# Assessing the Latency of Network Layer Security in 5G Networks

Sotiris Michaelides
RWTH Aachen University
Aachen, Germany
michaelides@spice.rwth-aachen.de

Jonathan Mucke
RWTH Aachen University
Aachen, Germany
jonathan.mucke@rwth-aachen.de

Martin Henze*
RWTH Aachen University
Aachen, Germany
henze@spice.rwth-aachen.de

## Abstract

In contrast to its predecessors, 5G supports a wide range of commercial, industrial, and critical infrastructure scenarios. One key feature of 5G, ultra-reliable low latency communication, is particularly appealing to such scenarios for its real-time capabilities. However, 5G's enhanced security, mostly realized through *optional* security controls, imposes additional overhead on the network performance, potentially hindering its real-time capabilities. To better assess this impact and guide operators in choosing between different options, we measure the latency overhead of IPsec when applied over the N3 and the service-based interfaces to protect user and control plane data, respectively. Furthermore, we evaluate whether WireGuard constitutes an alternative to reduce this overhead. Our findings show that IPsec, if configured correctly, has minimal latency impact and thus is a prime candidate to secure real-time critical scenarios.

## CCS Concepts

• **Security and privacy** → **Security protocols**; **Mobile and wireless security**; • **Networks** → *Mobile networks.*

## Keywords

5G; Network Layer Security; Latency; IPsec; WireGuard; TLS

**ACM Reference Format:**
Sotiris Michaelides, Jonathan Mucke, and Martin Henze. 2025. Assessing the Latency of Network Layer Security in 5G Networks. In *18th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec 2025), June 30-July 3, 2025, Arlington, VA, USA*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3734477.3734722

## 1 Introduction

Up until the Fourth Generation (4G) of mobile networks, the primary focus was on enhancing mobile broadband for commercial use, prioritizing bandwidth [14]. However, the advent of Fifth Generation (5G) networks marked a significant shift towards addressing both commercial *and* industrial deployments. This transition targeted industrial applications and critical infrastructure, meeting the growing demand for robust, low-latency communication [1, 15].

To this end, 5G supports Ultra-Reliable Low Latency Communication (URLLC), crucial for real-time applications, making it an ideal backbone for critical infrastructure such as industrial control systems and healthcare. In these sectors, even minor latency shifts can disrupt production lines or create safety hazards in real-time data transmission scenarios, such as remote surgeries.

At the same time, 5G's enhanced security over its predecessors resonates extremely well with the strict security requirements of such sectors faced by growing security threats [20]. Concretely, 5G implements advanced security protocols and controls addressing previous vulnerabilities, such as the lack of User Plane (UP) integrity protection in 4G [18]. These controls and protocols are essential for ensuring security guarantees, such as protection against tampering and eavesdropping attacks. However, the utilization of many of these is optional and left to the discretion of network operators.

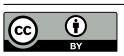
A notable example is the use of network layer security to ensure these essential security guarantees. The 3GPP 5G specification suggests implementing IPsec on most 5G interfaces, including those within the 5G Core (5GC), which manages critical functions such as authentication, billing, and security. Without proper protection, these interfaces remain vulnerable to data tampering, privacy breaches, or service outages. While the specification mandates their protection, utilizing security protocols can be avoided if alternative measures, such as physical security, are used [6]. Furthermore, even if network operators principally decide to apply such optional security controls, they have to choose between different configuration options, e.g., ciphers, authentication options, and operation modes.

This decision is further complicated as, besides tremendous security benefits, these optional controls increase latency, contradicting efforts to achieve reliable communication with minimal latency. Still, with the rise of distributed and cloud-based 5GC deployments [10], where physical protection becomes impractical, network operators must consider enabling these controls to improve security. Consequently, they require a profound understanding of how these controls impact 5G's ability to support time-critical applications and which configurations are most latency-friendly.

This paper provides this much-needed understanding by empirically evaluating the latency overhead of network layer security in 5G. Specifically, we analyze the latency impact of IPsec in various configurations on both the User Plane (UP) and the Control Plane (CP) of the 5GC in a simulated setup. Furthermore, we assess the overhead of WireGuard, a modern security protocol often proposed as a promising alternative [12, 24]. In detail, our contributions are:

(1) By orchestrating and extending existing open-source components, we facilitate the first open 5G testbed that implements IPsec and WireGuard [16].
(2) By measuring the overhead of IPsec for UP data transmission and CP communication within the 5GC, we identify latency-friendly IPsec configurations that add an overhead of just 55 µs for UP and between 300 µs-600 µs for CP.
(3) By assessing WireGuard, we find that while it shows no advantage in w.r.t latency, it is still a fast and more resource-friendly alternative compared to IPsec.

