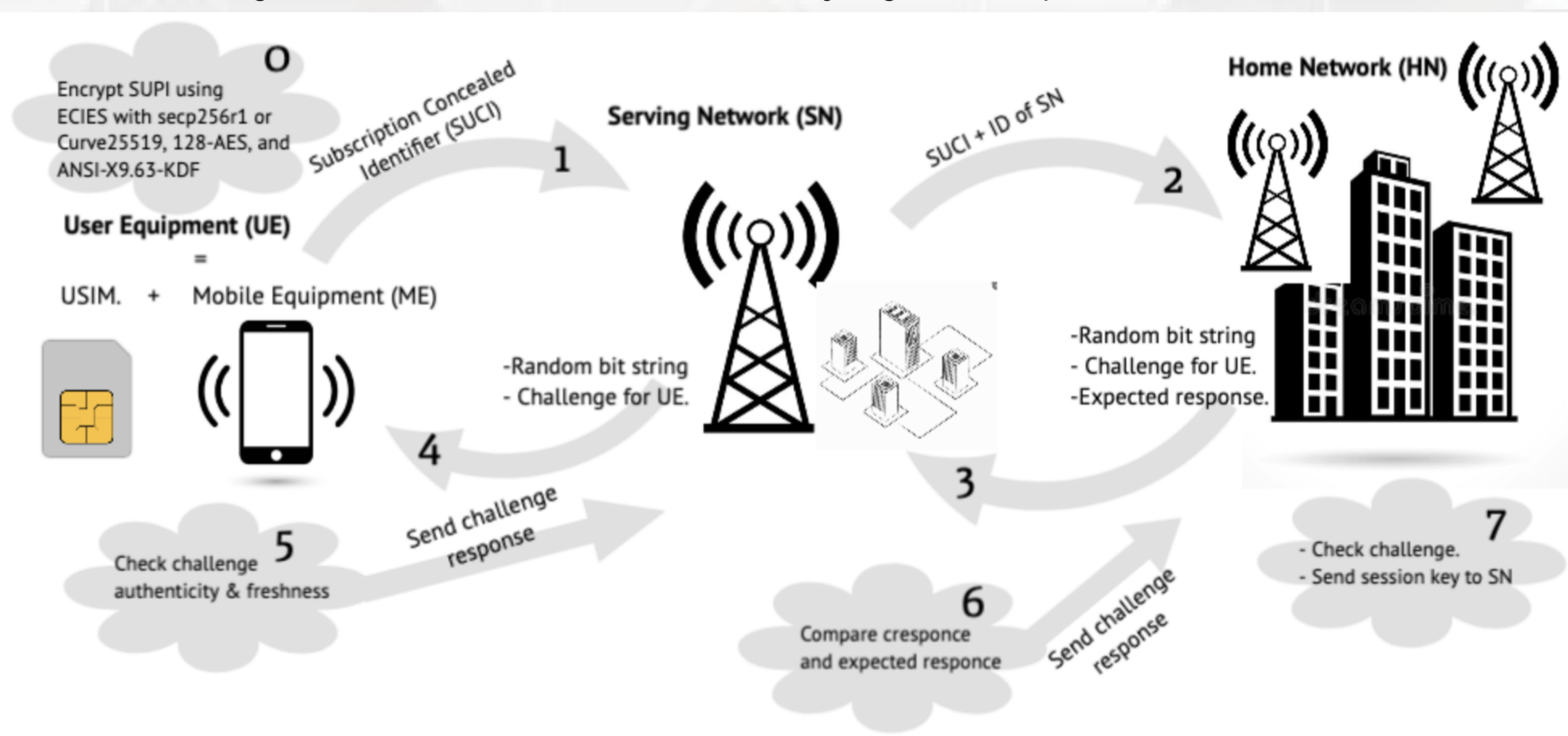


## INTRODUCTION

In 5G networks, security issues are mainly handled using symmetric key cryptography, i.e, AES, SNOW 5G...etc. However, public key cryptography is still a crucial ingredient in the authentication process in 5G technology. In fact, the current Authentication and Key Agreement protocol in 5G (5G AKA) is based on an Elliptic Curve Integrated Scheme (ECIES) [3GPP TS 33.501], hence the interest in studying the possibility of extending 5G AKA to a quantum resistance protocol by replacing ECIES by Post-Quantum Cryptography (PQC). Unfortunately, quantum attacks are not the only issue facing 5G AKA. Indeed, further privacy and security threats were discovered shortly after adopting 5G AKA by 3GPP, to name a few we mention, linkability attacks and the lack of forward/backward security; thus, it is natural to consider such issues while implementing PQC in 5G AKA. In the present work, we first study the extension of the 5G AKA protocol to a quantum resistant framework, that is by investigating NIST round 3 finalist *Key Encapsulation Mechanisms* (KEMs). Then we will use our understanding of post-quantum KEMs to propose an upgrade to 5G AKA that we call 5G AKA<sup>\*</sup>, which is a standard compatible post-quantum authentication and key agreement protocol offering both forward and backward security at the user side and resistance to known linkability attacks.

