Invade the Walled Garden: Evaluating GTP Security in Cellular Networks

Yiming Zhang*, Tao Wan†‡, Yaru Yang*, Haixin Duan*§✉,
Yichen Wang*, Jianjun Chen*§, Zixiang Wei* and Xiang Li¶
* *Tsinghua University,* † *CableLabs,* ‡ *Carleton University,* § *Zhongguancun Laboratory,* ¶ *Nankai University*
zhangyiming@tsinghua.edu.cn, t.wan@cablelabs.com, yyr22@mails.tsinghua.edu.cn, duanhx@tsinghua.edu.cn,
wyc9333@gmail.com, jianjun@tsinghua.edu.cn, weizixiang0@outlook.com, lixiang@nankai.edu.cn

***Abstract*—Cellular backhaul and core networks have traditionally been considered as *Walled Garden*, with their security ensured by physical isolation. Therefore, prior security studies primarily focused on radio access networks with limited treatment of backhaul and core network interfaces. In this paper, we performed a security evaluation of real-world GPRS Tunnelling Protocol (GTP) deployments. GTP is the fundamental protocol for user traffic management between base stations and core networks (inside the *Walled Garden*) from 3G to 5G, thus often assumed inaccessible and non-exploitable from the Internet.**

However, our study reveals for the first time the troubling state of GTP access control in real-world deployments. Aided by a semi-automated tool, our measurements discovered around 749,000 valid GTP hosts accessible via the public Internet, spanning across 1,176 service providers in 162 countries. Our results demonstrate potential exposure of mobile core network infrastructures to external threats. We then evaluated the attack surface of exposed GTP infrastructures, and found out that as many as 38 types of GTP messages can be misused to launch various attacks such as denial-of-service and session hijacking. Our experiments using open source 4G and 5G projects in isolated lab environments further confirm the feasibility of those GTP-based attacks, including remote hijacking of user traffic sent through cellular core networks. In addition to threats against cellular networks and their subscribers, exposed GTP devices could also be weaponized to launch large-scale reflective denial-of-services (RDoS) attacks. We hope our findings will increase awareness of GTP vulnerabilities among operators and the security community, highlighting the urgent need to further strengthen security in cellular core networks.

1. Introduction

Global cellular networks have been under rapid development for many years. According to GSMA Mobile Economy Report [12], the total number of mobile subscribers worldwide has exceeded 5.2 billion in 2023. With its widespread applications, such as industrial Internet of Things (IoTs), emergency, and transportation, cellular networks have been deeply integrated into various aspects of our society and become an increasingly critical communication infrastructure.

To accommodate the rapid growth of mobile services, especially data services, the underlying architecture of cellular networks has been evolving, particularly the Core Network (CN) architecture. Acting as the brain and backbone of cellular networks, CN controls and connects base stations and user equipment to external networks such as the Internet. As depicted in Figure 1, the CN of 2G cellular networks was based on circuit-switched infrastructure, which is isolated from public networks. From 3G onwards, the CN transitioned to Internet Protocol (IP) based networks. Evolved Packet Core (EPC) of 4G was entirely built on packet-switched networks, and 5G CN further evolved into a service-based architecture. Notably, despite the constant changes in core network control protocols and architectures across generations, the key protocol for user plane data transport remains unchanged. More specifically, the GPRS Tunnelling Protocol (GTP) [21], [23], [24], which manages user plane data tunneling between base stations and the core network, as well as among core network components, has been retained since 3G.

Research gap. The rapid development of cellular networks has gathered attention from the security community. However, published research mainly focused on security issues over radio interfaces between User Equipment (UE) and Radio Access Networks (RANs) and between UE and CN, such as fake base stations [45], [57], [75], [85], man-in-the-middle attacks [42], [72], [73], user tracking [31], [40], [53], [54] and SMS spoofing [52], [79], [83]. Recently, Akon et al. [27] studied access control mechanisms in 5G Service-based Architecture. However, beyond billing frauds [36], [65], [66], there remains a gap in understanding practical threats against cellular CNs deployed in real world. Particularly, we are interested in security threats of GTP, which is the fundamental protocol for user plane tunneling in cellular networks but lacks inherent security protection.

Motivation. The lack of attention to core network security may be due to its reputation as an "isolated" system. Historically, the cellular core network had been widely regarded as the "Walled Garden" [26], which means physical isolation was sufficient to fend off external attacks. However, the evolution of packet-based core networks has weakened their physical isolation. As noted in 3GPP standard TS 33.210 [22] (IP Network Layer Security), the IP-enabled architecture may increase the opportunities for external attackers to access and compromise core networks via the public Internet. For GTP, the primary concern is whether

DOI 10.1109/SP61157.2025.00028

network elements could detect and block unauthenticated GTP messages (from external attackers). In certain scenarios such as roaming and outsourced back-haul networks, GTP interfaces (e.g., Serving Gateway) are required to handle messages from external networks. If access control is inadequately implemented, attackers could directly reach core network elements via GTP, posing significant security risks.

Research questions. Our research aims to fill the gap in cellular core network security by focusing on GTP security, particularly the risks associated with exposing core network GTP nodes to the public Internet. We attempt to answer the following research questions: *How effective is the current protection of GTP infrastructures (e.g., via access control)*, and *what security risks are posed by any such publicly accessible GTP infrastructure to cellular networks and the Internet?*

Our approach. To answer the first research question on GTP exposure on the Internet, we executed a large-scale scanning experiment to identify exposed GTP nodes of all protocol versions (GTPv1-C, GTPv2-C, GTP-U). Our scanning was assisted by a semi-automated tool we developed for GTP device identification. To answer the second question on potential security risks from GTP, we analyzed all GTP message types and found 6 categories of attacks arising from the lack of message authentication in GTP. Due to ethical concerns, we cannot perform such testing in operational networks. Instead, we used open-source 4G and 5G projects to validate the feasibility of those attacks.

Major findings. Our study uncovers concerning issues with GTP access control in real-world deployments. We identified a total of 749K valid GTP hosts accessible via the public Internet. Their distribution is extensive, covering 1,176 service providers from 162 countries. The majority of them (669K, 89.34%) are GTPv2-C hosts (used in 4G and 5G NSA), which corresponds to the current dominance deployments of 4G/5G [12]. We shared the findings with GSMA and nine major operators, all of which were surprised by the results, indicating a lack of awareness of potential GTP exposure in real deployments. While most operators are reluctant to discuss details of our findings due to sensitivity of core network security, three operators did provide feedback. One operator confirmed our discovered GTP nodes belong to their mobile networks and has fixed the issue. Another operator confirmed our discovered GTP nodes belong to their Wi-Fi access networks, and is investigating the mitigation. A third operator informed us that some of the exposed GTP nodes in their networks are IP routers with GTP ports open by default [1], but did not provide information about the rest of the exposed GTP nodes.

We then evaluate the security risks introduced by exposed GTP interfaces by thoroughly analyzing GTP messages for potential exploitation and attacks. Our analysis indicates that 38 types of GTP messages can be exploited to execute 6 classes of attacks (Table 5), including denial-of-service, session hijacking, user tracking, and among others. We have successfully validated most of those attacks using open-source projects (Open5GS [14] and free5GC [13]) except for user tracking (due to the lack of implementation

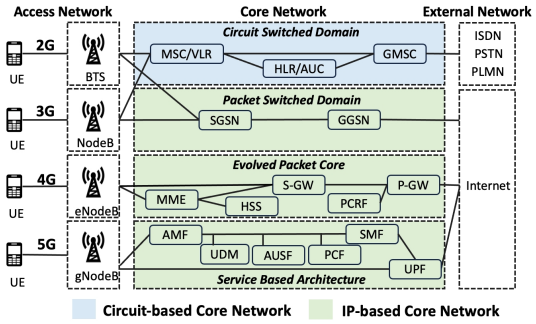


Figure 1: Cellular network architectures from 2G to 5G.

of vulnerable GTP message types in those projects). Our validation experiments (Section 5) demonstrate the feasibility and severe impact of GTP-based attacks against exposed interfaces. For example, the experiments show that external attackers have the potential ability to remotely disrupt a large number of UEs in a region, or even perform remote off-path hijacking of traffic sent to users via cellular core networks. Compared with air interface attacks, where attackers must be located close to victims, GTP-based attacks can be remote and are more powerful (e.g., by exploiting one network element to attack all UEs in its service area).

We also found an extended attack surface due to the nature of GTP and its implementation flaws. Attackers can carefully craft GTP requests to prompt much larger responses (e.g., containing numerous information elements (IEs)). Coupled with source address spoofing, GTP devices could be abused as a new type of amplifier for Reflective Denial-of-Service (RDoS) attacks. Our experiments with the basic GTP echo message resulted in a Bandwidth Amplification Factor (BAF) up to 170 in real-world implementations.

Finally, we analyzed the root causes and proposed mitigation suggestions. We hope this paper raises awareness of GTP security within the cellular community and encourages more operators to address the issues proactively.

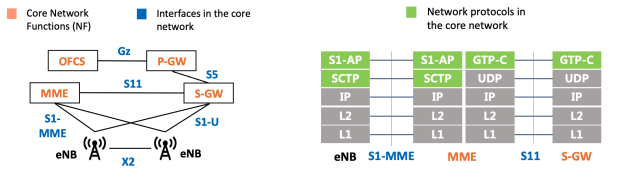
Paper organization. Section 2 provides the background on cellular networks and GTP. Section 3 outlines the methodology for identifying GTP hosts and ethical considerations. Section 4 analyzes the scale and characteristics of identified GTP hosts. Sections 5 and 6 elaborate on the security threats of GTP. Section 7 discusses the root causes and mitigation solutions. Section 8 discusses limitations and disclosure responses. Section 9 reviews related work, and then Section 10 concludes the paper.

2. Background

This section outlines the architecture and protocols of cellular core networks, with a particular emphasis on GPRS Tunnelling Protocol (GTP), the focus of this paper.

2.1. Cellular Core Network

Network architecture. As shown in Figure 1, a mobile network generally consists of User Equipment (UE), Radio



(a) Core Network Functions (b) Protocol stacks

Figure 2: Examples of network functions, interfaces and protocols in (4G) evolved packet core network architecture.

Access Network (RAN) and Core Network (CN). The UE connects to the CN via RAN, which typically includes base stations and base station controllers. The CN handles myriad important functions such as mobility management, session management and billing, and is the most critical and complex part of the cellular system. One key evolution of cellular networks over generations is the change of CN architecture, e.g., from circuit-switched to packet-switched network architecture. Circuit-switched networks allocate dedicated physical channels (circuit) between endpoints. The 2G core network is circuit-switched-based. 3G voice calls and text messages still use the circuit domain, while data services are based on packet-switched domains [25]. The Evolved Packet Core (EPC) of 4G is fully IP-based. 5G CN took a step further by adopting a service-based architecture, often deployed in virtualized cloud environment. Recently, leading operators like AT&T [28], Swisscom [37], and Dish [55] are considering deploying their 5G cores in the public cloud. We could see that *the architectural gap between the cellular core and the Internet is progressively closing*.

Core Network protocols. The Internet Engineering Task Force (IETF) typically standardizes Internet protocols through Request for Comments (RFCs). Historically, cellular core network protocols, particularly in the circuit-switched era, differed significantly from the Internet. From 4G, the core network has shifted to an IP packet-switched architecture. Consequently, IETF-defined IP-based protocols such as Session Initiation Protocol (SIP) [70] and Diameter [35] have been more widely adopted in core networks. Figure 2 presents a partial 4G core network schematic, and also an example protocol stack for the interface between the base station (eNB) and network entities including Mobility Management Entity (MME) and Serving Gateway (S-GW).