*Also with Fraunhofer FKIE.




*WiSec 2025, Arlington, VA, USA*

ACM ISBN 979-8-4007-1530-3/2025/06
https://doi.org/10.1145/3734477.3734722

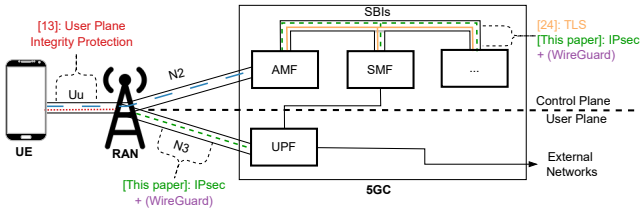



**Figure 1: In a 5G network, separated into control and user plane, various security controls apply to different planes and interfaces: Control Plane Security, User Plane Security, IPsec, and TLS. In this work, we study the latency impact of IPsec and WireGuard to secure the UP of 5G and the CP of the 5GC.**

## 2 Background on 5G Systems and Their Security

A 5G system comprises the User Equipment (UE), Radio Access Network (RAN), and 5GC (cf. Figure 1). The UE is the end-user device equipped with authentication credentials while the RAN manages radio resources to provide wireless connectivity to the UE via the Uu interface. The 5GC, linked to the RAN, interconnects Network Functions (NFs) controlling network operations and managing connections [3, §4.2]. A 5G system is logically divided into two planes: the Control Plane (CP) handling control functions such as authentication and session management, and the User Plane (UP) transmitting user data [2, §4.3]. This separation is most evident in the 5GC, where control tasks are handled by CP NFs such as the Authentication and Mobility Function (AMF), Session Mangement Function (SMF), and Policy Control Function (PCF), while the User Plane Function (UPF) in the UP routes user data to external networks. CP NFs communicate with each other over Service Based Interfaces (SBIs), while CP and UP data are transmitted from the RAN to the AMF and UPF via the N2 and N3 interfaces, respectively.

### 2.1 Security Controls in 5G

To secure data over 5G, 3GPP defines several security controls, which inevitably affect network performance, especially latency.

**CP and UP Security** The specification mandates security controls for both planes, comprising three encryption and three integrity protection schemes based on AES, SNOW, and ZUC [13]. However, operators are only required to enable integrity protection for CP data, while the other security controls remain optional. While both planes use the same schemes, their termination points differ: CP security is established between the UE and AMF, whereas UP security terminates at the RAN. Thus, control data from the UE is always integrity-protected over Uu and N2. On the other hand, even with UP security enabled, user data is only protected over Uu, remaining unprotected over N3 unless higher-layer end-to-end security is applied. This distinction is critical for low-latency communication, as securing user data across 5G requires *both* UP and N3 security, introducing two latency overheads (cf. Figure 1).

**N3 Security.** Securing UP data over the N3 interface (i.e., between RAN and UPF), requires the utilization of IPsec, configured with specific parameters such as cryptographic algorithms and corresponding key lengths, as mandated by 3GPP. However, operators may optionally choose not to implement these security measures if the RAN is placed in a "*secure environment*" [6, §9.3].

**SBI Security.** Similar to the N3 interface, SBIs within the 5GC must be secured unless deemed "*trusted (e.g., physically protected)*" [6, §13.1]. To secure SBIs, the specification mandates the support of Transport Layer Security (TLS), but also notes NDS/IP (3GPP's specification for IPsec configuration) as an alternative option.

### 2.2 Security Protocols

**5G Standardized Protocols.** *IPsec* is a suite of protocols that provides security between two hosts at the network layer. Initially, the *Internet Key Exchange Version 2 (IKEv2)* protocol handles mutual authentication and establishes security associations by exchanging certificates, cryptographic challenges and key material, usually in two round trips. The *Encapsulation Security Payload (ESP)* protocol then establishes the secure tunnel using the parameters negotiated in IKEv2. ESP can operate in transport mode (encrypting the payload) or tunnel mode (encrypting the entire packet), with tunnel providing more security by hiding topology information [6, §9.2]. While both protocols introduce additional overhead, IKEv2 has a greater impact, especially in scenarios that require multiple authentication events, as it is often the case within the 5GC.

*TLS* in contrast realizes end-to-end security at the transport layer [24]. Similar to IPsec, it uses an initial handshake to exchange certificates and key material for authentication. Afterwards, a secure channel is established. The current standards, TLS 1.2 and TLS 1.3, differ in latency impact; TLS 1.3 requires only one round trip for the handshake compared to two in TLS 1.2.

