Uncovering Security Vulnerabilities in Real-world Implementation and Deployment of 5G Messaging Services

Yaru Yang
yyr22@mails.tsinghua.edu.cn
Tsinghua University
Beijing, China

Yiming Zhang✉
zhangyiming@tsinghua.edu.cn
Tsinghua University
Beijing, China

Tao Wan
t.wan@cablelabs.com
CableLabs
Louisville, United States
Carleton University
Ottawa, Canada

Chuhan Wang
wch22@mails.tsinghua.edu.cn
Tsinghua University
Beijing, China

Haixin Duan✉
duanhx@tsinghua.edu.cn
Tsinghua University
Beijing, China
Quancheng Laboratory
Jinan, China

Jianjun Chen
jianjun@tsinghua.edu.cn
Tsinghua University
Beijing, China
Zhongguancun Laboratory
Beijing, China

Yishen Li
li-ys20@mails.tsinghua.edu.cn
Tsinghua University
Beijing, China

ABSTRACT

5G messaging services, based on Global System for Mobile Communications Association (GSMA) Rich Communication Service (RCS) and 3rd Generation Partnership Project (3GPP) IP Multimedia Subsystem (IMS), have been deployed globally by more than 90 mobile operators serving over 421 million monthly active users via 1.2 billion devices. Despite the widespread use, security research of 5G messaging remains sparse. In this paper, we present a comprehensive security analysis and measurement of 5G messaging services, assisted by a semi-automated testing tool we developed. We considered both carrier-side deployment and phone-side software implementations by testing against three large operators, each with hundreds of millions of subscribers, and six popular 5G messaging-enabled devices. We uncovered 4 categories of vulnerabilities, allowing for a wide range of attacks, including Man-In-The-Middle (MITM) attacks, zero-click remote information leakage, phone storage exhaustion and mobile data consumption, and Denial-of-Services (DoS) attacks. Our study underscores the need for further security enhancements in security specifications, implementation, and deployment of 5G messaging services.

CCS CONCEPTS

• **Security and privacy → Mobile and wireless security**.

WiSec '24, May 27–30, 2024, Seoul, Republic of Korea

ACM ISBN 979-8-4007-0582-3/24/05.
https://doi.org/10.1145/3643833.3656131

KEYWORDS

5G Messaging, Rich Communication Service

ACM Reference Format:
Yaru Yang, Yiming Zhang✉, Tao Wan, Chuhan Wang, Haixin Duan✉, Jianjun Chen, and Yishen Li. 2024. Uncovering Security Vulnerabilities in Real-world Implementation and Deployment of 5G Messaging Services. In *Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '24), May 27–30, 2024, Seoul, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3643833.3656131

1 INTRODUCTION

First introduced 30 years ago, Short Message Service (SMS) continues to be popular. Over the years, SMS technology and functions have evolved significantly, transitioning from basic text transmission to supporting multimedia content. Especially, the advent of Rich Communication Services (RCS) has notably expanded SMS functionality. RCS not only facilitates high-quality image and video transmission, surpassing traditional SMS capabilities, but also introduces features like group chats and video calls. As a result of these advancements, SMS has found extensive use in diverse fields such as e-commerce and banking applications.

GSMA has specified RCS as a mandatory feature for 5G terminals [11]. Its rich functionalities and promising commercial prospects also have led to substantial support and promotion from mobile operators worldwide. For example, several large mobile operators jointly released the "The 5G Messaging Service" whitepaper in 2020 [13], demonstrating their strong commitment to the deployment of RCS-based 5G messaging. As of 2022, RCS has been adopted by more than 90 mobile operators in 60 countries, serving over 421 million monthly active users via 1.2 billion devices [10].

The SMS system has been found to have numerous security vulnerabilities, as highlighted in prior studies [20, 21, 26, 31, 35].