We can see unique features of current core network protocols: the network layer is IP-based, but the transport (e.g., SCTP, Stream Control Transmission Protocol [76]) and application layers (e.g., GTP) still use specialized cellular protocols. Despite constant changes to control plane protocols over generations, the data plane in the cellular core consistently relies on GTP to meet its specific requirements (e.g., mobility). *The pivotal role of GTP in core networks is the primary motivation of this paper.*

2.2. GPRS Tunnelling Protocol

As shown in Figure 3, the core idea of GTP is to enable mobility through tunnel creation and management, thus

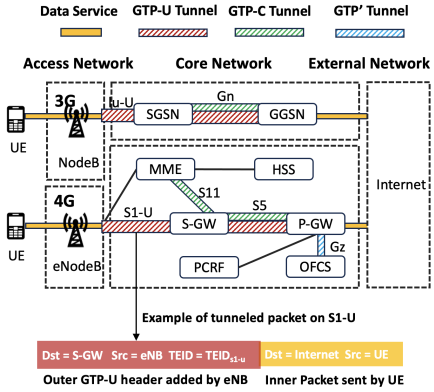


Figure 3: GTP tunnels in 3G and 4G core networks.

users could stay connected while moving. It encompasses a suite of protocols that are functionally classified into GTP-C for control plane, GTP-U for data plane, and GTP' for charging purposes, as detailed in Table 1. Version 0 of GTP has been obsoleted. GTP-C version 1 (GTPv1-C) is used in 3G, and GTP-C version 2 (GTPv2-C) is used in 4G. In 5G networks deployed in Standalone (SA) mode, GTP-C has been removed. Despite the evolvement of GTP-C, GTP-U remains in version 1 and continues to serve the 5G data plane. Furthermore, as the Non-Standalone Access (NSA) remains a prevalent 5G deployment approach [12], the 5G NSA mode still employs the 4G EPC, which is susceptible to the security concerns associated with GTP-C. As GTP' is solely used for billing and interacts with a limited range of network elements, it is excluded from our work. This paper primarily focuses on the protocols GTPv1-C, GTPv2-C, and GTPv1-U (shortly GTP-U thereafter).

TABLE 1: Overview of GTP variants.

Protocol	**GTP-C** [21], [23]	**GTP-U** [24]	**GTP'** [20]
Functions	Control Plane	Data Transfer	Charging
Port	2123	2152	3386
Transport	UDP	UDP	TCP/UDP
Version 0	Obsoleted		3G/4G/5G
Version 1	3G	3G/4G/5G	
Version 2	4G/5G (NSA)	-	

Message types and formats. Generally, a GTP message consists of a GTP header (and extended headers) with critical fields like *Protocol Version*, *Message Type* and *Tunnel Endpoint Identifier* (TEID), along with a set of Information Elements. *Message Type* spans a value range from 0 to 255, and mandatory and optional information elements for each message type are explicitly defined in the standards [21], [23], [24]. Table 8 (in Appendix A) categorizes GTP messages by their functions. Notably, the G-PDU message of GTP-U is a tunneling message whose payload carries user data, as shown in Figure 3. The types of GTP-C messages are more diverse, and they are essentially used for tunnel management (e.g., creation, modification, and deletion). The

TEID (4 bytes) is also one critical field in the GTP header, which uniquely identifies a GTP tunnel.

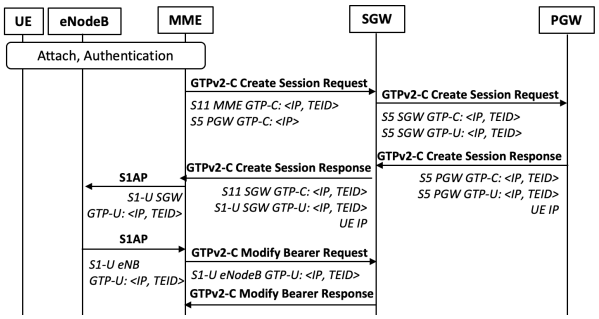


Figure 4: Creation of default GTP tunnels during UE attach.

GTP tunnel establishment. To help better understand this work, we first explain the definition of GTP tunnels, and introduce its working mechanism with an example. A GTP tunnel is a "path" established between network elements, transferring data or signaling information, and uniquely identified by the TEID in the GTP header. A GTP tunnel between two core network elements is unidirectional, and the TEID is assigned by the receiving end of the tunnel. Figure 4 depicts the setup of default GTP tunnels (as shown in the 4G core of Figure 3) during UE attach process, including the eNB↔SGW and the SGW↔PGW GTP-U tunnels, the MME↔SGW and the SGW↔PGW GTP-C tunnels. After UE establishes radio connection with the base station and is authenticated, the MME selects the SGW and PGW from the core network and initiates tunnel creation. Given that data communication is inherently bidirectional, there are always a pair of GTP tunnels established between two endpoints, both of which must share their TEIDs (and possibly IP addresses) to the other end. As shown in Figure 4, the MME begins by sending a GTPv2-C Create Session Request to the SGW, which includes its own IP and TEID for the GTP-C SGW→MME tunnel, as well as the IP address of the PGW to which the SGW should connect. The SGW then initiates a request to the targeted PGW with its own information. The PGW will respond to the SGW accordingly, so will the SGW to the MME. Consequently, the bidirectional GTP tunnels between the SGW and PGW (both GTP-C and GTP-U) and between the MME and SGW (only GTP-C) are established. Notably, the response from the SGW to MME also includes the TEID of the GTP-U eNB→SGW tunnel. The MME conveys the SGW details to the eNB, which then provides the MME with the information on GTP-U SGW→eNB tunnel. The MME finally issues a GTPv2-C Modify Bearer Request to the SGW to complete the GTP tunnel establishment.

GTP protection. As with most other IP-based tunneling protocols except IPsec Encapsulated Security Payload (ESP) [49], GTP itself does not offer any security protection. Rather, it depends on physical security, IPsec, or Datagram Transport Layer Security (DTLS) [68]. In practice, the underlying security protection may not always be in place and mobile operators often rely upon access control (e.g., using GTP firewalls) to protect their GTP infrastructure. Given the complexity of commercial cellular systems (e.g., in supporting various roaming scenarios), it is challenging to implement stringent access control in the network perimeter.

3. Identify GTP Hosts

To answer the first research question on how effective the current protection of GTP infrastructure is, we aim to discover exposed GTP nodes in cellular core networks from the public Internet. This section details our detecting strategy, and the next section characterizes the detected GTP nodes. For ethical reasons, we aim to uncover a broad range of GTP-operational devices without impacting them. Naturally, our detection leverages the built-in GTP echo request/response mechanism for live probing. Key steps include: 1) creating an extensible probing tool to discover GTP live nodes, and 2) verifying scanning results to filter out false positives from decoys like honeypots. Figure 5 visualizes our methodology, which we will elaborate below.

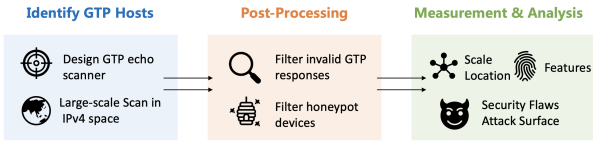


Figure 5: Overview of GTP hosts identification and analysis.

3.1. Active Scan of GTP Hosts

Select probing methods. We employ active probing to pinpoint GTP devices on the public Internet. By examining the specifications of GTPv1-C, GTPv2-C, and GTP-U, we observed that these protocols support various request/response messages. To avoid any adverse effects of active probing, we need to ensure our probing messages do not lead to operations that could impact network or service functionality. After careful consideration, we opted to use only path management messages (GTP Echo Messages) for probing. All three GTP protocols under study support the echo message to allow a GTP node to check the status of its peer. The specification mandates the response to a received GTP echo request, with an error status code if applicable. This feature enables us to reach a wide range of GTP entities. Since sending and receiving protocol-compliant GTP echo messages do not lead to (harmful) operations of network elements, we utilize echo messages with controlled rate limiting to probe active GTP hosts on the public Internet.

Develop extensive scanning tools. Network scanning is a relatively mature research field, but existing scanning tools such as Zmap [34] and Nmap [59] are primarily geared towards Internet protocols, with limited support for protocols in cellular systems. In this work, we extended an open source tool, Xmap [56], to support the construction and parsing of GTP messages, and fast scanning of GTPv1-C (port 2123), GTPv2-C (port 2123) and GTP-U (port

2152) via echo messages. The extended tool also supports checking the validity (see details in Section 3.2) of GTP messages according to standards. Considering ethical risks, our tools will only be made available for research purposes on a request-on-demand basis. In addition, we note that among the mainstream network search engines, Shodan [10] does implement GTP scanning (using the command *product: "GPRS Tunneling Protocol"* for relevant results). However, it only supports version 1 (GTPv1-C and GTP-U), leaving out version 2 (GTPv2-C) which is more widely used in 4G core networks. Moreover, it does not verify scan results or filter out invalid responses. Detailed comparison between our data and Shodan will be presented in Section 3.3.

Large-scale measurement. Our scans were executed on 4 AliCloud servers located in China (CN), India (IN), France (Fr) and Australia (AU) in March 2023. Final (filtered) results from each scanning node are listed in Table 2. The scanning targets are the entire IPv4 public address space, with a scan rate of 10k packets (towards 10k unique targets) sent per second. After sending, we maintain a listening state for the response for 60 seconds and preserve the raw data from the responses. Note that in each experiment, we send only one probe packet to each target address (the length of the UDP packet is 12 bytes). At most, only one retransmission would be attempted when no response is received, to minimize the impact on the target server. Detailed ethical considerations of the scanning experiment are listed in Section 3.4. At this step, all responses on the UDP port would be retained.

3.2. Post-Processing

Since a response from a GTP port does not always mean that GTP services are supported, we need to further verify GTP responses to exclude as many nodes as possible that may not belong to the cellular system. Therefore, we performed a series of post-processing on the received responses.

Filter invalid responses. Firstly, we need to ensure that the reply messages conform to the format of GTP responses. This is not a simple task. Although the GTP header is straightforward, the message body may contain many types of Information Elements, some of which can be operator-defined. Therefore, we first referred to the protocol parsing capabilities of Scapy [6] to help identify the header formats of GTPv1-C, GTPv2-C, and GTP-U. We deconstructed the response messages using scripts provided by Scapy, then clustered the specific response content and used manually constructed rules to filter out invalid segments. Specifically, we filter out responses with invalid values in the header (e.g., non-existent or invalid message types). For IEs, we parse publicly defined IEs and treat unrecognized data as private IEs. Responses with at least one recognizable IE are considered valid, and those with only private IEs are filtered out. For example, we noticed that over $4,000$ GTPv2-C echo request messages received a response of a type of *Delete Bearer Request* (message type value 99), indicating an anomaly. Further analysis revealed that the raw data of this response was the English string "Access Forbidden", whose binary signature coincidentally matched the header of a type 99 GTPv2-C message, leading to a misinterpretation. Overall, in this step, invalid responses from 13,228 hosts were excluded.

Exclude network honeypots. Network honeypots may also respond validly to our GTP echo messages, thus we need to identify and exclude potential false positives arising from them. We recognize that the precise identification of honeypots is quite challenging, and tried our best to minimize their impact. We first surveyed GTP-specific honeypot productions through search engines, but did not find any well-known or widely deployed commercial products (only found TunnelTrap [8] of Nokia, which has no deployment information and has not been updated for years). Therefore, we drew on the practices in prior work of filtering common network honeypots [30], as our best efforts.

Our approach to filter honeypots consists of two aspects: 1) the open service fingerprints on the host, and 2) its network interaction behavior. Specifically, for each potential GTP active host we scanned, we first reviewed its scanning history on Shodan [10], and recorded hosts labeled as "honeypot" by Shodan as ground-truth. Subsequently, following prior works [30], we hypothesize that devices within the same subnet as the honeypots and exhibit similar network service deployment scheme are likely to be honeypots as well. Therefore, we utilized Nmap [59] to scan the Top 1,000 Internet Services[1] for potential GTP devices, and removed hosts in the same /24 segment running identical services on the same ports as the honeypot ground-truth set. Besides, to avoid interference from devices that mimic echo functionality without truly supporting the GTP service protocol, we also send random payloads (e.g., "abc" and "123") to the scanned GTP active ports. If a UDP response is still received, we also classify such a device as a honeypot and exclude it. In this step, we exclude 2,542 hosts that are suspected to be honeypots.