Both IPsec and TLS are established security protocols that offer flexibility by supporting various cryptographic algorithms and key lengths as well as authentication using either Pre-Shared Keys (PSK) or certificates. In this work, we identify 12 mandatory configurations for IPsec, while related work identifies 14 for TLS [24].

**WireGuard as Alternative.** *WireGuard* is a relatively new protocol (proposed in 2017), that challenges both IPsec and TLS w.r.t. latency and bandwidth [8]. Although not part of the 5G standards, it is discussed as an alternative to improve latency in 5G [12, 24]. Similar to IPsec, it realizes security at the network layer. It utilizes a simple, small-size, one-round-trip handshake for authentication. Then, a modern Authenticated Encryption with Associated Data (AEAD) algorithm, CHACHA20_POLY1305, is used to establish the secure channel. AEAD algorithms are faster than traditional schemes as they perform encryption and authentication simultaneously. In contrast to IPsec and TLS, WireGuard intentionally only supports one configuration with PSKs and 256-bit keys.

### 2.3 Related Work

While 5G is well-established, the impact of its (optional) security controls, especially w.r.t latency, has received limited attention. Heijligenberg et al.[13] evaluate the impact of UP integrity protection (red in Figure 1), showing noticeable effects across all schemes, even in low-latency configurations. Zeidler et al.[24] assess the impact of TLS in the 5GC (orange in Figure 1), finding minimal impact on UE registration and PDU establishment in a running network but significant overhead on a freshly started one. Haga et al. evaluate WireGuard and OpenVPN (TLS) for slice isolation, finding that WireGuard outperformed OpenVPN across all metrics[12].

Without focusing on 5G, several works compare the performance of security protocols. Kotuliak et al. compare IPsec and TLS for interconnected IP multimedia subsystems, finding comparable performance, with a slight advantage for IPsec. Similarly, Dekker et al. [7] compare WireGuard, Strongswan (IPsec), and OpenVPN in 1 Gbit/s environments, showing Strongswan ciphers outperforming the others in terms of latency. In contrast, Donenfeld [8] reports that both WireGuard and IPsec add less latency than TLS, with WireGuard performing best. However, the WireGuard website notes that these results are *"old, crusty, and not super well conducted"* [9].

**Novelty of Our Work:** As highlighted in Figure 1, our work significantly extends prior research in the following manner. First, while previous work on UP data protection [13] focused on the UE-RAN link, we expand this by evaluating 3GPP standardized security controls over N3, offering a comprehensive assessment of UP data protection across 5G. Second, we complete the evaluation of 3GPP-standardized security protocols within the 5GC. While previous research has analyzed TLS [24], we extend this by evaluating IPsec and reproducing some of the best-performing TLS configurations for comparison. Finally, we assess WireGuard as a potential alternative for protection over N3 and within the 5GC, providing an up-to-date comparison of the three protocols.

## 3 Methodology

To provide a profound understanding of the latency impact of 5G's network layer security, we perform latency measurements over the N3 interface during UP data transmission and over the SBI during the UE attachment to the network. In the following, we first discuss our methodology for extracting mandatory IPsec configurations from 3GPP standards (§3.1), before we describe the containerized deployment used to conduct our measurements (§3.2).

### 3.1 Selection of IPsec Configurations

IPsec can be configured in various ways, involving different modes of operation, ciphers, and authentication algorithms. To ensure baseline compatibility among 5G components implementing IPsec, we adhere to the 3GPP specification TS 33.501, which outlines the security architecture and procedures for 5G systems. According to TS 33.501 [6, §9.3,13.1.0], IPsec configurations must comply with TS 33.210[4] and TS 33.310 [5], applicable to both N3 and SBIs. For ESP configuration, the standards refer to RFC 8221 [23].

From these documents, we identify and test the mandatory-to-support configurations for both IKEv2 and ESP to ensure alignment with compatibility and security standards in 5G deployments. We exclude mandatory-to-support configurations explicitly classified as not recommended, such as RSA signatures with PKCS#1 v1.5 padding—expected to be prohibited by 2030 [5, §6.1]—and the utilization of the Authentication Header, which is discouraged as ESP can provide encryption and authentication more efficiently [23, §4].

