5G Security in the Post-Quantum Era

21st May 2025

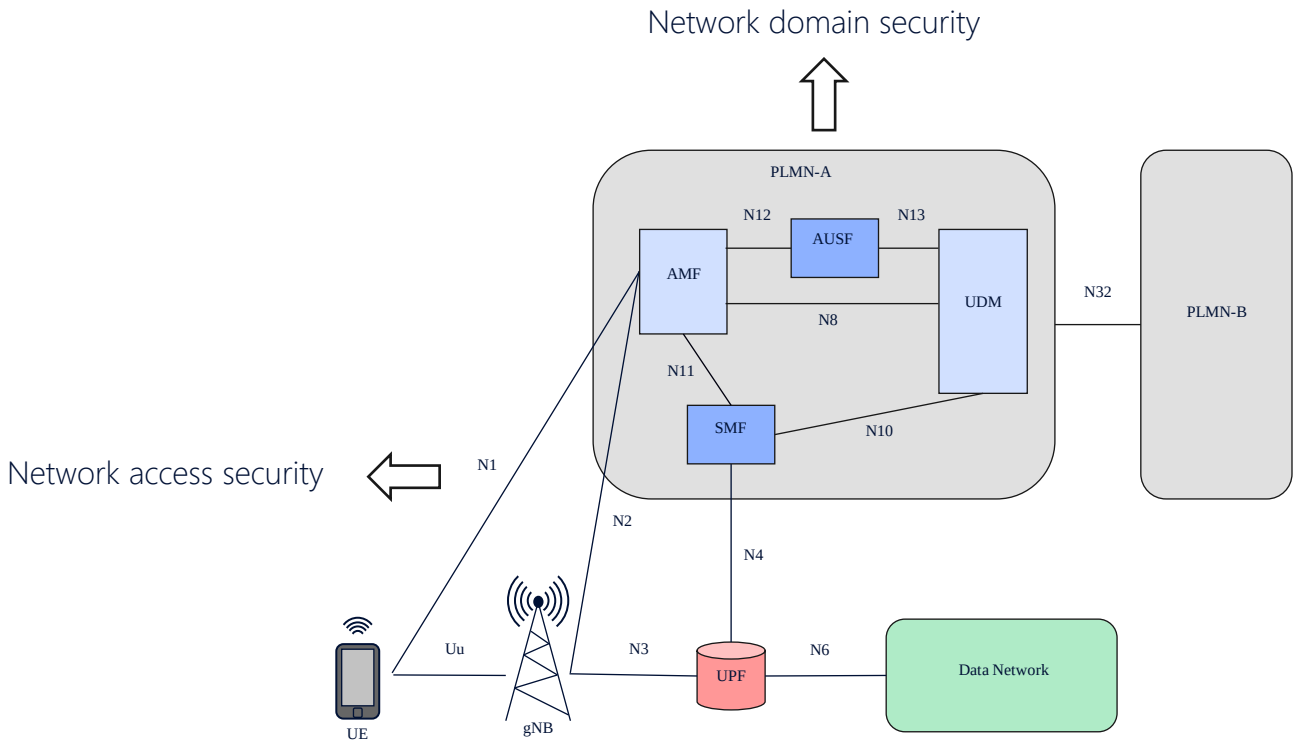
Sai Anirudh M
Aritra Banerjee
Prof. Dr.-Ing. Wolfgang Kellerer

VDE ITG Fachtagung Mobilkommunikation 2025

Overview

1. 5G Security Architecture
2. Quantum Attacks
3. Standardization of PQC
4. Integration of PQC

5G Security: Architecture overview



5G Security: Procedures

Network Access Security

Primary authentication:

- When a UE tries to connect to the network, a subscription concealed identifier (SUCI), an encryption of the subscription permanent identifier (SUPI) is sent to the network for authentication
- Elliptic curve integrated encryption scheme (ECIES) is used based on a shared key derived from the public key of the home network and an ephemeral private key generated at the UE

Authentication and key agreement:

- 5G-AKA/EAP-AKA' are procedures used to first authenticate (based on SUCI or GUTI – globally unique temporary identifier) and agree on a set of keys
- Authentication is based on a pre-shared symmetric key K that is unique to each subscriber
- K is also used to derive subsequent keys using a HMAC-based key derivation function (HKDF), for encryption and integrity protection of (a) NAS signalling, (b) RRC signalling and (c) user plane traffic

5G Security: Procedures

Network Domain Security

Service-based interfaces:

- Interfaces in the 5G service-based architecture between core network functions (within PLMN)
- These functions act as service producers/consumers and are connected through standardized APIs over HTTP/2 or HTTP/3 transport.
- OAuth 2.0 is used to generate and assign tokens needed to access services of other functions. Network repository function (NRF) acts as the authorization server
- TLS 1.2 or 1.3 is used to provide flexible point-to-point security between NFs, ensuring encryption, authentication and integrity protection.

Perimeter:

- NFs between two PLMNs communicate over N32 through secure edge protection proxies (SEPPs)
- Uses TLS (direct IP-based roaming) or PRINS (roaming with intermediaries) for end-to-end security
- JWE for field level encryption of sensitive data, JWS for integrity protection

5G Security: Procedures

Network Domain Security

Non-service based interfaces:

- Interfaces with direct control plane (CP) or user plane (UP) signaling, more tightly coupled to specific network elements. Follow traditional IP based protocols
- Provided with NDS/IP security, generally with the following supported protocols
  - Mandatory IKEv2 certificate-based authentication
  - IPsec ESP (encapsulating security payload) – provides topology hiding
  - DTLS for mutual authentication, integrity, confidentiality – message size limitations
- Includes following key interfaces:
  - N2 (gNB to AMF)
  - N3 (gNB to UPF)
  - Xn (between gNBs), F1 (gNB-CU to gNB-DU), E1 (gNB-CU-CP to gNB-CU-UP)

Summary of security procedures with cryptography used

Network procedure	Security functions				Cryptography
	Auth	Int	Key	Enc	
SUPI protection			x	x	ECDH
AKA	x	x	x		PSK, SHA
NAS		x		x	AES, ZUC, SNOW V
AS		x		x	AES, ZUC, SNOW V

Network protocol	Security functions				Cryptography
	Auth	Int	Key	Enc	
TLS/DTLS 1.3	x	x	x	x	ECDH RSA/EC/Ed DSA
OAuth2.0	x				RSA/EC DSA
IKEv2	x	x	x		DH/ECDH RSA/EC/Ed DSA
IPsec		x		x	AES
PRINS-JWE			x	x	RSA, ECDH or PSK
PRINS-JWS	x	x			RSA, ECDSA

Auth – authentication

Int – integrity protection

Key – key derivation/agreement/exchange

Enc – encryption

The Quantum Threat

Quantum algorithms

Grover's algorithm:

- Quantum search algorithm that achieves a complexity of $O(\sqrt{N})$ in a search space with N values
- Impact on security: at worst, symmetric key sizes need to be doubled to retain same security level

Shor's algorithm:

- Quantum algorithm that solves, in polynomial time,
  - The Integer Factorization Problem (IFP) on which RSA is based
  - The Discrete Logarithm Problem (DLP) on which DH is based
  - The Elliptic Curve DLP (ECDLP) on which ECC is based
- Impact on security: public key cryptography can be attacked in reasonable frames, provided one has access to a Cryptographically Relevant Quantum Computer (CRQC)
- Key sizes would have to be unmanageably large (several gigabytes) to maintain same security level

The Quantum Threat

Resource estimates and timelines

- Landmark work [1] showed how RSA-2048 can be broken in 8 hours using 20 million noisy qubits

Cipher	Logical qubits required	Cost (megaqubitdays)
RSA-2048	6190	1.17
RSA-3072	9288	4.03
RSA-7680	23239	86.5
ECDSA-256	2619	7.43
ECDSA-384	3901	10.0
ECDSA-512	5273	15.6

- Today's biggest known quantum computer, IBM's Condor, has 1121-qubits
- Most experts believe CRQCs could emerge anywhere between 5 and 30 years from now
- Store Now, Decrypt Later (SNDL) threat already makes current encrypted data vulnerable, and necessitates an urgent migration to quantum-safe solutions