3.3. Evaluation

After the aforementioned filtering, we identified a total of 749,000 active GTP hosts, as listed in Table 2. These exposed nodes encompass core network elements, as well as base station devices, which also support GTP functions. Note all GTP interfaces of mobile networks should be internal and not accessible from the public Internet. However, a primary challenge is the absence of ground truth. Operators do not publicly disclose the details of their core network deployments. Those, with which we discussed our findings, were notably cautious in sharing information on exposed GTP nodes. A few operators did provide feedback on the potential sources of these nodes (see Section 4.2 for details), but no quantitative data were shared, let alone the usages of specific IP addresses. Therefore, we use the results from the leading network search engine Shodan as a reference to first

1. To improve efficiency, Nmap provides the option to scan the most popular 1,000 protocol port pairs (https://nmap.org/book/performance-port-selection.html).

evaluate the coverage of our method. Subsequently, we will also discuss the accuracy of our approach.

TABLE 2: GTP-supported hosts discovered via scanning.

Scan Node Location	GTPv1-C # IP	GTPv2-C # IP	GTP-U # IP	All # uniq IP
CN	94,191	600,942	159,902	681,257
IN	45,831	335,094	81,331	410,247
FR	44,862	333,861	81,629	409,845
AU	38,130	345,577	59,141	402,105
All (unique)	101,984	669,161	173,298	749,000

TABLE 3: GTP active hosts found by our results vs. Shodan.

Protocol	Our results	Shodan	Intersection	False Negative
GTPv1-C	101,984	74,484	70,072	4,267 (4,412)
GTPv2-C	669,161	-	-	-
GTP-U	173,298	128,869	122,350	6,421 (6,519)
All (unique)	749,000	147,591	139,504	7,874 (8,087)

Comparison with Shodan. We found mainstream Internet search engines currently fall short in supporting cellular services, e.g., Censys [33] does not offer GTP scanning, and Shodan is limited to the GTP version 1 (including GTPv1-C and GTP-U) scanning module. To evaluate our findings, we used the results from Shodan (in Sept. 2023) as a benchmark[2]. The comparative analysis is detailed in Table 3. While the number of GTP active hosts detected by a single regional node may be limited (see Table 2), the aggregated results from the 4 scanning nodes make a larger coverage than Shodan, proving that our distributed scanning strategy is effective in extending coverage. We specifically analyzed the hosts that our scan missed, as indicated in the "False Negative" column. The raw results we missed are shown in (), while the numbers outside () are adjusted as we found some results in Shodan should be filtered out as invalid responses. For instance, 103 GTPv1-C responses in Shodan were not invalid echo response messages, which have been excluded in our post-processing step. After this calibration, our overall rate of missed detections is 5.34% (7,874 of 147,378). The discrepancy may primarily arise from the different geographic locations of scanning nodes and the six-month interval between the scan timings (e.g., devices with dynamic IP addresses may be missed; some GTP nodes may have been decommissioned). Notably, by employing GTPv2 scanning, which Shodan does not support, our scans identified a large number of (600K) active GTP hosts undiscovered by Shodan. Coupled with the validation checks in our tool to eliminate false positives, we provided a far more accurate assessment of GTP exposure than Shodan. We believe our results are adequate in revealing the current state of lack of protection in real-world GTP deployment.

2. Note that there is a 6-month gap between Shodan data collection and our scans. We cannot obtain fully time-aligned Shodan data since Shodan doesn't support downloading historical raw data currently.

3.4. Ethics

The primary ethical concern of this work is the scanning conducted in the real-world network environment. We strictly followed the guidelines of the domain authorities [48], [64] and carefully designed the experiments to mitigate potential ethical risks.

First, we used the extended Xmap to scan GTP devices and Nmap to scan the Top 1000 services of discovered GTP devices. Our Xmap scan followed the best ethical practices (as Zmap), including randomizing target addresses, controlling the scanning rate (conservatively scanning one IPv4 address space over 50 hours) and maintaining exclude-lists (e.g., special-purpose addresses [4]). The Nmap scan followed the same principles. Furthermore, despite the diverse range of GTP messages, we exclusively used legal GTP echo request messages for scanning, which are non-intrusive and do not harm target devices. We retried only once if no successful response was received. We did not construct or send any malicious payload to GTP devices in our scanning. Our measurement operations align with those of well-recognized Internet search engines like Shodan [10], without introducing new risks. Our measurement tools will be made available exclusively to operators or certified researchers upon verification.

Second, we did not disclose specific information (e.g., IP addresses) of the detected GTP devices in this paper, and anonymized corresponding operators based on their feedback. We reported our findings to the GSMA, which is the organization facilitating vulnerability disclosure within the cellular community. Details of vulnerability disclosure are discussed in Section 8.

Last, our experiments did not directly involve human subjects and only used nodes under our control for scanning. We did not collect any actual user data, nor attempt to attack real users or devices. All attack validation experiments (in Section 5) were conducted against our own UE devices using open-source platforms in a controlled environment. Therefore, we believe that our research has no negative impact on operational networks or real users.

4. Characterize Exposed Hosts

4.1. Scale and Distributions

We first analyze the distributions of the 749,000 identified GTP hosts. As shown in Table 2, the most prevalent protocol (89.34%) was GTPv2-C. According to the standard [23], the 4G control plane in core networks should use GTPv2-C. This indicates that the majority of the detected devices are likely operating on 4G services, and potentially on 5G networks with NSA architectures. Our findings are in line with the global trend of mobile communication systems, where 4G and 5G networks are prevalent (accounting for over 70% of cellular traffic in 2022 [12]).

According to IP WHOIS information, the devices we found are distributed across 1,176 Internet Service Providers (ISP) in 162 countries. The distribution among individual

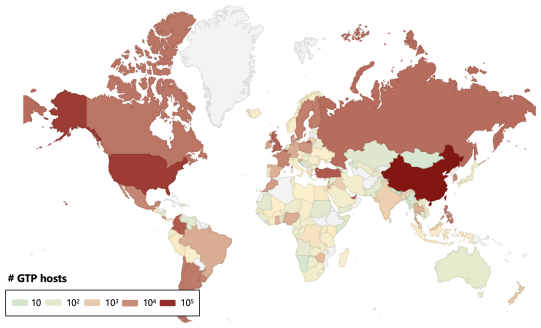


Figure 6: The global distribution of active GTP hosts.

ISPs follows a long-tail distribution, with the top 5 accounting for 51.79% of all GTP devices. We found that several of the top holders of GTP IP addresses are associated with the telecommunication industry. Specifically, 7 of the top 10 are well-known mobile operators, and another 2 have keywords like "tele" in their names. Besides, upon analyzing their geographic distribution (Figure 6), we found the exposed devices are more concentrated in several countries: China (349,555, 46.67%) has the highest proportion, followed by the United States (78,217, 10.44%), Colombia (30,916, 4.13%), Turkey (29,214, 3.90%), and Belgium (24,701, 3.30%). Considering the limitations of our scanning nodes in geographic coverage, the results could be biased if GTP nodes in certain networks respond only to nodes located in the same region. We analyzed the Shodan results (the same used in the evaluation part), and found that the majority were also located in China (62,239 devices, accounting for 45.37%, even exceeding the proportion we calculated), with the United States ranking second (22,387 devices, 16.32%). This suggests that regional bias may not be significant enough to affect our conclusions.

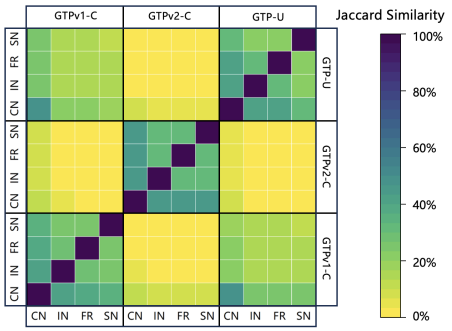


Figure 7: Similarities of the active GTP host sets scanned using different protocols from different measurement nodes.

We also examine the overlap between active GTP host sets across different protocols. As shown in Figure 7, the overlap is quantified using Jaccard similarity, i.e., the intersection of two sets divided by their union. We find that the overlap rate between GTPv1-C and GTP-U services is relatively high (95,653 hosts, 93.79% of GTPv1-C). According to the application scenarios of GTP, these nodes might be SSGN or GGSN in 3G networks. 73,794 hosts only support GTP-U, which could include base stations, SGW-U, and PGW-U. The overlap between GTPv2-C and other protocols is relatively low, consistent with the design principle of control and user plane separation in 4G. 569,371 hosts only support GTPv2-C, which could include MME, SGW-C, and PGW-C. 3,379 hosts support both GTPv2-C and GTP-U, which could include SGW and PGW. We also find that 96,123 hosts support both GTPv1-C and GTPv2-C, which may be deployed for the purpose of backward compatibility and interoperability between 3G and 4G.

4.2. Potential sources of GTP devices

Despite the effort we took to filter out invalid results, one question remains: how many of the detected GTP hosts actually belong to operational cellular networks? Although GTP is designed specifically for cellular systems, there are also other potential application scenarios, e.g., experimental nodes, applications or systems supporting GTP functionalities by default.

To this end, we examined operating systems and applications that are inherently configured with GTP libraries. Linux is the only mainstream OS with native support of GTP through libgtpnl [5]. However, it only implements the encapsulation and decapsulation of GTP-U, and users need to implement GTP-C in user space to manage GTP tunnels [7]. Routing devices such as Juniper Networks products based on Junos OS [2] and Cisco switches also support GTP [3], but they too require manual activation. Unless due to software bugs [1], ordinary Internet devices not used for cellular functions are unlikely to respond to GTP echo requests in their default settings.

We also considered classifying exposed GTP nodes more precisely, but faced several practical challenges. Firstly, we cannot send other types of GTP messages to further fingerprint those devices, due to ethical considerations. GTP messages other than Echo request may have impact on a target device, e.g., resulting in state changes, thus cannot be sent to operational devices without proper authorization from an operator. Secondly, we tried to communicate with operators both directly and via GSMA to understand the nature of those exposed GTP nodes, but were unable to obtain precise information. Most of the operators we contacted are unwilling to share any information due to the sensitivity of the issues. A few did provide feedback and acknowledged that the IPes we reported to them indeed belong to them. But they did not provide us with concrete information on those IPes. One thing we learned from the discussions with operators is that IP address management within an operator is complicated, and often involves multiple groups and even customers. It is even challenging for them to find out what a device is given an IP address.

In summary, we feel confident that our discovered GTP nodes belong to operator networks including mobile networks, but we acknowledge our limitation in not having an accurate categorization of the results.

4.3. Deployment Characteristics

A recent trend in cellular core networks is transitioning towards cloud service architectures. As mentioned in the background, operators like AT&T and Dish plan to deploy their 5G core networks on the public cloud platform of Amazon [28], [55]. The state of cloud deployment in cellular networks remains unknown to the public. Therefore, we attempt to investigate how many of the exposed GTP devices are being deployed on the public cloud.

TABLE 4: GTP hosts deployed on cloud platforms.

Category	Provider	# FQDN	Example of FQDN Pattern
Public Cloud Service	Akamai	306	[*].deploy.static.akamaitechnologies.com
	Amazon	20	[*].compute-1.amazonaws.com
Telecom Cloud Service	operator-I	3,303	total-pool[*].operator-I.net.co
	operator-II	2,061	[*].pool[*].static.operator-II.es
	operator-III	1,722	host-[*].operator-III.am

Due to the lack of ground-truth, we refer to previous works [60] to use auxiliary information to identify cloud devices. Specifically, we first use reverse DNS lookups (PTR) to see if the found GTP hosts have fully qualified domain name (FQDN) deployed on that IP, and infer deployment details by analyzing the semantics of FQDNs. Among all active GTP hosts, a total of 271,899 (36.30%) IPs have configured PTR records of 260,895 unique FQDNs under 3,026 second-level domains (SLDs). We then used a semi-supervised method to classify cloud-related FQDNs: 1) search for keywords related to cloud deployment such as "cdn", "host", "pool" in FQDN strings, and 2) collect a list of popular cloud services, and match their brand domains in the FQDNs.