We summarize the IPsec configurations used in our study in Table 1. For IKEv2, we identify one mandatory set of configurations that can be used with either certificates or PSKs for authentication, resulting in two distinct test cases. For the ESP configuration, either an AEAD algorithm may be used, or a combination of separate algorithms for encryption and integrity (cf. dashed line in Table 1). While integrity protection can be applied without encryption, the reverse (encryption without integrity protection) is prohibited by RFC 8221 [23, §4]. Based on this, we identify a total of six distinct test cases for ESP. Thus, the combination of the two protocols results in twelve (2 IKEv2 × 6 ESP) mandatory-to-support IPsec configurations. Finally, we configure IPsec in tunnel mode, as this setup provides an additional layer of security, specifically topology hiding by also encrypting the IP headers [6, §9.2].

**Table 1: Our experiments cover all IPsec configurations which are defined as mandatory to support in 5G.**

| | Type | Configuration | Document |
|---|---|---|---|
| IKE | Encryption | AES128-GCM, ICV128 | TS 33.210 §5.4.2 |
| | PRF | HMAC-SHA256 | TS 33.210 §5.4.2 |
| | Integrity | HMAC-SHA256-128 | TS 33.210 §5.4.2 |
| | Key exchange | DH Group 19 | TS 33.210 §5.4.2 |
| | Authentication | ECDSA-SHA256$^{\alpha}$ | TS 33.310 §6.2.1 |
| | | SKMIC$^{\beta}$ | RFC 7296 §3.8 |
| ESP | AEAD | AES128-GCM, ICV128 | RFC 8221 §5 |
| | | AES256-GCM, ICV128 | RFC 8221 §5 |
| | Encryption | AES128-CBC | RFC 8221 §5 |
| | | AES256-CBC | RFC 8221 §5 |
| | | NULL | RFC 8221 §5 |
| | Integrity | HMAC-SHA256-128 | RFC 8221 §6 |
| | | GMAC-AES128$^{\gamma}$ | TS 33.210 §5.3.4 |

$^{\alpha}$ Certificate Authentication $^{\beta}$ PSK Authentication $^{\gamma}$ Only with *NULL* encryption

Besides IPsec, we note that WireGuard requires no configuration, as it intentionally supports only one configuration (cf. §2.2). For our reproduction of TLS measurements over the SBI, we use TLS 1.2 and 1.3 with AES-GCM at 128-bit and 256-bit key lengths, as these have been shown to perform best w.r.t. latency [24].

### 3.2 Experimental Setup

To measure the latency impact of these security protocol configurations, we set up a testbed based on a containerized deployment of open-source components, which we make publicly available [16].

**Testbed:** We rely on open-source 5G components widely used in academia: Open5GS [17] for the 5GC and UERANSIM for the RAN and UE simulation [11]. Each NF as well as the RAN and UEs are deployed as separate Docker containers on the same physical host. By realizing a controlled environment, where all components run on a single physical host, we ensure accurate measurements by minimizing external factors that could affect latency.

We add IPsec-secured communication by utilizing strongSwan [21]. NFs communicating over the SBIs establish IPsec tunnels using pre-distributed certificates or PSKs. We also use IPsec to secure communication between the RAN and UPF over the N3 interface. In our deployment, strongSwan handles the traffic transparently, enabling its easy replacement with other IPsec solutions. Additionally, we modify Open5GS to fully support IPsec in trap mode (cf. §4.2), where the tunnel is established upon detecting traffic that matches the tunnel's policy. In this mode, packets sent before the tunnel is established are dropped, causing slower SBI connection setups due to retransmissions. To mitigate this, we modify the NFs to send a dummy packet and synchronously wait for the tunnel to be established before proceeding. To evaluate WireGuard, we use

the WireGuard implementation in Linux kernel version 6.5.0-1027-oem. For the replicated TLS measurements over the SBIs, we enable Open5GS's built-in TLS support in each NF's configuration file.

**Host Machine:** For our measurements, we deploy the containers on a computer running Ubuntu 22.04 LTS and equipped with an Intel Core i7-13700 (8 Performance and 8 Efficient cores, with a Performance-core base frequency of 2.10 GHz) as well as 64 GB of DDR5 RAM. We utilize AES acceleration through AES-NI provided by the CPU, as real-life 5G deployments are likely to operate on hardware that also supports AES acceleration.

## 4 Latency Overhead of Tunneling Protocols

We evaluate the latency overhead of security controls in 5G by measuring the time required for specific procedures compared to a baseline scenario (without security controls). We assess their impact on UP data transmission over the N3 interface (§4.1) and the CP data in the 5GC (§4.2). For each experiment we report the mean over multiple runs with 99% confidence intervals ensuring high accuracy in pinpointing the true average. Different configurations of IPsec (and TLS) are presented as *"Encryption"_"Integrity"*, or a single algorithm for AEAD schemes. We further investigate the CPU time of each protocol (§4.3) and validate latency results with real UEs (§4.4). Finally, we discuss our results with MNOs (§4.5).