Particularly noteworthy is the fact that the introduction of new SMS mechanisms and architectures can potentially open up new avenues for attacks. For instance, the transition from 3G to 4G, which involved a shift from circuit-switched domain to packet-switched domain, presented new security challenges. In 2016, Tu et al. [42] identified various new security threats in the IP Multimedia Subsystem (IMS)-based SMS, including spoofing and data injection, which could be exploited, e.g., for fraud. Therefore, a natural question arises: *what are the potential security pitfalls in 5G messaging, which is also based on IMS?*

So far, only Zhao et al. [46] have studied the security issues in 5G messaging, by primarily exploring One Time Password (OTP) based subscriber authentication procedure in RCS deployments by four US operators. However, OTP is not the only subscriber authentication procedure used in RCS, and it may not be used by other operators. For instance, our investigation found that the three operators we tested use Authentication and Key Agreement (AKA) rather than OTP for authentication, suggesting the possibility of other security issues. In addition, the overall process involved in 5G messaging is quite extensive. Many other procedures besides user authentication are closely related to security and worth analyzing. Nevertheless, previous discovery of security issues is dependent on manual efforts without the assistance of automated tools.

This work delves into an in-depth security analysis of 5G messaging services, with a focus on their deployment by mobile operators (carrier-side) and the software implementation on end devices (phone-side). Our methodology starts with an examination of relevant standards [3, 12, 15] and network traffic generated between actual 5G messaging applications and operators, providing insights into real-world 5G messaging operations. Based on the standards and its actual deployment, leveraging our security expertise, we propose the threat model and potential vulnerabilities of 5G messaging at two aspects: cryptographic protocols and application layer protocols. Then we designed Sipano, a semi-automated tool, to systematically detect and confirm potential vulnerabilities in real-world 5G messaging services. Our tests covered three large mobile operators (OP-I, OP-II and OP-III, from the same country), each serving hundreds of millions of users, and six popular 5G messaging-enabled devices (Xiaomi, Redmi, Huawei, Honor, ZTE, and Samsung).

Our findings reveal numerous unreported security flaws in practical 5G messaging systems, briefly summarized below (see details in Table 1). On the carrier side, we found that all three operators do not comply with the security requirement in IMS security specification [3] on protecting the interface between the User Equipment (UE) and the Proxy Call Session Control Function (P-CSCF) (i.e., the Gm interface defined by 3GPP TS 23.002 [1]). More specifically, when 5G UE uses the cellular network of any of the three operators for 5G messaging services, the integrity of SIP signaling channel between the UE and P-CSCF is not protected with IPsec Encapsulating Security Payload (ESP), as required by [3]. On the end device side, we also uncovered security issues of 5G messaging implementations in all the phone models we tested. For example, while Transport Layer Security (TLS) is used to protect 5G messaging traffic between UE and P-CSCF under Wi-Fi, we found that devices from Xiaomi, Redmi, and ZTE do not implement TLS certificate validation properly, allowing MITM attacks. We also identified other issues that allow various attacks including zero-click remote information leakage, phone storage and data usage exhaustion, and Denial-of-Services (DoS) attacks towards an arbitrary server or the messaging applications on devices from Samsung, Xiaomi, Redmi, and ZTE. All the attacks are validated in our lab environment (with controlled devices). We have reported all vulnerabilities to affected mobile vendors and the national CERT agency, and received their confirmation. We also sent reports to GSMA to discuss the mitigations. Mitigation of V3 by updating TS 33.203, as recommended by GSMA based on our findings, has been approved by 3GPP[4].

In summary, our study reveals a landscape of security vulnerabilities and potential attacks in real-world 5G messaging deployment, highlighting the urgent need for further security improvements in specifications, implementation, and deployment. Our main contributions are summarized as follows:

- We conducted an in-depth security assessment of the deployment and implementation of 5G messaging, and proposed a threat model and potential security issues, including protocol design flaws, non-compliant deployments and implementation vulnerabilities. We also developed a semi-automated detection tool, Sipano, to detect and verify potential vulnerabilities, which will be shared with other researchers upon request to facilitate future research.
- Using Sipano, we tested 3 large mobile operators, and 6 popular smartphone brands and identified 4 categories of previously unreported security vulnerabilities. We conducted 5 attacks against real-world 5G messaging systems, demonstrating serious risks to end devices and user privacy. Our study also provides root cause analysis and mitigation strategies, including security enhancements of specifications.