We identified 12,022 (4.61%) FQDNs related to cloud service deployments, covering two popular public cloud service vendors (Amazon and Akamai) and 97 SLDs that are potentially owned by mobile operators. Table 4 presents details of the two discovered public cloud platforms and the top three telecom cloud services (specific names of operators have been anonymized). We also examined the naming rules of these FQDNs. In Table 4, the "*" typically represents the (part of) IP address hosted by that domain. Compared to public platforms, we found the most exposed GTP devices are deployed on telecom operators' self-built cloud architectures. For example, operator-I offers cloud-based voice solutions and business telephone services, and holds the most (3,303) FQDNs.

5. GTP Security Analysis

Based on measurements, we have answered the first research question, confirming potential exposure of real-world GTP infrastructure to the public internet. The next goal is to understand security risks associated with publicly accessible GTP devices. This section systematically analyzed the security risks posed by unauthorized GTP messages to cellular core networks (including core elements and base stations). We also conducted validation experiments to demonstrate the effectiveness and feasibility of attacks. For ethical reasons, all the experiments were conducted on open-source projects in a controlled laboratory environment.

5.1. GTP Security Risks

Threat model. We assume the attacker is external (located in the public Internet), without direct control over cellular core network nodes, but capable of sending arbitrary GTP messages. The attacker first identifies devices that belong to the cellular network by scanning (akin to the proposed method in this work), and then sends carefully constructed GTP messages to these devices. The goal of the attack is to disrupt normal cellular network services, and the specific effects include but are not limited to, denial-of-services, traffic hijacking, and user tracking.

Risky GTP messages. To understand which types of GTP messages can be exploited and the resulting security risks, we performed an in-depth analysis of GTP specifications [21], [23], [24]. We started by categorizing GTP messages based on their functionalities, and then analyzed the security risk of each type. We identified 38 GTP message types that could be exploited for 6 classes of attacks, as detailed in Table 5. Due to space limitations, we briefly describe the high-level idea of the analysis. We found that GTP messages can be classified into 3 categories:

- *Messages that can directly alter tunnel states, including creation, deletion, and update*. An overabundance of creation messages being transmitted to an exposed GTP node can deplete its resources and potentially instigate DoS attacks. Falsified deletion messages may delete user contexts or GTP tunnels and have the potential to disrupt ongoing services. Moreover, well-crafted update messages may redirect the tunnel endpoint to a node controlled by an attacker, facilitating session hijacking.
- *Context queries or notification messages*. Such messages may facilitate remote node identification and information disclosure. For example, echo messages can expose active nodes, and UE registration status queries can be exploited for user tracking.
- *Tunneling messages carrying user data*. Such messages could be spoofed to inject data into a tunnel, e.g., to inflate billing charges to a victim.

Note we only consider threats from a single GTP message. More sophisticated scenarios, such as combining multiple GTP messages or with other signaling protocols are left for future work. Next, we present the experiments in a lab environment for validating the feasibility of the attacks outlined in Table 5, offering additional insights into the attacking process and its prerequisites.

5.2. Evaluation of Attacks

Survey of GTP implementations. Considering that 3G is in the phase-out stage, we focus on 4G and 5G implementations, including srsRAN [19], Open Air Interface [16], Open5GS [14], and free5GC [13]. The versions we surveyed and the detailed support information can be found in Table 9

TABLE 5: Attacks that could be triggered by GTP messages.

Protocol	All		GTP-C (v1)											GTP-C			GTP-C (v2)																										GTP-U	
Attack/Message	*Echo Request*	*Supported Extension Headers Notification*	*Create PDP Context Request*	*Delete PDP Context Request*	*Update PDP Context Request*	*PDU Notification Reject Request*	*Send Routeing Information for GPRS Request*	*Failure Report Request*	*MBMS Notification Reject Request*	*Create MBMS Context Request*	*Delete MBMS Context Request*	*MBMS De-Registration Request*	*UE Registration Query Request*	*MBMS Session Start Request*	*MBMS Session Stop Request*	*Create Session Request*	*Delete Session Request*	*Remote UE Report Notification*	*Delete Bearer Command*	*Delete Bearer Failure Indication*	*Bearer Resource Command*	*Bearer Resource Failure Indication*	*Downlink Data Notification Failure Indication*	*Trace Session Deactivation*	*Stop Paging Indication*	*Create Bearer Request*	*Delete Bearer Request*	*Modify Bearer Request*	*Forward Relocation Request*	*Delete PDN Connection Set Request*	*Relocation Cancel Request*	*Create Forwarding Tunnel Request*	*Create Indirect Data Forwarding Tunnel Request*	*Delete Indirect Data Forwarding Tunnel Request*	*Release Access Bearers Request*	*Downlink Data Notification*	*End Marker*	*G-PDU*
Node Discovery	●																																									
DoS (Resource Consumption)			●							●				●		●					●					●			●			●	●			●					
DoS (Service Disruption)		●		●		●		●	●		●	●			●		●		●	●		●	●	●	●		●			●	●			●	●		●			
Session Hijacking					●																							●													
Data Injection																																									●	
User Tracking							●						●					●																								

in the Appendix B. We found that they all implement the basic functions of GTP-U (especially G-PDU), while the support of GTPv2-C is quite limited, with 59 (70.24%) types of messages not supported by any implementation. Even for the supported messages, experiments may still be constrained due to the simplified implementation not supporting full message functionalities. Node Discovery is how we identify the exposed GTP nodes, and will not be described in this section. We validated 4 other types of attacks except for User Tracking due to the lack of implementation of corresponding GTP messages.

Experiment setup. Our attacks exploit both GTP-C (5.2.1–5.2.3) and GTP-U (5.2.4) messages. Given that Open5GS EPC (4G) has the best support of GTP-C messages among all open-source projects (see Table 9 in Appendix B), it is chosen to demonstrate GTP-C related attacks. We set up the Open5GS EPC in an Ubuntu 22.04 system, paired with a srsRAN eNB connected to a USRP B210 for radio functions, and used a commercial mobile device (Redmi Note 9 5G) as the potential victim. The GTP-U attack (5.2.4) impacts both 4G and 5G networks, so we opted for a 5G environment for completeness. This 5G testbed used the core network of free5GC and the gNB and UE components of UERANSIM [13] (the default configuration of free5GC), installed in an Ubuntu 20.04 system.

Note that an attacker needs to infer specific parameters in GTP headers and Information Elements to execute attacks. As different message types may necessitate distinct parameters, we explain the necessary inferences and their corresponding methods in each attack scenario separately.

5.2.1. DoS Attack (service disruption). GTP-C messages that are employed for "delete" operations can be misused for service disruption. For example, *Release Access Bearers Request* sent by the MME to the SGW via the S11 interface is used to release a UE's established bearers (GTP-U tunnels between the eNB and SGW and the radio resources) for reasons such as energy saving in UE idle mode [23]. An attacker could launch a denial-of-service attack by sending a carefully forged message to the SGW to release the GTP-U tunnel to disrupt normal data service for a victim UE. According to the standard [23], aside from the GTP header, this message is not required to carry additional Information Elements. Thus, the minimum attacking requirement is to know the TEID of the victim's GTPv2-C tunnel on the MME→SGW interface (i.e., to reuse the legitimate control channel). We will discuss the feasibility of guessing TEID at the end of this subsection.

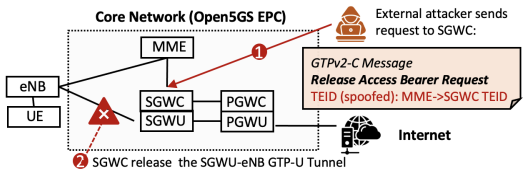
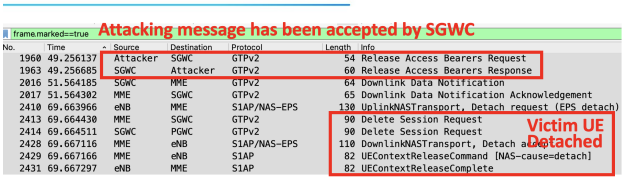


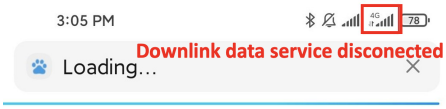
Figure 8: Abuse GTPv2-C release messages for DoS attack.

Attack verification. We verified this attack in Open5GS, as shown in Figure 8. In the Open5GS EPC, the control plane and data plane of SGW and PGW are named with the suffixes C and U, respectively. We deployed Open5GS using docker (one container per network element) and launched another docker instance to simulate an attacker. We assume that the attacker already knows the IP address of the SGWC and can send packets to it (e.g., by scanning, see Section 3). Before the attack, the user connects to the core network using a commercial phone and can access Internet services. The attacker then sends a *Release Access Bearers Request* to the SGWC, with a header containing the TEID of the MME→SGW GTP-C tunnel, to forge an MME-initiated release request. Figure 9 demonstrates the effect of this attack on the victim UE. Our carefully crafted attack message

has been successfully received and processed by SGWC (in Figure 9a). Then the downlink data service of UE was disconnected (in Figure 9b), resulting in the UE's inability to access Internet services. We observed that the UE was detached from the network about 20 seconds after the attack occurred. This disconnection would last for at least 1 minute until the UE is re-attached to the network. Besides, we found that if an incorrect TEID was used, the SGWC would send a "Context not found" code in the response. This can facilitate the guessing of TEIDs.



(a) The attacker sent a forged Release Access Bearers Request to SGWC, causing the SGWU→eNB downlink tunnel to be released. After a while (20 seconds), the victim UE was detached.



(b) The downlink data service of victim UE has been disconnected.

Figure 9: Affect of DoS Attack (service disruption).

5.2.2. DoS Attack (resource consumption). Locally maintainable resources of GTP are finite (e.g., IP addresses and GTP tunnels). If an external attacker can arbitrarily create GTP tunnels, resources on legitimate GTP nodes may be exhausted, also leading to DoS attacks. We identified several GTP messages with such properties, as summarized in Table 5. We use *Create Session Request* as an example for demonstration. Under normal circumstances, this message might be sent by the MME to the SGW, and from the SGW to the PGW, to establish a PDN connection. During this process, the PGW needs to allocate an IP address for the UE, and both the SGW and PGW need to create and maintain context for the new tunnels. Both IP addresses and memory spaces for active tunnels are finite. If attackers could send a massive number of creation requests directly to the SGW/PGW, it could deplete its resources and prevent legitimate UEs from connecting. It is worth noting that a legitimate PGW only accepts *Create Session Request* containing an IMSI with valid network subscription. To eliminate the need for collecting subscribed IMSIs, an attacker can fake a PGW (in the request sent to the SGW) using a forged IP address under its control, allowing tunnels with arbitrary IMSIs to be created in SGW. In this way, the attacker could exhaust the resources of the SGW efficiently. Alternatively, the attacker could also follow proposed feasible methods to collect valid IMSIs [31], [63], then send requests embedded these IMSIs to the SGW. Although this strategy requires more prerequisites, it also yields a more powerful impact (compromising both the PGW and SGW). Our tests also showed that unsubscribed IMSIs will return an "APN Access Denied - No Subscription" error code, which helps filter out IMSIs without subscriptions.

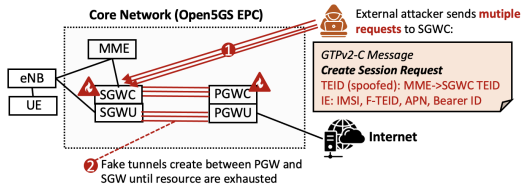


Figure 10: Abuse GTPv2-C creation messages for DoS attack.