### 4.1 User Data Transmission over N3

To evaluate the latency overhead of IPsec and WireGuard on the N3 interface, we measure the round-trip time (RTT) between the UE and the host machine. The RTT reflects twice the transmission latency for UP data traveling from the UE to its destination.

**Evaluation Method.** To generate UP traffic, we use `Ping` within the UE container. `Ping` measures the RTT of a single packet between two hosts using ICMP. To minimize external factors that could influence our measurements, we ping the host machine, ensuring that traffic remains confined to the physical host. As the RAN and the UPF establish and maintain the tunnel upon exchanging data for the first time—shortly after the first user packet—the impact of tunnel establishment (i.e., IKEv2) becomes negligible. Therefore, we only consider the six ESP configurations, and the single configuration of WireGuard, and start measuring after the tunnels are established. We perform 20,000 repetitions for each configuration, divided into 10 sets of 2,000 repetitions. Additionally, we test different payload sizes: (i) 64 bytes (the default `Ping` packet size in Ubuntu) and (ii) 1,024 bytes (to amplify the impact of cryptographic operations).

**Results.** Our results in Figure 2 demonstrate a clear difference between the baseline scenario, IPsec, and WireGuard. All six IPsec ESP configurations show comparable performance, with an average latency overhead of 55 µs (for 64-byte payload) and a standard deviation of ±7 µs, corresponding to a ~5% increase. In contrast, WireGuard introduces a larger overhead of 260 µs, making it 17.2% slower than the fastest IPsec configuration. For both, the overhead remains consistent as payload size increases. While the difference between them is noticeable, both protocols remain fast in absolute terms, with IPsec being better suited for time-critical applications.

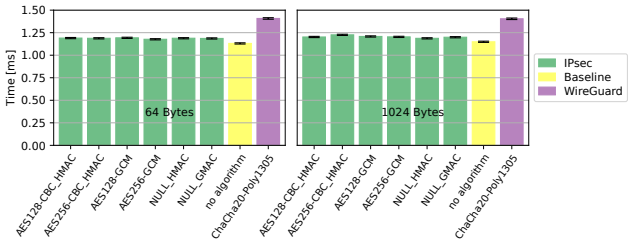



**Figure 2: The average RTT for user data transmission on the N3 interface behaves similarly for different payload sizes, with IPsec performing comparable to the insecure baseline while WireGuard shows a slightly worse performance.**

### 4.2 Control Communication within the 5GC

To evaluate the impact of security protocols on the CP of 5GC, we measure UE attachment time—from registration to transmission readiness. During this process, NFs communicate with each other to fulfill tasks (e.g., authentication). By analyzing traffic in the 5GC, we identify 17 communication channels, including 14 previously found by Zeidler et al. [24] and three new ones between the SMF/Network Repository Function (NRF)/PCF and the Binding Support Function (BSF), which was recently introduced in Open5GS.

**Evaluation Method.** Unlike measurements over N3, execution time in this scenario is primarily influenced by the number of connections that must be established within the 5GC CP. We analyze two scenarios: the cold scenario, where security associations between NFs are not established, and the warm scenario, where security associations are already in place, limiting overhead to encryption and integrity protection. These represent worst-case (cold) and best-case (warm) latency conditions. While cold scenarios may seem unintuitive in (commercial) real-world deployments, where 5G systems are (almost) always operational, operators may deploy new NFs for scalability, recovery, and load balancing, requiring the re-establishment of many security associations [3, §5.21.3.1], leading to at least a partial cold scenario. In the cold scenario, initial security setup adds significant overhead. However, of the 17 total channels needed for UE attachment, 8 are automatically established without traffic from the UE, as each NF contacts and registers with the NRF upon startup for NF discovery. Therefore, cold scenario measurements account for the time to establish 9 secure channels in addition to encryption and integrity protection.

We conduct measurements by booting up the 5GC and RAN, then sequentially deploying two UEs. The first UE represents the cold scenario, with no pre-established secure channels, while the second UE represents the warm scenario, leveraging channels created by the first. We capture traffic using Tshark and measure the time from the UE's registration request to the PDU session establishment message sent by the SMF, signaling readiness for data transmission. We measure all 12 IPsec configurations identified in §3.1 in tunnel mode and the single WireGuard configuration. We also reproduce measurements for TLS 1.2 and 1.3 using AES-GCM with 128 and 256-bit keys, which were shown to offer the best latency [24]. Finally, we evaluate two IPsec tunnel establishment modes: *trap*, where tunnels are created when captured traffic matches the tunnel's policy, and *start*, where tunnels are established immediately at daemon startup.