[1] C. Gidney and M. Ekerå, "How to factor 2048-bit RSA integers in 8 hours using 20 million noisy qubits", Quantum, Apr. 2021

NIST PQC algorithms

National Institute of Standards and Technology (NIST) of USA is at the forefront of standardizing post-quantum cryptography through a public competition. 82 candidates were submitted and after a decade of evaluation on the basis of 1) security, 2) performance and 3) algorithm characteristics, five algorithms were recommended for standardization:

- ML-KEM - Module lattice-based key encapsulation method
- HQC-KEM – Hamming quasi-cyclic key encapsulation method

- ML-DSA - Module lattice-based digital signature algorithm
- SLH-DSA - Stateless hash-based digital signature algorithm
- FN-DSA - Falcon digital signature algorithm

NIST PQC algorithms

Algorithm	Public key size	Ciphertext size
ML-KEM-512	800	768
HQC-128	2249	4497
SLH-DSA-128f	32	17088
FN-DSA-512	512	512
ML-DSA-44	1312	2420

Key and ciphertext sizes (in bytes) for various cryptographic algorithms for 128-bit security level

Existing research on PQC integration

SUCI computation/5G AKA

Several implementations proposed and evaluated in network context

- KEMSUCI [2] – replaces ECIES with any KEM, evaluated for NIST PQC finalists. Faster execution speed, minimal communication overhead w.r.t legacy 5G
- Hybrid SUCI encryption with Perfect Forward Secrecy [3]- emphasizes forward secrecy through multiple ephemeral schemes, with in-depth evaluation of computation time
- Formal verification post-quantum 5G AKA with Kyber (security analysis) [4]

Takeaway: Asymmetric operations take up a large portion of computational resources for security. (Pure) PQ algorithms offer significant speed-up over ECIES in key generation. Hybrid methods have comparable (slightly higher) time requirements

[2] V. Ulitzsch et al., "A post-quantum secure subscription concealed identifier for 6G"

[3] R. Vuppala et al., "Post-Quantum Secure Hybrid Methods for UE Primary Authentication in 6G with Forward Secrecy"

[4] M. Damir et al., "A Beyond-5G Authentication and Key Agreement Protocol"

Existing research on PQC integration

TLS 1.3

Extensive work in integration of PQC, resulting in these SoTA (?) implementations

1. KEMTLS [5] – uses PQ (/hybrid) KEMs to provide authenticated key exchange without use of DSAs
2. PQTLS [6] – uses hybrid key exchange with pure PQ authentication

Security analysis – only specific implementations of post-quantum TLS have been formally verified

Impact of PQC on latency is not large due to efficient computation of keys, though existing network parameters (key sizes, congestion windows) could limit performance

Key agreement	Authentication	Handshakes per second	CPU cost (ms)	
			Server	Client
X25519	RSA-2048	402	3.1	1.9
ML-KEM-512	ML-DSA-44	473	1.6	1,7
ML-KEM-512	SLH-DSA-128	59	24.1	5.1
P256+ML-KEM-512	ML-DSA-44	382	2.0	2.2

[5] P. Schwabe et al., "Post-quantum TLS without handshake signatures"

[6] C. Paquin et al., "Benchmarking post-quantum cryptography in TLS"

Existing research on PQC integration

Miscellaneous

- Combined Elliptic-Curve and Post-Quantum 2 (CECPQ2): developed by Google to evaluate performance of PQ key exchange on actual users' devices (supported on Chrome for TLS 1.3 connections)
- iMessage with PQ3: end-to-end encryption with PQC key establishment and ongoing PQC-rekeying (level 3 security)
- PQXDH – Post quantum extended Diffie-Hellman, an open-source key exchange protocol developed and used for the Signal messaging app.

Conclusions

Challenges in the context of next-generation mobile networks

Trust

decades of security analysis

Interoperability

object sizes, computation

Network constraints

configurable parameters

Resource constraints

power, memory, bandwidth

Thank you!

Questions?