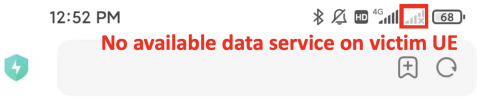
Attack verification. This attack was also validated on Open5GS EPC, as shown in Figure 10. This attack assumes that an attacker can learn the IP addresses of both the SGW and PGW, e.g., through scanning or other methods, and can send packets to the SGW (exposed) GTPv2-C interface. It (disguised as an MME) sends a *Create Session Request* to the SGW. Assuming that the SGW fails to authenticate the request, upon accepting the request, it will also send a *Create Session Request* to the PGW. The attacker also needs to guess the MME→SGW GTP-C TEID, and include it in the GTP header. When the attacking messages meet all the above conditions, our tests confirmed that two specific resource exhaustion attacks succeeded: 1) exhaust the range of IP addresses that PGWC can allocate (default value in Open5GS is 256), and 2) exhaust the number of UEs that SGWC can maintain GTP tunnels (default value in Open5GS is 1024). Requests that exceed these limits will cause the corresponding network element to assert errors and crash. After this, new users will not be able to access the network, until the network elements have been restarted. Figure 11 shows the effects of this attack. We acknowledge that a commercial SGW/PGW would be much more capable (e.g., supporting up to one million GTP-U connections [18]), and it would take a longer time to exhaust its resources.

5.2.3. Session Hijacking. Session hijacking is possible when an attacker has the opportunity to directly insert control messages that can alter the session state to take over established sessions. We use *Modify Bearer Request* as an example for demonstration. Due to UE mobility, this message is used to change the base station or SGW that a UE is currently connected to, in scenarios such as handovers. Attackers could abuse this mechanism to hijack sessions by sending forged messages to change established tunnel endpoints to IP addresses they control. Previously proposed MITM attacks [73], [74] in cellular systems are mainly focused on air interfaces (e.g., by fake base stations), thus requiring attackers to be in the same geographical location as the victim. However, hijacking user traffic directly from the core network can be done remotely, making it more powerful and difficult for users to detect the attack.

Attack verification. We also validate this attack via Open5GS EPC. Our experiment showed that by simply sending a forged *Modify Bearer Message*, an attacker can

(a) Forged Create Session Requests sent by the attacker exceed the limitation of IPs that PGWC could allocate. PGWC reports an error.

(b) Forged Create Session Requests sent by the attacker exceed the range of UE pool that SGWC could maintain. SGWC reports an error.



(c) Data service of victim UE has been interrupted.

Figure 11: Affect of DoS Attack (resource consumption).

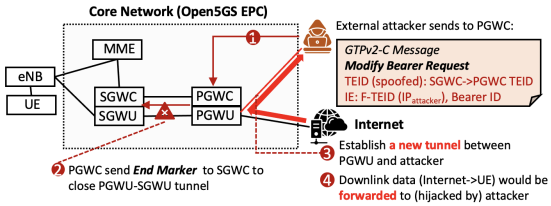


Figure 12: Abuse GTPv2-C messages for session hijacking.

hijack the victim's downlink GTP-U tunnel data directly from the core network, achieving adverse effects such as off-path hijacking. The specific attacking process is illustrated in Figure 12. The goal of the attacker is to hijack the PGW-SGW GTP-U tunnel, by impersonating a new SGW and sending forged *Modify Bearer Request*. Notably, uplink and downlink tunnels of GTP are independent. Since existing open source GTP implementations only support modifications of the downlink tunnel, we demonstrated hijacking only downlink traffic by changing the path from Internet→PGW→SGW to Internet→PGW→attacker. The attacker needs to carefully fill the following fields in the request: 1) TEID in the header, using the SGW→PGW GTP-C TEID, to reuse the legitimate control channel; 2) F-TEID, using a new IP address and TEID controlled by the attacker as the new endpoint, and set the interface value to 5 (the SGW-PGW s5-U interface); 3) Bearer ID, set to the default value 5. It was observed that, after sending the attack message, the PGW would send an *End Marker* (GTP-U) message to the SGW, to close the previous connection, and establish a new GTP-U tunnel with the attacker. Thus the downlink traffic could be immediately intercepted. Note that the hijacking is not a full MITM, as the UE was still connected to the old SGW in the uplink (from base station to the old SGW), but the uplink tunnel between the old SGW and PGW has been closed (by the *End Marker*). Even so, we observed that the duration of downlink hijacking can last 10 to 30 seconds. Figure 13 demonstrates the attack effect. The attacker can capture all of the downlink traffic (domains, IPs, and services) during the attacking window. Especially for TCP connections, the attacker can continue to obtain the complete sessions by sending ACKs to the servers. If the user was transmitting information in plaintext, it would be fully leaked to the attacker. Although the attack window is limited and only the downlink traffic can be captured (due to the incomplete open-source core network implementations), it achieves a remote off-path hijacking of cellular network users, showing powerful attack effects.

Figure 13: Affect of Session Hijacking attack.

5.2.4. Data Injection. Attackers can also target GTP-U, which uses G-PDU type messages to tunnel user data. Currently, core network elements distinguish different GTP-U tunnels solely by the TEID number. Therefore, attackers can inject forged GTP-U PDU packets into core network interfaces to achieve various attack objectives.

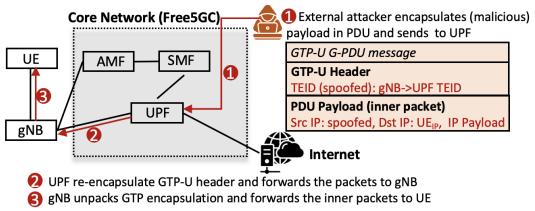


Figure 14: Abuse GTP-U G-PDU messages for injection.

Attack verification. We verified a GTP-U Data Injection attack targeting the UPF in free5GC, showing that 5G is also vulnerable. Note any other open-source projects (e.g., Open5GS) can also be used since they all support GTP-U. The core network of free5GC was deployed on a virtual machine, while another virtual machine was set up to simulate an attacker, as shown in Figure 14. The assumption is that the attacker knows the IP (GTP-U interface) of the UPF (e.g., by GTP scanning) and the IP of the UE (e.g., by sniffing over the air interface as the user plane is usually unencrpted [61]). Note that the UE may use a private IP. In this case, the attacker cannot directly send unsolicited traffic to the UE from the public Internet without abusing GTP-U. During the attack, the attacker first sends a GTP-U PDU message to the UPF, which contains the spoofed TEID as the legitimate gNB→UPF GTP-U tunnel. The inner packet (payload) of the G-PDU message is the actual IP packet sent to the UE. Since the attacker has spoofed the TEID (gNB→UPF), the UPF would accept the message, remove the GTP-U header and prepare to forward the internal packet. As the target IP of the inner packet is the UE, UPF would then (erroneously) treat it as a downlink packet from

the Internet, then re-tunnel it with GTP, and forward it to the gNB. The gNB would decapsulate and forward the inner packet to the UE, completing the traffic injection. The effects of this attack would be muti-fold. Firstly, an attacker could inflate the victim's bill by (silently) sending large amounts of traffic to the victim UE. Besides, intensive data injection would also occupy the network bandwidth and affect the service (e.g., downloading) quality of victim UEs. Figure 15 shows the impact of traffic injection. The victim user had access to the Internet via the free5GC core network and was downloading files. When the attacker started to abuse the GTP-U PDU message to inject arbitrary traffic to consume the victim's downstream bandwidth, a significant decrease in download speed is observed, which can even drop to almost zero (denial-of-service).

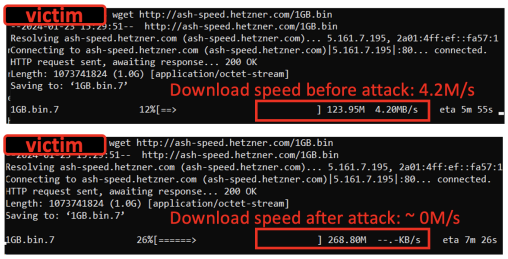


Figure 15: Affect of Data Injection attack.

5.2.5. User Tracking. Inferring a mobile user's presence in a certain network or its geographic location is a typical attack scenario in cellular security research [31], [40], [53], [54]. GTP messages could also be abused for tracking users. For instance, both GTPv1-C and GTPv2-C support the *UE Registration Query Request*, which allows querying the MME to check whether a specific UE (IMSI) is registered. If the "Cause" in the response is "Request Accepted", it indicates that the UE is registered, and the MME will also reply with its specific PLMN. As MME usually operates regionally, an attacker could abuse the query to infer the status of a specific IMSI. We have not validated this attack as the relevant messages/functions are currently not implemented in any of the open source projects we investigated.

5.2.6. Feasibility of real-world attacks. The GTP attacks demonstrated in the open source environment are primarily based on the design weaknesses of GTP (i.e., lack of origin authentication), thus they are theoretically transferable to real-world deployment, albeit with significantly increased challenges. Although we can not validate attacks in real-world deployment due to ethical considerations, we discuss the feasibility below.

An (external) attacker first needs to identify GTP devices and exposed interfaces, which has been verified in the measurement part of this paper. The attacker can further fingerprint network elements into specific functions, e.g., MME and SGW. Although this is a challenging task with ethical considerations, it may not be so for attackers. Nevertheless, since the number of open GTP ports that can be detected is also limited (less than a million found in this work), even if an attacker launches indiscriminate attacks on all GTP devices, the additional costs required appear feasible.

Another key factor is whether the critical fields required to construct an attack GTP message can be obtained, including IMSI, TEID, and other values. 1) IMSI is the identity of a victim. Many studies have proposed effective methods for sniffing or inferring IMSI from 3G to 5G networks [32], [40], [54]. In addition, we also found that it is possible to infer whether a specific IMSI is correct/registered based on the error information returned by the core network. Therefore, obtaining the IMSI (even its registration status) is not impossible. 2) TEID, a unique identifier of a GTP tunnel, is of 4 bytes. Theoretically, an attacker needs to brute-force a TEID with up to 2^{32} guesses. However, practically, implementation characteristics or flaws often allow for reduced attack space. For example, some implementations may start with an initial TEID value of 1, then increment by one for each new tunnel. Some may not use the full 32-bits for TEID. For example, Open5GS [14] only uses 2 bytes for TEID (by default). Besides, according to the *Error Indication* mechanism [24], a TEID without an established context will trigger error codes in the response while a correct TEID will not, allowing an attacker to guess TEID effectively. 3) Other fields can also be guessed. For example, a bearer ID is mandatory in the *Modify Bearer Request* in the Session Hijacking attack. This field has only 4 bits, and with a default starting value of 5. An attacker can start trying from 5 until the attack is successful.

To summarize, we suggest that it is feasible to implement GTP-based attacks in a real-world deployment. We believe that the attacks in a lab environment serve the purpose of demonstrating security risks from exposed GTP nodes in operational networks.

6. Extended Attack Surface

Besides threats against cellular subscribers and infrastructure due to the lack of GTP message authentication and access control, we found that the attack surface exposed by GTP is far more extensive. This section discusses the implementation issues of identified GTP hosts, which could be abused for Reflective Denial-of-Service (RDoS) attacks.

In our measurements, we discovered that the echo responses from several GTP devices carried excessive information, making the packet lengths far exceed the standard response (∼14 bytes UDP payload). Some devices repeatedly sent responses after receiving an echo or even proactively sent echo requests to probe our scanning nodes. These behaviors are consistent with the patterns of reflection amplification attacks, meaning attackers can abuse GTP devices as a new type of UDP-based reflector on the Internet.

Amplification factors. Drawing from previous RDoS studies [71], to evaluate the impact of GTP devices as amplifiers, we need to measure two factors: 1) bandwidth amplification factor (BAF), the ratio of payload length received by the victim to that sent by the attacker, and 2) packet amplification factor (PAF), the ratio of the number of packets

TABLE 6: Evaluation of BAF. The 1st column indicates the number of hosts that satisfy the BAF value on the right side.