2 BACKGROUND

2.1 Messaging in the 5G Era

Rich Communication Services (RCS), the foundation of 5G messaging, was originally introduced in 2007 to facilitate multimedia messaging interoperability between mobile operators. It is standardized by the Global System for Mobile Communications Association (GSMA) and has undergone significant development over the years. RCS has gained extensive operator support with the global expansion of Long-Term Evolution (LTE) networks post-2008, especially the deployment of IP Multimedia Subsystem (IMS). To boost further global development and inter-operator connectivity, GSMA launched the RCS Universal Profile (UP) as the standard in 2016, and has since released multiple updates.

Currently, 5G already has RCS as an official implementation option for messaging. With its comprehensive feature support (e.g., multimedia transmission, group chat, location push) and straightforward implementation (native phone support without additional apps), 5G messaging has received strong support from mobile operators and phone vendors. In April 2020, several large operators jointly unveiled the whitepaper *The 5G Messaging Service* [13]. This initiative received endorsement from more than 12 leading cellphone vendors, including but not limited to Huawei, Xiaomi, ZTE, and Samsung. Note that in the following content of this paper, the term "5G messaging" could also refer to "RCS".

Table 1: Summary of key findings on 5G messaging vulnerabilities, impacted vendors, and verified attacks. In this work, we tested a total of 3 popular 5G mobile operators (OP-I, OP-II and OP-III), and 6 brands of popular cellphones that support 5G messaging (Xiaomi, Redmi, ZTE, Huawei, Hornor and Samsung). * indicates that V2 can be used to facilitate DoS but is not required.

Category	Vulnerability	Vendors (Impacted)	Attack (Verified in this work)				
			MITM	Info Leak	DDoS @Server	Storage/Traffic Consumption	DoS @SMS App
Carrier Issues	V1: Unprotected RCS traffic using cellular access	OP-I, OP-II, OP-III	✔				
	V2: Tamperable URL of files to be sent	OP-I, OP-II, OP-III		✔	✔	✔	✔*
Cellphone Issues	V3: Incorrect handling of TLS verification	Xiaomi, Redmi, ZTE	✔				
	V4: Crash-inducing security issues	Samsung, ZTE, Xiaomi, Redmi					✔

The base architecture of RCS is IP Multimedia System (IMS) [2], which includes different types of Call Session Control Functions (CSCF). Of particular interest is a Proxy CSCF (P-CSCF), which interfaces with User Equipment (UE) and other back-end CSCFs and gateways to facilitate end-to-end rich communications. The Session Initiation Protocol (SIP) [34], serving as the control protocol for IMS, is a multimedia communication protocol developed by the Internet Engineering Task Force (IETF). SIP messages can be transmitted over any of the three transport layer protocols TCP [30], UDP [29] and SCTP [41]. When Transport Layer Security (TLS) is required to protect SIP signaling messages, Session Initiation Protocol Secure (SIPS) URI [5] scheme should be used. In this case, SIP server is authenticated by SIP User Agent (UA) using public key certificate, and the UA can be authenticated by the SIP server using either a client certificate or SIP digest.

IMS security is specified in 3GPP TS 33.203 [3]. Of our interest is the security protection between SIP UA on the UE (referred to as UE for simplicity) and P-CSCF (referred to as Gm interface in 3GPP), including authentication of UE by P-CSCF (namely UE authentication) and SIP signaling protection. For cellular access (referred to as 3GPP access), Authentication and Key Agreement (AKA) protocol, namely IMS-AKA [3], is used for mutual authentication between UE and P-CSCF. At the end of the authentication, a shared secret is derived by both UE and P-CSCF to enable the use of IPsec ESP for protecting SIP signaling. For non-cellular access (e.g., Wi-FI), referred to as non-3GPP access, SIP Digest [37] is used by P-CSCF to authenticate UE and TLS is used for SIP signaling protection.

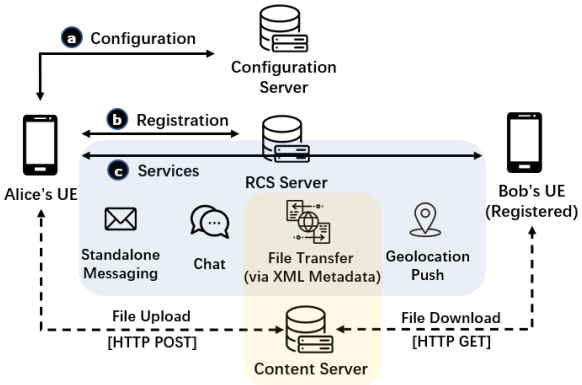

Figure 1: Process of utilizing 5G messaging services

2.2 Architecture and Workflow of RCS

Figure 1 illustrates the workflow of 5G messaging (RCS) via a case, where Alice wants to send a 5G message to Bob. The steps she needs to take include:

a) Configuration Process. Alice's UE communicates with the Configuration Server to configure various functionalities and settings to meet the specific requirements of the service provider and end users. This process is done by fetching an Extensible Markup Language (XML) file (referred to as configuration XML) from Configuration Server via Hypertext Transfer Protocol Secure (HTTPS).

b) Registration Process. Alice interfaces with the RCS Server (consisting of P-CSCF and back-end authentication network functions) to complete the registration process. During this process, her identity is authenticated using either SIP Digest [37] or IMS-AKA methods [3]. Note that the primary processes of IMS-AKA and SIP Digest are similar. Both methods use the same headers as HTTP Digest [38] to carry challenge and response values. In the SIP Digest method, the UE obtains a username and password during the configuration process, which is then used to generate a response. Conversely, in the IMS-AKA method, the UE utilizes the IP Multimedia Services Identity Module (ISIM) within the Universal Integrated Circuit Card (UICC) to generate a response.

c) Services Process. After registration, Alice's UE can use various RCS services, such as standalone messaging, chat, file transfer, and geolocation push. Each service may employ distinct protocols and mechanisms. The standalone messaging services opt for either pager mode (utilizing SIP MESSAGE) or large message mode [14] (using Message Session Replay Protocol [19], MSRP, instead). The chat service also employs MSRP. For file transfer, Alice's UE initially uploads the file to the Content Server through an HTTP POST request. Then the 5G message sent to Bob would include file download information in XML format, allowing Bob to access and download the file from the Content Server. The geolocation push feature facilitates location sharing by sending geolocation data in XML format in the message. Additionally, a fallback mechanism, known as the "geo" URI (Uniform Resource Identifier) [40], can be used for geolocation sharing. This involves directly sending a "geo" URI as the message content (e.g., "geo:12.3456789,21.9876543;u=10;rcs-l=A%20Place%20Name"). The "u" (uncertainty) parameter in the "geo" URI specifies the degree of uncertainty in meters.

Table 2: The configuration properties of targeting operators and the standard. * indicates that encryption is just recommended in the standard, but not mandatory.

Property	OP-I | OP-II | OP-III	Standard
Port	5460 (cellular access) 5260 (Wi-Fi access)	Not Specified
Authentication	AKA	AKA, SIP Digest
Cryptography	None (cellular access) TLS (Wi-Fi access)	IPsec, TLS*

3 METHODOLOGY

This section outlines our method for designing 5G messaging security test experiments. First, we measured real-world commercial 5G messaging services to understand specific configurations. Then we proposed a concrete threat model and systematically analyzed the potential cryptographic and application layer security risks of 5G messaging. Based on these identified threats, we developed the semi-automated testing tool Sipano, detailed in Section 4, to facilitate experimentation.

3.1 Measurement of Specific Configurations

The standards of 5G messaging are mainly defined by RCC.07 [15] and Universal Profile (UP) 2.4 [12], which are the primary references for security testing. The standards leave several key deployment configurations undefined or open to alternatives. For example, the authentication method for 5G messaging is defined in RCC.07 [15], and the options include IMS AKA as well as SIP Digest, which are chosen by the operator. Therefore, to enable security test experiments, we first captured and analyzed real-world 5G messaging traffic to measure key operator configurations in commercial networks.

Our research scope covered three popular operators, anonymized as Operator I (OP-I), Operator II (OP-II), and Operator III (OP-III). The configuration properties we concerned with are listed in Table 2, including the services port, authentication mechanism and protection methods used by 5G messaging. We inspected 5G messaging traffic with rooted Android phones, capturing traffic with tcpdump and forwarding it via ADB to Wireshark on laptops.

As shown in Table 2, the key configurations of 3 operators are consistent. Overall, we find their configurations differ between cellular and Wi-Fi access networks. For cellular access, port 5460 is used with no protection, whereas for Wi-Fi access, port 5260 is used with TLS. Besides, all three operators employ AKAv1-MD5 as the authentication algorithm.