## Post-Quantum Identification in 5G

Figure 1 – 5G Authentication and Key Agreement protocol (5G AKA)



We investigate the communication and computational costs of NIST round 3 finalist KEMs in the 5G identification. We recall that  $pk_{HN}$  and  $sk_{HN}$  are stored at the UE and HN respectively, while the ciphertext  $c$  is sent over the radio channel.

| Algorithm               | $sk_{HN}$ | $pk_{HN}$ | Ciphertext $c$ | 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)

We led a comparative study of the running time of the the operations at the UE and the HN, where we implemented the current standardized ECIES profiles, namely, Curve 25519 and Secp256r1, and NIST round 3 finalist KEMs. Our implementation uses Liboqs and OpenSSL on a 3.5 GHz Core i7 workstation.

| Algorithm               | At UE ( $\mu s$ ) | At HN ( $\mu s$ ) |
|-------------------------|-------------------|-------------------|
| ECIES                   | 49.221            | 48.110            |
| ECIES                   | 131.201           | 131.001           |
| 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 2 – Running time of the operations at the UE/HN

## Operations at the HN, ME and USIM in 5G AKA

Figure 3 – Authentication Vector generation at the HN

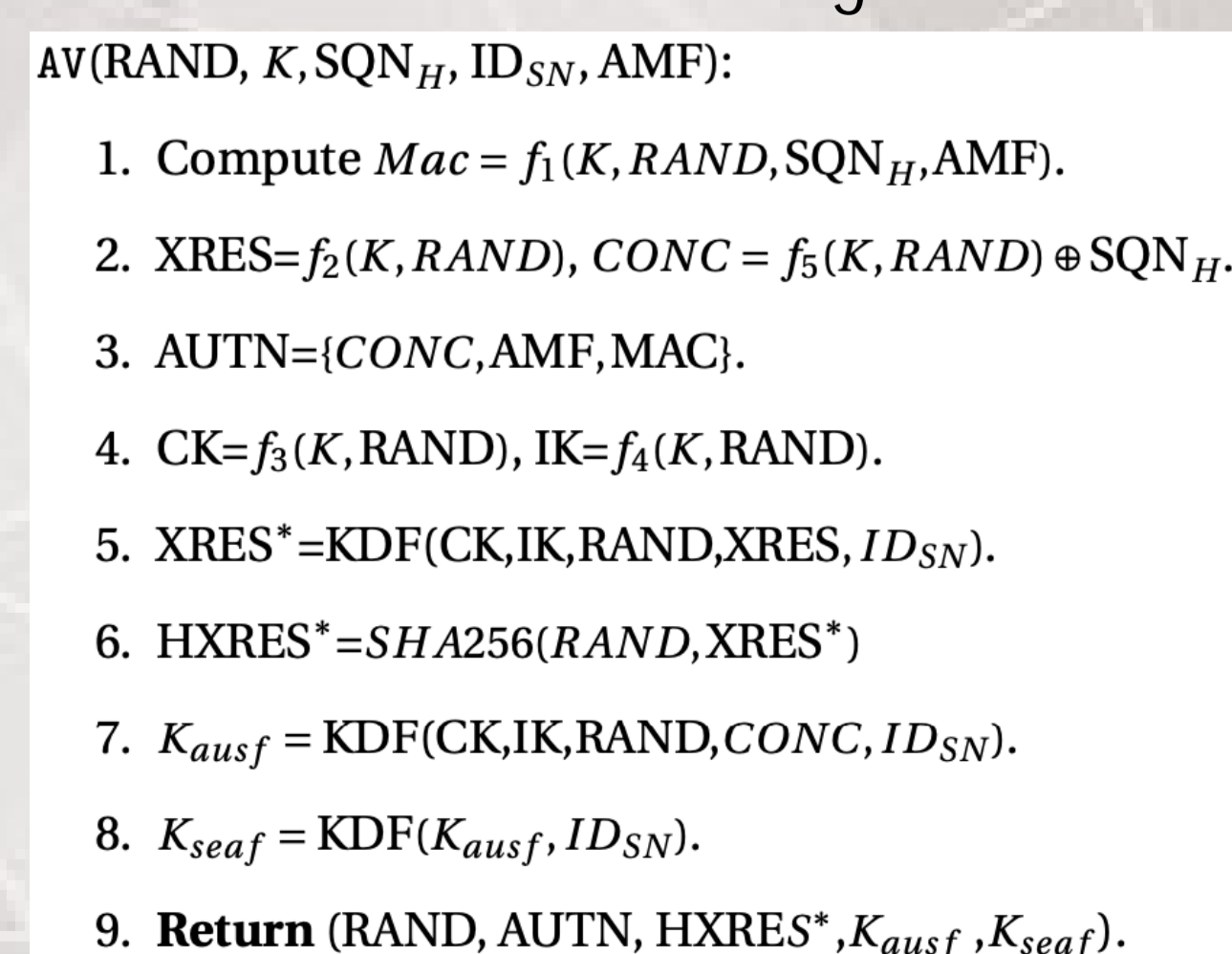