Protocol	GTPv1-C		GTPv2-C		GTP-U	
# hosts	BAF (valid)	BAF (all)	BAF (valid)	BAF (all)	BAF (valid)	BAF (all)
MAX (1)	23.3	333.3	23.3	333.3	170.7	333.3
>10	5.9	116.5	6.3	41.7	6.5	116.5
$>10^2$	3.9	116.5	4.0	4.9	4.0	116.5
$>10^3$	1.2	2.6	1.1	1.5	1.2	3.9
$> 10^5$	1.2	1.2	1.1	1.1	1.2	1.2

TABLE 7: Evaluation of PAF. The 1st column lists PAF values with numbers of hosts reaching the value on the right.

Protocol	GTPv1-C		GTPv2-C		GTP-U	
PAF	# host (valid)	# host (all)	# host (valid)	# host (all)	# host (valid)	# host (all)
2	10	372	28	37	286	635
3∼10	22	25	23	33	14	16
>10	0	3	3	8	2	6
All	32	400	54	78	302	657

received by the victim to those sent by the attacker. For the former, as defined by GTP standards [21], a GTP message comprises a header and a series of IEs. For each message type, several IEs are mandatory, many are optional, and operators can use their private IEs. Therefore, an attacker sending an initiating message might receive a triggered message carrying a lengthy IE (i.e., with a high BAF) as the payload. For the latter, while typically an initial message only prompts a single response, experiments revealed that several scanned GTP devices would perceive our test node as another GTP device, and irregularly send GTP messages to us, leading to multiple responses for a single request (i.e., with a PAF greater than 1).

For ethical reasons, we only tested the amplification of GTP devices using echo messages, while other message types may carry significantly longer payloads than echo. For instance, *Modify Bearer Response* specifies up to 25 IE types and permits operator-customized IEs [23]. Therefore, our test only provides a "bottom line" of amplifier capability.

Evaluation. The results are shown in Table 6 and 7. Notably, the GTP devices analyzed earlier in this paper are all (valid) GTP devices (excluding invalid responses and honeypots). To act as an amplifier, it does not need to be a (valid) GTP device. Thus, we evaluated two scenarios: one with only verified GTP devices (labeled as valid) and another with all devices responding to GTP echo messages (labeled as all). As shown in Table 6, a typical response is around 1.1∼1.2 times larger than the request due to extra IEs. However, GTP devices in practice may reply with excessively long messages. For instance, over 100 *valid* GTPv2-C nodes responded with packets x4 larger than the request. Manual inspection revealed that nodes added custom IEs or replied with irregular, mixed, and duplicated payloads, leading to a maximum BAF of 170.7 for *valid* GTP devices and even over 300 when considering devices like honeypots. For the PAF, as shown in Table 7, over 300 valid GTP nodes sent us multiple replies, following different sending patterns. For example, a GTPv2-C node at address 119.111.*.*, located in China, sent GTP responses at intervals of 5s, 5s, 5s, and 35s. Another GTP-U node, also located in China, seemed to treat our sending node as a valid peer and sent us echo request messages at irregular intervals. These diverse multi-response behaviors may be related to specific implementations of GTP, but they all can be exploited for reflection attacks.

7. Mitigation

Root cause. The security risks highlighted in this paper stem primarily from the inherent lack of security mechanisms within GTP and ineffective deployment of additional defenses. Based on 3GPP security standards [22], GTP protection relies on network and IP domain security. Network domain security primarily relies on physical measures, e.g., physical isolation, to prevent unauthorized access. It is ineffective for GTP, as GTP nodes may communicate across network domains (e.g., via roaming IPX or the Internet). IP domain security is based on IPsec, typically implemented at the network edge using IPsec gateways. While site-to-site IPsec provides traffic confidentiality and integrity across domain boundaries, it fails to prevent unauthorized access to GTP services. Besides, operators can enforce IP-based GTP message filtering. However, this approach faces deployment challenges, as several GTP interfaces (e.g., S8 between SGW and PGW in 4G, and N9 between UPFs in 5G) are roaming interfaces that run across network boundaries. Roaming traffic often traverses third-party networks, complicating IP-based filtering. Consequently, despite the potential to safeguard GTP interfaces, current cellular networks remain vulnerable due to GTP's inherent flaws.

Our recommendations. For the short term, operators can perform periodic GTP exposure checks. The scanning method and tools we developed can be provided to operators, facilitating periodic audits of their cellular core infrastructures. For the medium term, we suggest adding a built-in security mechanism to GTP to prevent unauthorized access. Specifically, GTP can add an authentication message field, e.g., based on shared secrets, to allow GTP tunnel endpoints to authenticate each other's messages. Such GTP authentication message serves a similar security purpose as TCP Authentication Option [77] and Secret Key Transaction Authentication for DNS (TSIG) [80]. For the long term, we expect that GTP will be replaced by protocols with inherent security protection including origin authentication. While GTP-C has been removed in 5G service-based architecture, GTP-U remains. We hope that GTP-U will also be moved toward a secure transport with mutual authentication (e.g., QUIC [47]) in the future (e.g., 6G). We plan to discuss with GSMA and 3GPP on both near and long-term mitigation.

8. Discussions

Limitation. The primary limitation of this paper is that, due to the absence of ground truth, we can not quantitatively

evaluate the coverage and accuracy of the identified GTP nodes. First, in terms of coverage, for ethical reasons, we exclusively use echo messages to find active nodes. Although it may restrict the capacity to obtain a more extensive view of GTP exposure, we prefer to provide a bottom line rather than introduce security risks. Compared to the known best GTP scanning platform, Shodan, we can cover 94.66% of its identified GTP nodes, and find 4 times more nodes (see details in Section 3.3). We believe our collected data is sufficient to highlight the practical security issues of GTP, especially as an early step in this new field. Second, in terms of accuracy, we can not ensure that all devices discovered belong to active commercial mobile core networks. Despite efforts to eliminate false positives such as honeypots, some experimental devices supporting GTP may still be included. Given that GTP has limited applications outside cellular networks, we believe the potential false positive rate is low. Our belief is reinforced by the analysis of supplementary data such as IP WHOIS and domain names (in Section 4.3) that show the relationship of exposed GTP nodes with cellular service providers. Additionally, two major operators have confirmed that the IPs we reported are real and should not be publicly accessible. Third, our attack evaluation can only be done in lab settings due to ethical reasons, thus does not directly demonstrate the effect against commercial networks. However, using open-source projects to verify attacks against core networks is common in cellular research (e.g., [27]). We attempted to discuss with one European and one Asian operators about reproducing attacks in their test environments, but both declined due to their IT security policies. We acknowledge that lab settings could not fully reflect the real-world conditions, and discussed the real-world attacking feasibility in Section 5.2.

Vulnerability disclosure. We have reported our findings to GSMA, which coordinates vulnerabilities and incident responses within the mobile ecosystem. The GSMA recognized GTP security risks but could not confirm if the reported IPs belonged to mobile core networks. One operator provided us feedback via GSMA that some of their exposed GTP nodes may be routers mistakenly enabled to respond to GTP echo requests (e.g., due to a known bug in Cisco routers [1]). But it did not provide information on the rest of the exposed nodes.

We have also directly notified 9 operators. One operator acknowledged that the reported IPes belong to their mobile networks and has since fixed their issues. Particularly, this operator told us that it took great effort within their company to find out what those IPs were and used this investigation as the opportunity to optimize their internal communication plan for future response. They also implemented internal monitoring for their GTP infrastructure, and thanked us for our reporting. A second operator told us that their exposed GTP nodes are from their Wi-Fi public hotspot networks, and is still investigating how to mitigate the issue. We provided our scanning tool to them based on their request and they confirmed that they can reproduce our results. Note that a Wi-Fi access gateway (WAG) may interwork with mobile core networks (e.g., SGW) using GTP to tunnel Wi-Fi traffic to the mobile core. A third operator acknowledged that our reported IPes belong to them, but was reluctant to discuss further. We did not hear anything from the rest.

Overall, mobile core network exposure is a sensitive topic among operators and it is difficult to obtain concrete feedback. Despite this, we feel that our research has helped some operators identify and reduce GTP exposures in their networks. We hope our work will continue to raise awareness of GTP security risks in the cellular community, and motivate more operators to respond proactively.

9. Related Work

There has been a growing body of research focused on security in cellular networks in recent years. Most of the existed studies focused on the Radio Access Network [29], [31], [43], [44], [45], [53], [54], [61], [73], [74], [82] and User Equipment [38], [46], [50], [51], [62], [84] side, as external researchers could easily obtain these devices and collect traffic data (by sniffing the air interface or capturing signaling on mobile devices). In comparison, fewer papers have explored security issues of the cellular core. Of these, the vast majority examine vulnerabilities in billing policies. For example, Peng et al. [65], [66] showed how non-billing services could be used as transmission channels to bypass billing. Go et al. [36] exploited TCP retransmissions to carry traffic data for free. Hong et al. [41] discovered billing flaws of Korean operators via measurement studies.

As one critical protocol of the cellular core, GTP has been proposed and deployed for several decades. Its lack of security mechanisms has been disclosed early on [69], but has not received adequate attention from either academia or industry, especially regarding the feasibility of attacks in real-world environments. A few industrial white papers [9], [11] have discussed GTP security, and demonstrated GTP-based attacks in operators' test networks. However, they only illustrate possible flaws by cases, lacking a comprehensive security analysis of GTP. Moreover, the feasibility and impact of these attacks in live cellular networks also remain unknown to the public. In contrast, we provide the first systematic analysis of the attack surface exposed by GTP in the real-world public Internet. Lutu et al. [58] analyzed network traffic including GTP signaling and tunneling data, but they focused on network performance and geo-deviation rather than security. Beyond GTP, several studies have examined security issues in other cellular core protocols. Holtmanns et al. [39] showed how vulnerabilities in the Diameter protocol could be exploited for SMS hijacking. Rao et al. [67] found the SS7-MAP protocol could be exploited to manipulate the validity of subscribers' phone numbers. A few other works have probed cellular network topology and performance bottlenecks from the public Internet. Traynor et al. [78] discussed how a scaled mobile-based botnet could impact core networks. Xu et al. [81] analyzed IP address allocation policies to infer details of US operators' core network architectures. Akon et al. [27] conducted a formal analysis of OAuth-based access control mechanisms in the 5G CN, delving into security flaws at the protocol design level.

In summary, existing research typically presumes that the core network is impervious to direct external attacks. For the first time, our study exposes the actual security vulnerabilities of the foundational core network protocol GTP, highlighting relevant security concerns to the community.

10. Conclusion

This paper is the first attempt to evaluate GTP security in real-world deployments. Our large-scale scanning experiment identified 749K valid GTP hosts across 162 countries accessible via the public Internet, highlighting possible lack of access control protection in mobile core networks. We found 38 types of GTP messages posing security threats such as node discovery, DoS, session hijacking, and user tracking, among others. We validated GTP-based attacks using open-source 4G and 5G projects in a lab environment. Notably, we demonstrated that an external attacker could even remotely hijack the downlink traffic of cellular users directly from the core network. We also discovered that GTP can be exploited for RDoS attacks due to protocol and implementation flaws, with even the simplest echo request messages achieving a maximum amplification factor of 170.7. Our findings underscore significant security risks from public exposure of GTP nodes, emphasizing the need for enhanced GTP security.

Acknowledgments

We thank our shepherd and all the anonymous reviewers for their valuable comments to improve this paper. We also thank the GSMA and relevant mobile operators for their valuable feedback. This work is supported in part by the National Natural Science Foundation of China (62302258). Yiming Zhang is supported by the Shuimu Tsinghua Scholar Program. Haixin Duan is supported by the Taishan Scholars Program.

References

[1] Cisco Bug: CSCus78987. https://bst.cisco.com/quickview/bug/CSCus78987.

[2] gtp — Junos OS — Juniper Networks. https://www.juniper.net/documentation/us/en/software/junos/cli-reference/topics/ref/statement/security-edit-gtp.html.