3.2 Threat Model

Our threat model assumes that an attacker possesses a 5G messaging-enabled phone, and can manipulate the content of messaging packets, i.e., to control the phone's 5G messaging traffic. The victim is another 5G messaging user, and the attacker only needs the victim's phone number. Additionally, a Man-In-The-Middle (MITM) attacker requires the ability to monitor and modify the victim's traffic using methods discussed in Section 6.1.

3.3 Security Analysis

We aim to contemplate the potential security risks faced by 5G messaging from two perspectives: the cryptographic protocols below the application layer (i.e., IPsec and TLS), and the protocols within the application layer (including SIP, MSRP, etc.). The objects considered include both RCS servers and clients.

3.3.1 Cryptographic Protocols. Since unprotected or weakly protected transmissions would enable attacks akin to MITM, we first evaluate the cryptographic protocols used by 5G messaging services. In the configuration process, TLS is used. In the registration and services processes, according to our measurement results, the cellular ports (5460) of all three tested operators are unprotected, and the Wi-Fi ports (5260) of them are protected by TLS. Therefore, at the cryptographic protocol level, we only need to further validate: whether TLS is correctly and securely deployed (e.g., the correctness of certificate validation).

3.3.2 Application Layer Protocols. The three primary steps of 5G messaging include configuration, registration and services (see Figure 1). Given that the configuration process primarily involves the UE retrieving a configuration XML via HTTPS, our chief concern for application layer protocol security centers on the registration and services phases.

Registration. Following the traffic analysis in Section 3.1, the three operators we tested employ AKAv1-MD5 algorithm during the registration process. It is worth validating if this authentication process is correctly implemented, specifically whether the 5G message server can accurately verify identity consistency during authentication. Two SIP headers are critical for authentication, `From` and `P-Preferred-Identity`. We can test the security of authentication by modifying these headers during registration. It's noteworthy that, to avoid causing harm to the operator servers, we will only replace these header values with another legitimate but theoretically unauthenticated identity.

Services. According to RCC.07 [15], 5G messaging services provide standalone messaging, chat, file transfer (via SMS), geo-location PUSH (via SMS), and chatbot functions. Since chat and chatbot services have not been fully deployed by the operators tested, we do not consider related security issues in this paper. Nonetheless, the MSRP protocol, integral to the chat service, is also employed in the large message mode of the standalone messaging service, which has been deployed. Consequently, we are still able to conduct tests on MSRP. We have run RCS services on the tested phones across all three operators and captured the traffic. Table 3 presents the potential targets we identified as critical to security, along with the selected fields for testing. The testing strategy is divided into three categories: 1) Illegality: changing a field to an illegal value, 2) Legitimate Alteration: changing fields to other legal values, and 3) Out-of-Bounds: testing situations where numeric-type fields go beyond reasonable ranges (e.g., negative values). The first and second strategies are applied to all the targeted fields, and the third strategy (Out-of-Bounds) is only applied to the numeric fields. It is worth noting that to avoid harming operator servers, the malformed packets for testing are categorized into two types: 1) packets that we believe may harm the server, which will only be sent to our controlled Local UE (without going through the server), and

Table 3: List of the tested fields. * indicates that this field is additionally subject to Out-of-Bounds testing strategy.

Service	Tested Protocol/Format	Tested Fields
Standalone Messaging (pager mode)	SIP	From, To, Contact, Call-ID, CSeq, Expires, Allow, Supported, P-Access-Network-Info, Via, User-Agent, Content-Type, Server, P-Preferred-Identity, Route, Accept-Contact, Max-Forwards*, Date*, Content-Length*
Standalone Messaging (pager mode)	CPIM	From, To, Content-Type, NS, imdn.Disposition-Notification, Message Content, DateTime*, Content-Length*
Standalone Messaging (large message mode)	MSRP	To-Path, From-Path, Message-ID, Success-Report, Content-Type, Byte-Range*
File Transfer	XML content	file-info, file-hash-algorithm, file-hash-value, content-type, file-name, data url, file-size*, until*, am:playing-length(only for audio)
Geolocation PUSH	Geolocation Push URI (for fallback)	rcs-l, crs, coordinate*, u*

2) packets deemed harmless to the server, which may be sent to Remote UE through the server.