Figure 4 – Operations at the Mobile Equipment

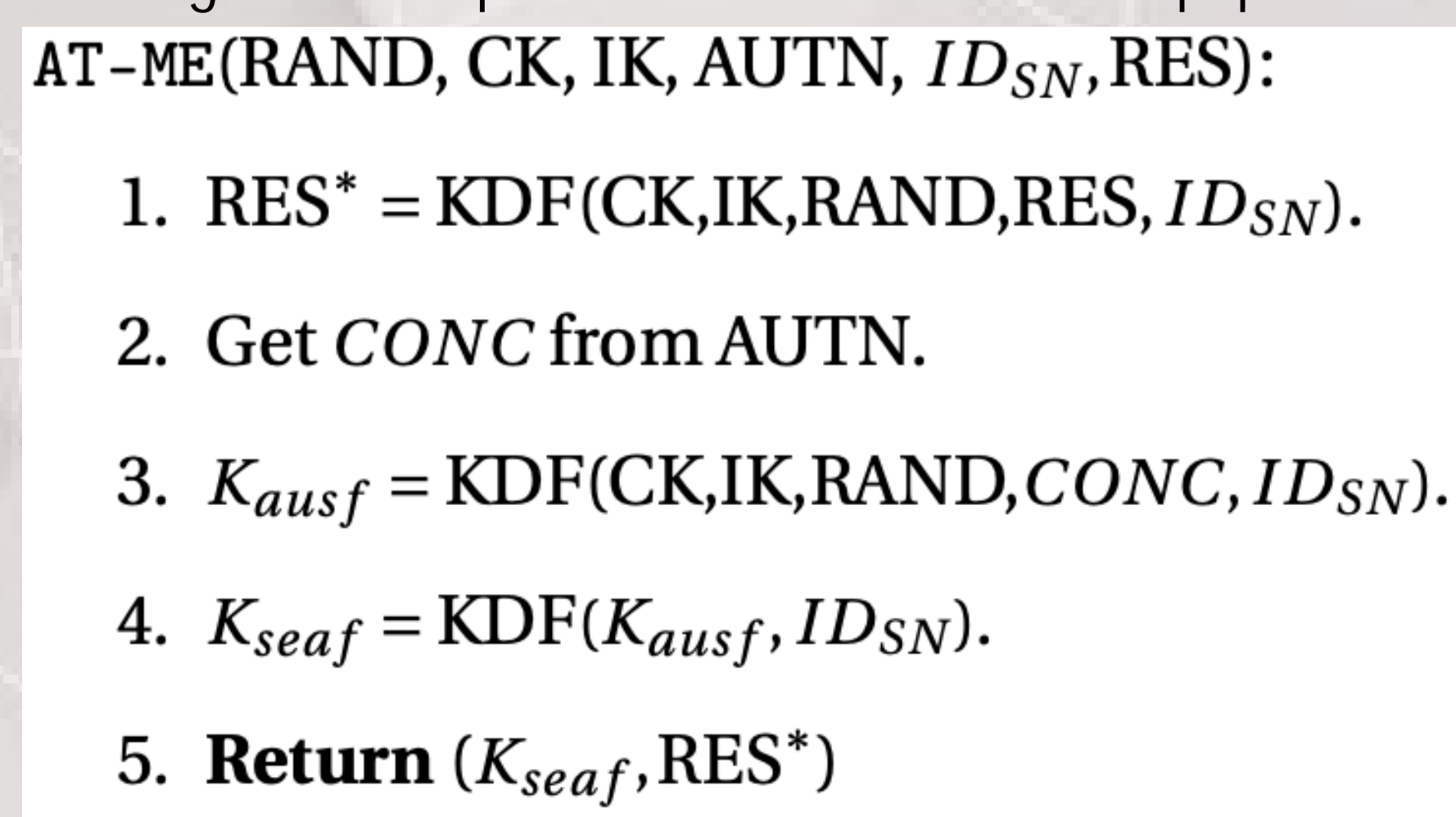
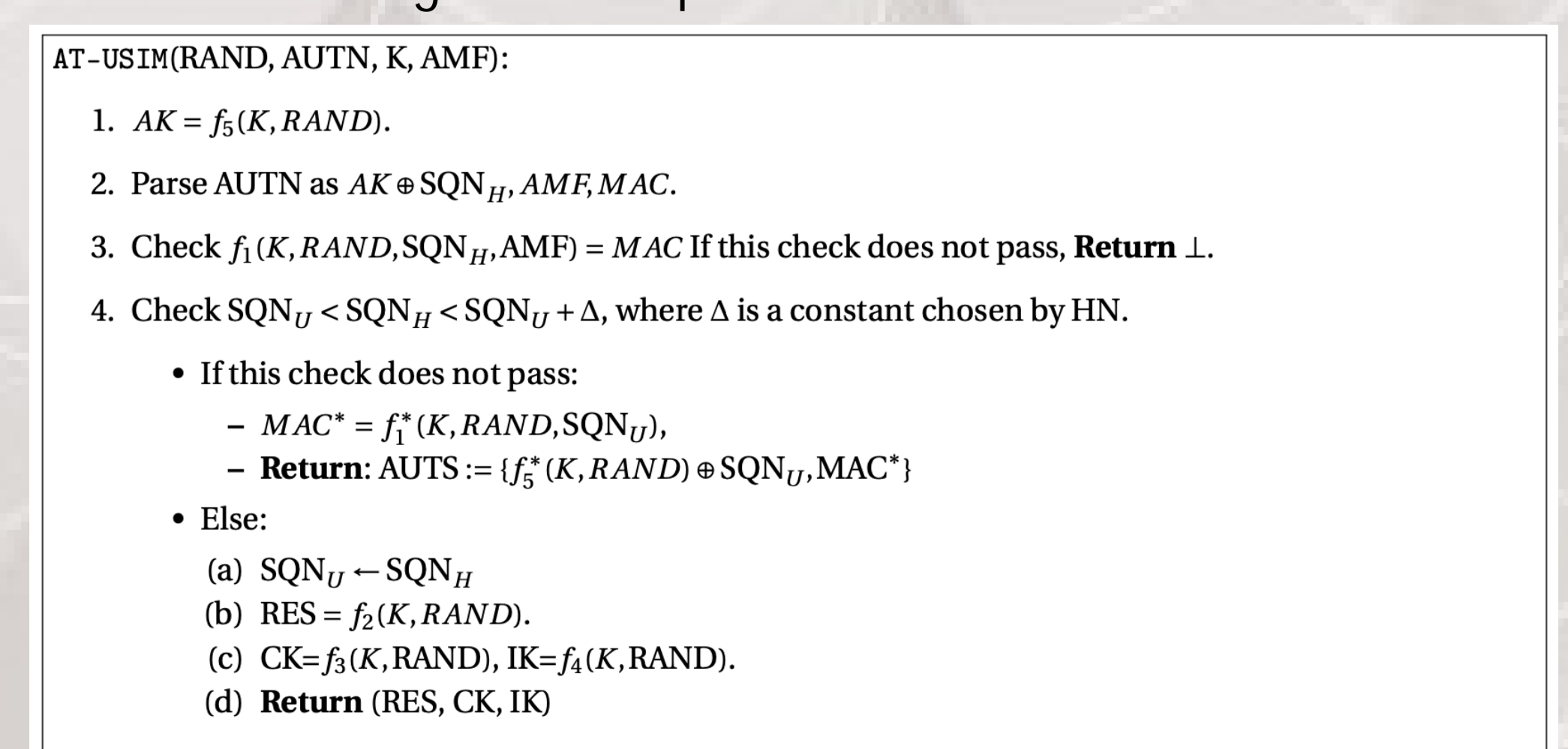
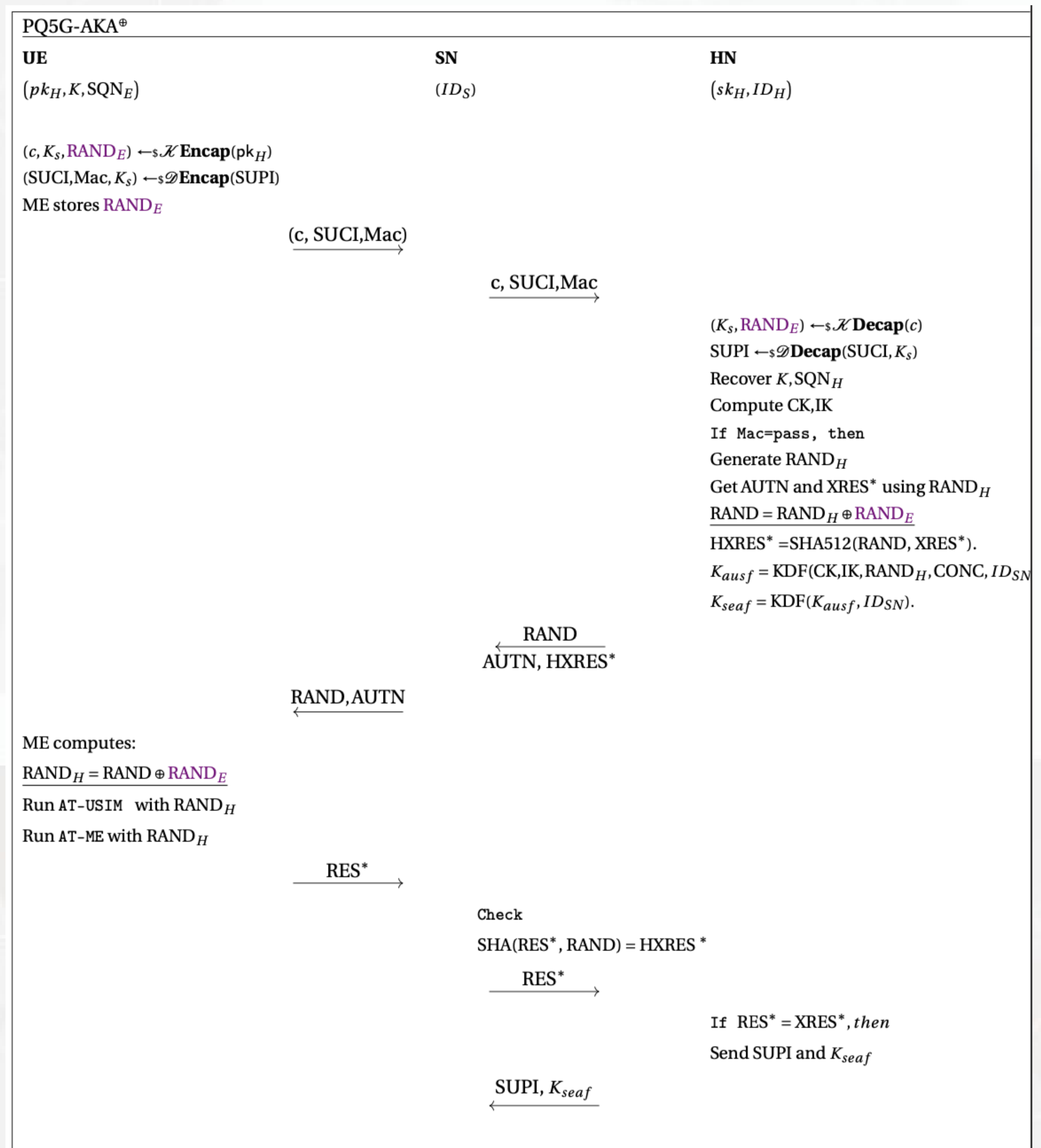


Figure 5 – Operations at the USIM



## 5G AKA

Figure 2 – 5G AKA



## Advantages of 5G AKA

- Quantum secure (used quantum resistant algorithms).
- Resistance to linkability attacks (ability to detect replied messages).
- Forward and Backward secrecy at the UE (the session key depends on the freshly generated randomness).
- Backward compatibility features:
  - No changes at the USIM (we can keep the old SIM card).
  - Requires the ME to perform one extra XOR operation.
  - No changes at the SN in the case of SUPI.

Note that the temporary identifier called *Globally Unique Temporary User Equipment Identity* (GUTI) is favoured over the use of SUPI, clearly because the use of GUTI does not require extra asymmetric encryption, thus, in theory, no need to consider PQC in the GUTI case. However, our protocol covers such case by requiring the SN to generate a random bit string  $RAND_S$  which will play the roll of  $RAND_E$  in the case of SUPI and it will be sent to the UE by the SN during *the GUTI assignment phase*:

- The SN generates a GUTI and  $RAND_S$  then sends  $c = Enc_{K_{session}}(GUTI, RAND_S)$
- The UE decrypts  $c$  and stores GUTI and  $RAND_S$