[3] GTPV1/V2 Echo Support for Peer MME and SGSN. https://www.cisco.com/c/en/us/td/docs/wireless/asr_5000/21-17_6-11/MME-Admin/21-17-MME-Admin/21-17-MME-Admin_chapter_0100110.html.

[4] IANA IPv4 Special-Purpose Address Registry. https://www.iana.org/assignments/iana-ipv4-special-registry/iana-ipv4-special-registry.xhtml.

[5] netlink library for Linux kernel GTP node. https://gitea.osmocom.org/cellular-infrastructure/libgtpnl.

[6] Scapy: the Python-based interactive packet manipulation program & library. https://github.com/secdev/scapy.

[7] The Linux kernel GTP tunneling module. https://docs.kernel.org/networking/gtp.html.

[8] TunnelTrap. https://github.com/nokia/TunnelTrap.

[9] Threat vector: GTP vulnerabilities in LTE and 5G networks 2020. https://www.politico.eu/wp-content/uploads/2020/06/POLITICO-Positive-Technologies-report-Threat-vector-GTP-June-2020.pdf, 2020.

[10] Shodan search engine. https://www.shodan.io/, 2022.

[11] GTP vulnerabilities: A cause for concern in 5G and LTE networks. https://secgen.com/SecurityGen-whitepaper-gtp-firewall.pdf, 2023.

[12] The Mobile Economy 2023. https://www.gsma.com/mobileeconomy/wp-content/uploads/2023/03/270223-The-Mobile-Economy-2023.pdf, 2023.

[13] free5GC. https://free5gc.org/, 2024.

[14] Open5GS. https://open5gs.org/, 2024.

[15] OPENAIR-CN. https://github.com/OPENAIRINTERFACE/openair-epc-fed, 2024.

[16] OpenAirInterface. https://openairinterface.org/, 2024.

[17] Openairinterface5g. https://github.com/OPENAIRINTERFACE/openairinterface5g, 2024.

[18] S-gw administration guide, staros release 21.27. https://www.cisco.com/c/en/us/td/docs/wireless/asr_5000/21-27/sgw-admin/21-27-sgw-admin/m_cscux62892-support-1-million-s1u-peers.html, 2024.

[19] srsRAN Project - Open Source RAN. https://www.srslte.com/, 2024.

[20] 3GPP. Charging Data Record (CDR) transfer. Technical Standard (TS) 32.295, 3rd Generation Partnership Project (3GPP), 08 2020. Version 16.0.0.

[21] 3GPP. GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface. Technical Standard (TS) 29.060, 3rd Generation Partnership Project (3GPP), 07 2020. Version 16.0.0.

[22] 3GPP. IP network layer security. Technical Standard (TS) 33.210, 3rd Generation Partnership Project (3GPP), 08 2020. Version 16.4.0.

[23] 3GPP. Evolved General Packet Radio Service (GPRS) Tunnelling Protocol for Control plane (GTPv2-C). Technical Standard (TS) 29.274, 3rd Generation Partnership Project (3GPP), 04 2021. Version 16.7.0.

[24] 3GPP. General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U). Technical Standard (TS) 29.281, 3rd Generation Partnership Project (3GPP), 04 2021. Version 16.2.0.

[25] 3GPP. Network Architecture. Technical Standard (TS) 23.002, 3rd Generation Partnership Project (3GPP), 05 2022. Version 17.0.0.

[26] Aftab Ahmad. *Wireless and mobile data networks*. John Wiley & Sons, 2005.

[27] Mujtahid Akon, Tianchang Yang, Yilu Dong, and Syed Rafiul Hussain. Formal analysis of access control mechanism of 5g core network. In Weizhi Meng, Christian Damsgaard Jensen, Cas Cremers, and Engin Kirda, editors, *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, CCS 2023, Copenhagen, Denmark, November 26-30, 2023*, pages 666–680. ACM, 2023.

[28] Harry Baldock. AT&T hands 5G core to Microsoft Azure. https://totaltele.com/att-hands-5g-core-to-microsoft-azure/, 2021.

[29] Evangelos Bitsikas and Christina Pöpper. Don't hand it over: Vulnerabilities in the handover procedure of cellular telecommunications. In *ACSAC '21: Annual Computer Security Applications Conference, Virtual Event, USA, December 6 - 10, 2021*, pages 900–915. ACM, 2021.

[30] Ludovico Cavedon, Christopher Kruegel, and Giovanni Vigna. Are BGP routers open to attack? an experiment. In Jan Camenisch, Valentin S. Kisimov, and Maria Dubovitskaya, editors, *Open Research Problems in Network Security - IFIP WG 11.4 International Workshop, iNetSec 2010, Sofia, Bulgaria, March 5-6, 2010, Revised Selected Papers*, volume 6555 of *Lecture Notes in Computer Science*, pages 88–103. Springer, 2010.

[31] Merlin Chlosta, David Rupprecht, Christina Pöpper, and Thorsten Holz. 5g suci-catchers: still catching them all? In Christina Pöpper, Mathy Vanhoef, Lejla Batina, and René Mayrhofer, editors, *WiSec '21: 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Abu Dhabi, United Arab Emirates, 28 June - 2 July, 2021*, pages 359–364. ACM, 2021.

[32] Merlin Chlosta, David Rupprecht, Christina Pöpper, and Thorsten Holz. 5g suci-catchers: still catching them all? In Christina Pöpper, Mathy Vanhoef, Lejla Batina, and René Mayrhofer, editors, *WiSec '21: 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Abu Dhabi, United Arab Emirates, 28 June - 2 July, 2021*, pages 359–364. ACM, 2021.

[33] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. A search engine backed by internet-wide scanning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, page 542–553, New York, NY, USA, 2015. Association for Computing Machinery.

[34] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast internet-wide scanning and its security applications. In *22nd USENIX Security Symposium (USENIX Security 13)*, pages 605–620, Washington, D.C., August 2013. USENIX Association.

[35] Victor Fajardo, Jari Arkko, John Loughney, and Glen Zorn. Diameter base protocol. *RFC*, 6733:1–152, 2012.

[36] Younghwan Go, Eunyoung Jeong, Jongil Won, Yongdae Kim, Denis Foo Kune, and KyoungSoo Park. Gaining control of cellular traffic accounting by spurious TCP retransmission. In *21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014*. The Internet Society, 2014.

[37] Linda Hardesty. Swisscom takes its time moving to 5G SA with AWS. https://www.fiercewireless.com/5g/swisscom-takes-its-time-moving-5g-sa-aws, 2023.

[38] Grant Hernandez, Marius Muench, Dominik Christian Maier, Alyssa Milburn, Shinjo Park, Tobias Scharnowski, Tyler Tucker, Patrick Traynor, and Kevin R. B. Butler. Firmwire: Transparent dynamic analysis for cellular baseband firmware. In *29th Annual Network and Distributed System Security Symposium, NDSS 2022, San Diego, California, USA, April 24-28, 2022*. The Internet Society, 2022.

[39] Silke Holtmanns, Siddharth Prakash Rao, and Ian Oliver. User location tracking attacks for LTE networks using the interworking functionality. In *2016 IFIP Networking Conference (IFIP Networking) and Workshops*, pages 315–322, 2016.

[40] Byeongdo Hong, Sangwook Bae, and Yongdae Kim. GUTI reallocation demystified: Cellular location tracking with changing temporary identifier. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society, 2018.

[41] Hyunwook Hong, Hongil Kim, Byeongdo Hong, Dongkwan Kim, Hyunwoo Choi, Eunkyu Lee, and Yongdae Kim. Pay as you want: Bypassing charging system in operational cellular networks. In *International Workshop on Information Security Applications*, pages 148–160. Springer, 2016.

[42] Syed Rafiul Hussain, Omar Chowdhury, Shagufta Mehnaz, and Elisa Bertino. Lteinspector: A systematic approach for adversarial testing of 4g LTE. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society, 2018.

[43] Syed Rafiul Hussain, Mitziu Echeverria, Omar Chowdhury, Ninghui Li, and Elisa Bertino. Privacy attacks to the 4g and 5g cellular paging protocols using side channel information. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society, 2019.

[44] Syed Rafiul Hussain, Mitziu Echeverria, Imtiaz Karim, Omar Chowdhury, and Elisa Bertino. 5greasoner: A property-directed security and privacy analysis framework for 5g cellular network protocol. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019*, pages 669–684. ACM, 2019.

[45] Syed Rafiul Hussain, Mitziu Echeverria, Ankush Singla, Omar Chowdhury, and Elisa Bertino. Insecure connection bootstrapping in cellular networks: the root of all evil. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, WiSec 2019, Miami, Florida, USA, May 15-17, 2019*, pages 1–11. ACM, 2019.

[46] Syed Rafiul Hussain, Imtiaz Karim, Abdullah Al Ishtiaq, Omar Chowdhury, and Elisa Bertino. Noncompliance as deviant behavior: An automated black-box noncompliance checker for 4g LTE cellular devices. In Yongdae Kim, Jong Kim, Giovanni Vigna, and Elaine Shi, editors, *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pages 1082–1099. ACM, 2021.

[47] Jana Iyengar and Martin Thomson. QUIC: A UDP-Based Multiplexed and Secure Transport. RFC 9000, May 2021.

[48] Erin Kenneally and David Dittrich. The menlo report: Ethical principles guiding information and communication technology research. *Available at SSRN 2445102*, 2012.

[49] Stephen Kent. IP Encapsulating Security Payload (ESP). (4303), December 2005.

[50] Eunsoo Kim, Min Woo Baek, CheolJun Park, Dongkwan Kim, Yongdae Kim, and Insu Yun. BASECOMP: A comparative analysis for integrity protection in cellular baseband software. In Joseph A. Calandrino and Carmela Troncoso, editors, *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, pages 3547–3563. USENIX Association, 2023.

[51] Eunsoo Kim, Dongkwan Kim, CheolJun Park, Insu Yun, and Yongdae Kim. Basespec: Comparative analysis of baseband software and cellular specifications for L3 protocols. In *28th Annual Network and Distributed System Security Symposium, NDSS 2021, virtually, February 21-25, 2021*. The Internet Society, 2021.

[52] Hongil Kim, Jiho Lee, Eunkyu Lee, and Yongdae Kim. Touching the untouchables: Dynamic security analysis of the LTE control plane. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pages 1153–1168. IEEE, 2019.

[53] Martin Kotuliak, Simon Erni, Patrick Leu, Marc Röschlin, and Srdjan Capkun. Ltrack: Stealthy tracking of mobile phones in LTE. In Kevin R. B. Butler and Kurt Thomas, editors, *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 1291–1306. USENIX Association, 2022.

[54] Nitya Lakshmanan, Nishant Budhdev, Min Suk Kang, Mun Choon Chan, and Jun Han. A stealthy location identification attack exploiting carrier aggregation in cellular networks. In Michael Bailey and Rachel Greenstadt, editors, *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, pages 3899–3916. USENIX Association, 2021.

[55] Ammar Latif, Ash Khamas, Sundeep Goswami, Vara Prasad Talari, and Dr Young Jung. Telco Meets AWS Cloud: Deploying DISH's 5G Network in AWS Cloud. https://aws.amazon.com/cn/blogs/industries/telco-meets-aws-cloud-deploying-dishs-5g-network-in-aws-cloud/, 2022.

[56] Xiang Li, Baojun Liu, Xiaofeng Zheng, Haixin Duan, Qi Li, and Youjun Huang. Fast IPv6 network periphery discovery and security implications. In *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 88–100, 2021.

[57] Zhenhua Li, Weiwei Wang, Christo Wilson, Jian Chen, Chen Qian, Taeho Jung, Lan Zhang, Kebin Liu, Xiangyang Li, and Yunhao Liu. Fbs-radar: Uncovering fake base stations at scale in the wild. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. The Internet Society, 2017.

[58] Andra Lutu, Diego Perino, Marcelo Bagnulo, and Fabián E. Bustamante. Insights from operating an IP exchange provider. In Fernando A. Kuipers and Matthew C. Caesar, editors, *ACM SIGCOMM 2021 Conference, Virtual Event, USA, August 23-27, 2021*, pages 718–730. ACM, 2021.