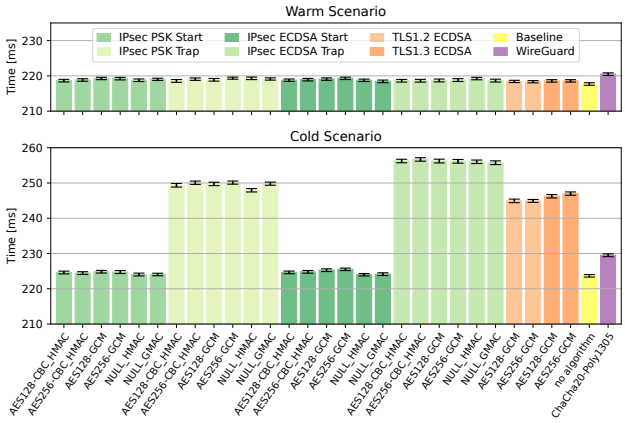



**Figure 3: The average UE attachment time in *warm* scenarios is similar across configurations and protocols, except for WireGuard which is slightly slower. In *cold* scenarios, WireGuard and IPsec in start mode demonstrate their advantages.**

**Results.** As expected, the results in Figure 3 show significant differences between the two scenarios. In the warm scenario, grouping the configurations of each protocol reveals that TLS is, on average, slightly faster than IPsec, imposing 0.8 ms of overhead compared to 1.2 ms for IPsec, corresponding to ~0.4% and 0.55% increases, respectively. While TLS is faster on average, multiple IPsec configurations are on par with TLS. In contrast, WireGuard introduces nearly twice the overhead, making it 1% slower than the best performing IPsec configuration. Overall, all protocols perform very well in this scenario, as the measurements are unaffected by authentication and key exchange, with IPsec and TLS showing a slight edge over WireGuard. In the cold scenario, where authentication and key exchange delay communication, the results are more scattered. A comparison of IPsec authentication methods in *trap mode* shows that PSK, on average, is faster than certificate-based authentication, due to bypassing certificate verification. However, even the fastest IPsec PSK configuration is slower than the worst-performing TLS configuration, adding approximately 1 ms more overhead (23 ms compared to 24 ms). WireGuard outperforms both due to its faster handshake, with just 6 ms (~2.5%) overhead. However, in *start mode*, where tunnels are established at network startup, IPsec demonstrates a clear advantage over TLS and WireGuard, with configurations adding on average ±1 ms (~0.4%). This makes IPsec in start mode, on average, 22 ms (~9%) faster than TLS and 5 ms (~2.15%) faster than WireGuard. Across both scenarios, IPsec is the only protocol with latency overhead below 1 ms—approximately 600 µs in the warm scenario and 300 µs in the cold scenario in the fastest configuration (*ECDSA with NULL_GMAC*). On the other hand, while WireGuard is slightly slower than IPsec and TLS in terms of encryption and integrity protection, it remains extremely fast, especially in cold scenarios. We further discuss the applicability of the protocols in §6.

## 4.3 Resource Consumption and Scalability

Complementing our focus on latency, we also evaluate resource utilization by measuring the CPU time of each protocol, as processes

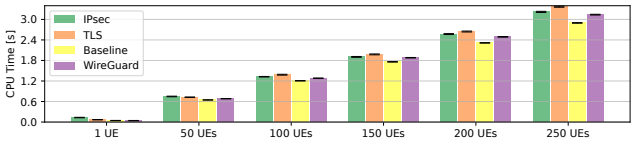



**Figure 4: The average CPU time for the successful registration of increasing UE numbers reveals scales linearly across security protocols, with a slight advantage for WireGuard.**

with lower CPU time scale better and are typically more resource-efficient. To assess scalability, we measure CPU time while scaling the number of UEs, generating a significant amount of traffic within the 5GC CP. Our measurements track CPU time from boot-up to attachment completion in warm scenarios, repeated 50 times and averaged with 99% confidence intervals (Fig. 4). Since TLS and IPsec show no measurable difference, we focus on *IPsec with ECDSA and AES-GCM-256 in tunnel mode* and *TLSv1.3 with ECDSA and AES-GCM-256*. At larger scales, we expect AEAD schemes (i.e., AES-GCM) to perform better due to their parallelizability (cf. §2.2) and PSK configurations to offer advantages due to the absence of resource-intensive public-key cryptography. Our measurements show that all protocols scale linearly, with WireGuard requiring the least resources. We expect the gap to widen further in devices without AES acceleration and at larger scales with more UEs.