4 DESIGN AND IMPLEMENTATION OF SIPANO

This section describes the design and implementation of Sipano, a system designed to investigate and exploit vulnerabilities in the 5G messaging service. Sipano primarily includes five parts: 1) `SipMITM`, to control 5G messaging traffic for testing, 2) `Payload Generator`, to automatically generate testing payloads, 3) `Message Constructor`, to construct complete SIP and MSRP messages based on payloads, 4) `Log Generator`, to collect information and generate useful logs and 5) `False Content Server`, to test the File Transfer functionality. The following will detail the design and implementation.

SipMITM. `SipMITM` is the core component of Sipano, which enables MITM attacks to control RCS traffic and uses messages generated by `Message Constructor` for testing. The MITM attack initiated by SipMITM is based on vulnerabilities identified in our analysis of cryptographic protocols. MITM attack in this work between UE and RCS Server will be detailed in Section 6.1.

Payload Generator. `Payload Generator` is a template-based payload creator. Within the template, specific placeholders can be employed to depict a particular scheme for string generation. Additionally, certain symbols can be used to denote simple mutations on a string (e.g., random insertions/deletions of characters). Testers have the flexibility to use Augmented Backus-Naur Form (ABNF) syntax to customize the specific placeholders for string generation. It also supports dynamic variable filling, which can automatically fill fields that can only be determined during actual testing (such as the recipient's phone number).

Message Constructor. `Message Constructor` is used to generate complete SIP and MSRP messages, allowing `Payload Generator` to focus on the fields that need to be tested.

Log Generator. `Log Generator` collects and organizes information from `SipMITM` and UEs, such as crashes, outputting logs of various levels for subsequent analysis.

False Content Server. `False Content Server` is an HTTPS server based on Python library `http.server`. Its primary purpose is to test File Transfer services, evaluating whether the file URL is susceptible to tampering. If tampering is feasible, `False Content Server` can replace the operator's content server, allowing the attacker to control downloaded files. This also avoids detrimental impact on the operator's servers by hosting malicious files (e.g. oversized files) on `False Content Server` instead. Besides, this tool can generate arbitrary-sized files or images to facilitate testing.

Figure 2 illustrates the complete testing procedure. We first deployed Sipano on a laptop, acting as a man-in-the-middle between UEs and the RCS Server. Based on security analysis results, we manually created test templates and fed them to the `Payload Generator`. Then, the `SipMITM` controlling RCS traffic of `Local UE` fed payloads to `Message Constructor` to construct complete SIP/MSRP messages, and sent them to the Local UE or the Remote UE. Note that the distinction between the Local/Remote UE is that the messages sent to the latter need to pass through the RCS Server and thus may be subjected to validation. Besides, for file type messages, the receiver UE would request the Content Server (the operator's Content Server or our implemented `False Content Server`) to download files. Finally, we identified potential security issues by analyzing logs generated by `Log Generator`. It is worth noting that this experimental framework can be used to test both the UE and the RCS Server. When testing the UE, if it is not vulnerable to MITM attacks, it can still be tested by placing it as the Remote UE and not deploying SipMITM for it.

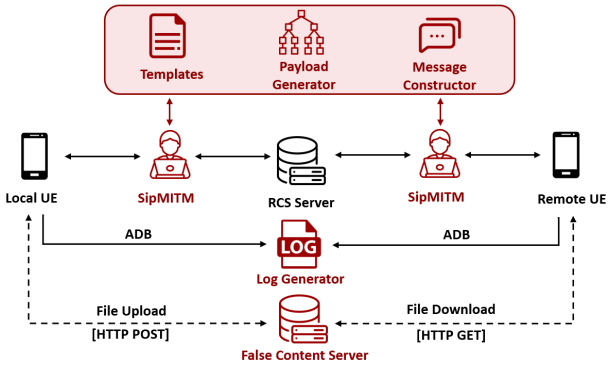

Figure 2: Experimentation framework

The design and implementation of Sipano empower us to semi-automatically conduct experiments aimed at unearthing potential security issues within the 5G messaging service. Specifically, on the service side, we tested three large operators. While on the UE side, we selected six popular mobile phone brands (models) with 5G messaging enabled, i.e., Samsung (A53 5G), Huawei (Mate 30 5G), Honor (50), ZTE (S30), Xiaomi (12) and Redmi (Note 9 5G). Detailed findings from these tests and our subsequent analyses are presented in the following section.