[59] Gordon Lyon. Nmap: the network mapper - free security scanner. https://nmap.org/, 2022.

[60] Foivos Michelinakis, Hossein Doroud, Abbas Razaghpanah, Andra Lutu, Narseo Vallina-Rodriguez, Phillipa Gill, and Joerg Widmer. The cloud that runs the mobile internet: A measurement study of mobile cloud services. In *2018 IEEE Conference on Computer Communications, INFOCOM 2018, Honolulu, HI, USA, April 16-19, 2018*, pages 1619–1627. IEEE, 2018.

[61] Shiyue Nie, Yiming Zhang, Tao Wan, Haixin Duan, and Song Li. Measuring the deployment of 5g security enhancement. In Murtuza Jadliwala, Yongdae Kim, and Alexandra Dmitrienko, editors, *WiSec '22: 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks, San Antonio, TX, USA, May 16 - 19, 2022*, pages 169–174. ACM, 2022.

[62] CheolJun Park, Sangwook Bae, Beomseok Oh, Jiho Lee, Eunkyu Lee, Insu Yun, and Yongdae Kim. Doltest: In-depth downlink negative testing framework for LTE devices. In Kevin R. B. Butler and Kurt Thomas, editors, *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 1325–1342. USENIX Association, 2022.

[63] Shinjo Park, Altaf Shaik, Ravishankar Borgaonkar, and Jean-Pierre Seifert. Anatomy of commercial imsi catchers and detectors. In *Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society*, WPES'19, page 74–86, New York, NY, USA, 2019. Association for Computing Machinery.

[64] Craig Partridge and Mark Allman. Ethical considerations in network measurement papers. *Communications of the ACM*, 59(10):58–64, 2016.

[65] Chunyi Peng, Chi-yu Li, Guan-Hua Tu, Songwu Lu, and Lixia Zhang. Mobile data charging: New attacks and countermeasures. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS '12, page 195–204, New York, NY, USA, 2012. Association for Computing Machinery.

[66] Chunyi Peng, Chi-Yu Li, Hongyi Wang, Guan-Hua Tu, and Songwu Lu. Real threats to your data bills: Security loopholes and defenses in mobile data charging. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, page 727–738, New York, NY, USA, 2014. Association for Computing Machinery.

[67] Siddharth Prakash Rao, Silke Holtmanns, Ian Oliver, and Tuomas Aura. Unblocking stolen mobile devices using SS7-MAP vulnerabilities: Exploiting the relationship between IMEI and IMSI for EIR access. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 1171–1176, 2015.

[68] Eric Rescorla and Nagendra Modadugu. Datagram Transport Layer Security Version 1.2. (6347), January 2012.

[69] Retains Full Rights. Global information assurance certification paper. *GIAC*, 2003.

[70] Jonathan D. Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan B. Johnston, Jon Peterson, Robert Sparks, Mark Handley, and Eve M. Schooler. SIP: Session initiation protocol. *RFC*, 3261:1–269, 2002.

[71] Christian Rossow. Amplification hell: Revisiting network protocols for ddos abuse. In *21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014*. The Internet Society, 2014.

[72] David Rupprecht, Kai Jansen, and Christina Pöpper. Putting LTE security functions to the test: A framework to evaluate implementation correctness. In *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, Austin, TX, August 2016. USENIX Association.

[73] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. Breaking LTE on layer two. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pages 1121–1136. IEEE, 2019.

[74] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. IMP4GT: impersonation attacks in 4g networks. In *27th Annual Network and Distributed System Security Symposium, NDSS 2020, San Diego, California, USA, February 23-26, 2020*. The Internet Society, 2020.

[75] Ankush Singla, Rouzbeh Behnia, Syed Rafiul Hussain, Attila A. Yavuz, and Elisa Bertino. Look before you leap: Secure connection bootstrapping for 5g networks to defend against fake base-stations. In Jiannong Cao, Man Ho Au, Zhiqiang Lin, and Moti Yung, editors, *ASIA CCS '21: ACM Asia Conference on Computer and Communications Security, Virtual Event, Hong Kong, June 7-11, 2021*, pages 501–515. ACM, 2021.

[76] Randall R. Stewart. Stream control transmission protocol. *RFC*, 4960:1–152, 2007.

[77] Joe Touch, Allison Mankin, and Ronald P. Bonica. The TCP authentication option. *RFC*, 5925:1–48, 2010.

[78] Patrick Traynor, Michael Lin, Machigar Ongtang, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta. On cellular botnets: Measuring the impact of malicious devices on a cellular network core. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, CCS '09, page 223–234, New York, NY, USA, 2009. Association for Computing Machinery.

[79] Guan-Hua Tu, Chi-Yu Li, Chunyi Peng, Yuanjie Li, and Songwu Lu. New security threats caused by ims-based SMS service in 4g LTE networks. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 1118–1130. ACM, 2016.

[80] Paul Vixie, Olafur Gudmundsson, Donald E. Eastlake III, and Brian Wellington. Secret key transaction authentication for DNS (TSIG). *RFC*, 2845:1–15, 2000.

[81] Qiang Xu, Junxian Huang, Zhaoguang Wang, Feng Qian, Alexandre Gerber, and Zhuoqing Morley Mao. Cellular data network infrastructure characterization and implication on mobile content placement. In *Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '11, page 317–328, New York, NY, USA, 2011. Association for Computing Machinery.

[82] Hojoon Yang, Sangwook Bae, Mincheol Son, Hongil Kim, Song Min Kim, and Yongdae Kim. Hiding in plain signal: Physical signal overshadowing attack on LTE. In Nadia Heninger and Patrick Traynor, editors, *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, pages 55–72. USENIX Association, 2019.

[83] Yiming Zhang, Baojun Liu, Chaoyi Lu, Zhou Li, Haixin Duan, Shuang Hao, Mingxuan Liu, Ying Liu, Dong Wang, and Qiang Li. Lies in the air: Characterizing fake-base-station spam ecosystem in china. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, pages 521–534. ACM, 2020.

[84] Jinghao Zhao, Boyan Ding, Yunqi Guo, Zhaowei Tan, and Songwu Lu. Securesim: rethinking authentication and access control for sim/esim. In *ACM MobiCom '21: The 27th Annual International Conference on Mobile Computing and Networking, New Orleans, Louisiana, USA, October 25-29, 2021*, pages 451–464. ACM, 2021.

[85] Zhou Zhuang, Xiaoyu Ji, Taimin Zhang, Juchuan Zhang, Wenyuan Xu, Zhenhua Li, and Yunhao Liu. Fbsleuth: Fake base station forensics via radio frequency fingerprinting. In Jong Kim, Gail-Joon Ahn, Seungjoo Kim, Yongdae Kim, Javier López, and Taesoo Kim, editors, *Proceedings of the 2018 on Asia Conference on Computer and Communications Security, AsiaCCS 2018, Incheon, Republic of Korea, June 04-08, 2018*, pages 261–272. ACM, 2018.

TABLE 8: Message types and functions of GTP.

Message Type	GTPv1-C	GTPv2-C	GTP-U	Function	Example
Path Management	4	3	3	Monitor the health of the transport between GTP peers	*Echo Request*
Tunnel Management	12	32	2	Create, modify and delete GTP tunnels	*Create Session Request*
Location Management	6	-	-	Handle location info (when MAP is not supported)	*Note MS GPRS Present Request*
Mobility Management	16	22	-	Manage and control the mobility of mobile devices	*Detach Notification*
MBMS	20	6	-	Support Multimedia Broadcast/Multicast Service	*MBMS Session Start Request*
MS Info Change Reporting	2	-	-	Support MS Info Change Reporting mechanism	*MS Info Change Notification Request*
CS Fallback and SRVCC related	-	9	-	Support voice services via CSFB and SRVCC	*CS Paging Indication*
Non-3GPP Access Related	-	2	-	Handle messages related to non-3GPP access	*Create Forwarding Tunnel Request*
Restoration and Recovery	-	8	-	Handle network failures and restore services	*PGW Restart Notification*
Trace Management	-	2	-	Active or Deactive session trace	*Trace Session Activation*
G-PDU Message	-	-	1	Transfer user data in the GTP tunnel	*G-PDU*

TABLE 9: Implementation of GTP messages in open-source projects.

Network	Project	Version	Network Elements	GTP-U (supported/all)	Supported Message Type
4G	Open5GS EPC	v2.6.6 [14]	SGWU, PGWU	5/6	1-2,26,254-255
	OAI-4G	v1.2.0 [15]	S-GW, P-GW	6/6	1-2,26,31,254-255
	srsRAN	v23.04 [19]	SPGW	6/6	1-2,26,31,254-255
5G	OAI-5G	v1.5.1 [17]	UPF	3/6	1-2,255
	free5GC	v3.3.0 [13]	UPF	1/6	255
	Open5GS 5G Core	v2.6.6 [14]	UPF	3/6	1-2,255
Network	**Project**	**Version**	**Network Elements**	**GTPv2-C (supported/all)**	**Support Message Type**
4G	Open5GS EPC	v2.6.6 [14]	SGWC, PGWC, MME	24/84	1,2,32,33-37,68-69,95-100,166-171,176-177
	OAI-4G	v1.2.0 [15]	S-GW, P-GW, MME	12/84	1,2,32-37,170-171,176-177
	srsRAN	v23.04 [19]	S-GW, P-GW, MME	9/84	32-36,70,170,176-177

Appendix A.
GTP Message Categories

We enumerate GTPv1-C, GTPv2-C, and GTP-U messages types through Table 8. These 3 protocols maintain a relatively independent structure in defining specific message types and functions. The majority of messages differ significantly with only a few such as echo requests/responses.

Appendix B.
GTP Implementation in Open-source Projects

To validate proposed GTP attacks, we investigated the support for GTP messages in mainstream open-source projects, as shown in Table 9. Considering that 3G has entered the phase of network retirement, our investigation focuses on core network open-source projects for 4G and 5G by manual code review. As shown in Table 9, all projects have implemented the GTP-U G-PDU message, as it is the basic tunneling message that directly carries user data packets. However, for the control plane GTP-C, existing implements have inadequate support. As GTP is mainly applicable to "mobile" scenarios, the inadequate implementation of open-source projects is reasonable as they are mainly used in lab environments with few UEs linked to fixed base stations, where mobility management such as base station switching could hardly be required.

Appendix C.
Meta-Review

The following meta-review was prepared by the program committee for the 2025 IEEE Symposium on Security and Privacy (S&P) as part of the review process as detailed in the call for papers.

C.1. Summary

The paper assesses the security of GTP practices in real-world cellular networks. It presents a measurement study of GTP-enabled IPv4 addresses. It also provides a security analysis of open-source cellular stacks emphasizing the potential risks associated with an exposed GTP service on the Internet.

C.2. Scientific Contributions

- Independent Confirmation of Important Results with Limited Prior Research
- Identifies an Impactful Vulnerability
- Provides a Valuable Step Forward in an Established Field

C.3. Reasons for Acceptance

1) The paper quantifies GTP-enabled services that are exposed to the Internet (which are likely part of core cellular networks) and highlights the potential exposure of core cellular infrastructure to attackers in the wild.
2) Using testbeds, the paper identifies potential attack vectors that can enable a threat actor to exploit exposed GTP hosts to disrupt or degrade cellular services.

C.4. Noteworthy Concerns

1) The scanning techniques and downstream analysis are unclear: Shodan and scan results from different time ranges are compared, possibly contributing to false negatives.
2) The details around removing "honeypot" hosts from scanning results are unclear.
3) Although the paper identifies numerous GTP-enabled Internet-facing IPs, it does not differentiate whether these endpoints are merely exposed or exploitable. It also does not explore commercial products/services that may have additional protection to mitigate harm despite exposing a GTP service to the Internet.
4) Some potential attack scenarios uncovered using the testbed may not translate to real-world networks. In some scenarios, the attacker's capabilities might be overstated and seem unrealistic.