## 4.4 Validation with Real UEs

Our measurements are designed to ensure statistical soundness, requiring many samples to minimize the effect of external factors. To achieve this efficiently, we use simulated UEs, though this may reduce real-world applicability. To validate our findings, we extend our testbed to support SRSRAN [19], which we use with a USRP X310, a Samsung Galaxy A14 5G, and a sysmocom ISIM-SJA5. With this more realistic setup, we reproduce selected measurements over N3 and the 5GC, specifically using *WireGuard* and *IPsec in start mode with ECDSA and AES256_GCM*. Across all measurements, latency and jitter increase notably, resulting in wider confidence intervals. Specifically, for 5GC measurements, this prevents us from identifying significant differences between the three protocols without additional samples. In contrast, over the N3 interface, while IPsec and the baseline remain indistinguishable (~29 ms), WireGuard consistently shows slightly higher latency (~31 ms)—an increase of approximately 7%. These results support IPsec in start mode as a strong candidate for low-latency communication.

## 4.5 Operators' Perspective

To gain deeper insights into IPsec utilization, the accuracy and impact of our results for real-world 5G deployments, we discuss our findings with two European MNOs, A and B, who request anonymity. Operator A confirms using both IPsec and TLS in the 5GC for different use cases. While they do not provide exact figures, they prefer TLS in low-latency cases, as internal experiments showed TLS had a lower latency impact on connection establishment than IPsec. However, their comparison does not consider IPsec in start mode, which aligns with our findings, suggesting TLS has lower overhead than non-start mode IPsec configurations. Operator

A finds our results promising and plans to investigate IPsec in start mode further. On the other hand, no operator currently secures the N3 interface with IPsec. Operator B emphasizes the importance of high availability and low complexity in the RAN. Unlike 5GC NFs, which can be redeployed quickly in case of failure, the RAN lacks full virtualization and often requires on-site technicians. IPsec introduces challenges, including configuration complexity, certificate management, and reduced traffic visibility. Given the RAN's need for continuous operation to avoid legal consequences, e.g., state-imposed fines, the operator deems IPsec an unsuitable option. In conclusion, while operators are actively exploring security, they sometimes prioritize low complexity, especially where operational stability is critical—making our findings valuable in guiding them to prioritize or balance security, performance, and complexity.

## 5 Discussion on Security and Latency

From a security standpoint, both IPsec and WireGuard offer similar guarantees, each with its advantages and drawbacks. WireGuard's single configuration is resistant to misconfigurations and easy to deploy, but it can be limiting, such as when pre-shared keys are impractical. In contrast, IPsec supports a broader range of configurations, providing greater flexibility but also increasing the risk of misconfigurations, a common security concern. However, one configuration specifically, *the start mode*, makes IPsec the best configuration in terms of latency, even compared to TLS. Our measurements indicate that, in start mode, other configurations have minimal impact on latency. Thus, we recommend the most secure setup: *any 256-bit cipher with ECDSA*, as DH-Group 19 ensures perfect forward secrecy, and *tunnel mode*, which encrypts IP headers. However, at larger scales or in resource-constrained devices, *PSK configurations with AES-GCM* may offer better latency performance due to their lower resource consumption (cf. Section 4.3).

## 6 Applicability

Our results show that both WireGuard and IPsec can meet most application requirements for user data transmission (*over the N3 interface*). For time-critical applications, IPsec is the preferred option for UP data transmission due to its slightly better performance in latency. However, where simplicity is prioritized, such as in commercial RANs (cf. §4.5), WireGuard is a promising alternative.

*Within the 5GC*, both protocols perform very well, with a slight advantage for IPsec. In commercial deployments where NFs may be frequently deployed (cf. §4.2), IPsec offers faster deployment. However, when pre-establishing tunnels (i.e., start mode) is challenging or when simplicity is preferred, WireGuard is the better option. In static deployments (e.g., in-house industrial 5G) where new NFs are rarely deployed, both protocols are suitable with minimal impact.

Lastly, we expect WireGuard to outperform IPsec on devices without AES acceleration and in resource-constrained environments. Our measurements show that while WireGuard's latency performance is slightly behind hardware-accelerated IPsec with AES, it requires fewer resources. Its fully parallelizable software implementation of ChaCha20-Poly1305, makes it more resource-efficient across various hardware platforms. Additionally, its smaller handshake further reduces network resource consumption, which is particularly beneficial in environments like Narrowband IoT.

## 7 Conclusion & Future Work

Our work strives to provide insights into the impact of network-layer security on 5G. To this end, we built a testbed to measure the latency of IPsec and WireGuard within the 5GC CP and during UP data transmission over the N3 interface. Our results show that properly configured IPsec is the best-performing protocol, introducing latency overhead in the microseconds range. WireGuard, while slightly slower, remains a lightweight and efficient alternative.

Future work should examine the impact of security controls on internal RAN and emerging O-RAN interfaces. Optimized IPsec implementations—e.g., using packet slicing—must be evaluated, as they have shown potential for achieving sub-1 ms RTTs [22]. Additionally, assessing the impact of post-quantum cryptography on 5G and future 6G systems is essential. In hybrid scenarios, where authentication and key exchange are more demanding and occur twice, our findings indicate that IPsec in start mode may be the only viable option for maintaining ultra-low latency.

## Acknowledgements

Funded by the German Federal Office for Information Security (BSI) under project funding reference number 01MO24003B (CSII). The authors are responsible for the content of this publication.

## References

[1] 3GPP. 2024. TS 22.261 V18.15.0: Service Requirements for the 5G System (5GS).
[2] 3GPP. 2024. TS 23.214 V18.0.0: Architecture enhancements for control and user plane separation of EPC nodes.
[3] 3GPP. 2024. TS 23.501 V18.7.0: System Architecture for the 5G System (5GS).
[4] 3GPP. 2024. TS 33.210 V18.1.0: Network Domain Security (NDS); IP Network Layer Security.
[5] 3GPP. 2024. TS 33.310 V18.4.0: Network Domain Security (NDS); Authentication Framework (AF).
[6] 3GPP. 2024. TS 33.501 V18.7.0: Security Architecture and Procedures for 5G System.
[7] Erik Dekker and Patrick Spaans. 2020. Performance comparison of VPN implementations WireGuard, strongSwan, and OpenVPN in a 1 Gbit/s environment.
[8] Jason A. Donenfeld. 2017. WireGuard: Next Generation Kernel Network Tunnel. In *NDSS*.
[9] Jason A. Donenfeld. 2023. WireGuard Performance. https://www.wireguard.com/performance/. Accessed: 2024-11-10.
[10] GSMA. 2024. 5G Security Guide. https://www.gsma.com/solutions-and-impact/technologies/security/wp-content/uploads/2024/07/FS.40-v3.0-002-19-July.pdf
[11] Ali Gungr. 2024. UERANSIM: A 5G RAN and UE Simulator. https://github.com/aligungr/UERANSIM Accessed: 2024-09-30.
[12] Simen Haga et al. 2020. 5G network slice isolation with WireGuard and open source MANO: A VPNaaS proof-of-concept. In *NFV-SDN*.
[13] Thijs Heijligenberg et al. 2023. BigMac: Performance Overhead of User Plane Integrity Protection in 5G Networks. In *ACM WiSec*.
[14] Amit Kumar et al. 2012. 3GPP LTE: The Future of Mobile Broadband. *Wirel. Pers. Commun.* 62, 3, 16 pages.
[15] Sotiris Michaelides et al. 2025. Secure Integration of 5G in Industrial Networks: State of the Art, Challenges and Opportunities. *FGCS*.
[16] Sotiris Michaelides et al. 2025. 5G-Network-Layer-Security. https://github.com/RWTH-SPICe/5G-Network-Layer-Security.
[17] Open5GS. 2024. Open5GS. https://open5gs.org/ Accessed: 2024-09-30.
[18] David Rupprecht et al. 2019. Breaking LTE on Layer Two. In *IEEE SP*.
[19] srsRAN. 2025. Open-source software for 5G and LTE. https://www.srsran.com
[20] Ioannis Stellios et al. 2018. A Survey of IoT-Enabled Cyberattacks: Assessing Attack Paths to Critical Infrastructures and Services. *IEEE Commun. Surv. Tutor.*
[21] strongSwan. 2024. strongSwan: Open Source IPsec VPN Solution. https://strongswan.org/ Accessed: 2024-09-30.
[22] Qianran Wang et al. 2023. The Optimization of IPSec VPN in 5G Mobile Communication Network. In *ACM CNCIT*.
[23] P. Wouters et al. 2017. Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH). IETF RFC 8221.
[24] Oliver Zeidler et al. 2024. Performance Evaluation of Transport Layer Security in the 5G Core Control Plane. In *ACM WiSec*.