5 SECURITY ISSUES

Our experimental findings reveal security vulnerabilities across the 5G messaging system at various levels, including deployment flaws (V1: Unprotected RCS Traffic Using Cellular Access), protocol design flaws (V2: URL Tampering When Sending Files), and implementation flaws (V3: TLS Validation is not Handled Correctly, V4: Crash-inducing Security Issues). We will detail these security issues in the following.

5.1 V1. Unprotected RCS Traffic Using Cellular Access

Description. According to the IMS security standards [3], the integrity of SIP signaling channel between a UE and the P-CSCF needs to be protected by IPsec when the UE uses cellular access. However, our tests showed that all three operators we tested, OP-I, OP-II and OP-III, use publicly accessible servers on the Internet to serve 5G messaging. A UE using cellular access directly establishes SIP channels with the server on 5460 port, without any integrity or confidentiality protection. Thus the Gm interface between the UE and the P-CSCF has not been protected as required by the standards [3], which is a vulnerability we denote as V1.

Security Risks. The flawed deployment of operators introduces considerable security risks to 5G messaging. The lack of confidentiality protection in RCS traffic allows attackers to intercept 5G messages by monitoring signaling data through the air interface. Previous studies [28] confirm that, current commercial networks lack encryption on user-plane data over the air interface, facilitating the eavesdropping attack. Furthermore, the absence of integrity protection enables adversaries to conduct MITM attacks on RCS traffic, potentially resulting in fraudulent and spamming activities. Although executing MITM attacks directly on the air interface is challenging [33], we found their feasibility increases if users employ Virtual Private Network (VPN). VPN is now a popular choice for Internet users to protect privacy [22], and has been used by over 31% of Internet users [18]. Our research found that under default settings, RCS traffic could be directly forwarded through the VPN upon cellular network connection. This unprotected RCS traffic is then susceptible to MITM attacks, especially if the VPN service node is compromised or if users install malicious VPN software under the control of an attacker.

Impact. We conducted detailed tests on the 6 brands of phones to examine their specific behaviors. Firstly, all the RCS traffic from these mobile phones connected to OP-I, OP-II and OP-III using cellular access is not protected. Secondly, if a VPN is enabled on the user terminal: 1) Xiaomi, Redmi, ZTE, Huawei and Honor would forward 5G message traffic to the VPN. This prevents MITM attacks in cellular networks (e.g., FBS) but discloses the unprotected traffic to the VPN, which would be malicious or compromised. 2) Samsung phones forcefully prevent 5G message traffic from being forwarded to the VPN. This avoids potential MITM attacks launched by VPN nodes but may expose the RCS traffic to attacks in cellular networks.

5.2 V2. URL Tampering When Sending Files

Description. Upon thorough examination, we discovered that operators may not validate the integrity of the XML metadata in the SIP MESSAGE request. As a result, an attacker could arbitrarily modify the URL in the XML, directing the recipient to access and download a file from any URL. Both image and file transfers in RCS-based 5G messaging follow the same process, with image messages allowing the transmission of a thumbnail and a full image. Further analysis revealed that an attacker could replace the URL of the thumbnail with a malicious URL. Consequently, the recipient's mobile device would automatically access this malicious URL upon receiving the message, even without any action from the victim.

Security Risks. This vulnerability can lead to several serious security threats impacting mobile operators' servers, end-user devices, and user privacy. Firstly, an attacker could send a manipulated XML file to multiple users, tricking them into initiating a (large) file download, thereby facilitating a reflection amplification DDoS attack on any targeted HTTP server. Secondly, this vulnerability could be exploited to consume substantial data from the recipient, leading to mobile data billing fraud and potentially exhausting their mobile storage. Lastly, as the victim uses a detailed User-Agent header containing their device model, operating system version when accessing the HTTP content server, an attacker could hijack the connection to their server to steal sensitive information and initiate privacy-invasive attacks.

Impact. Our tests found that all three operators did not check the integrity of the downloading URL. In terms of terminals, we identified 4 affected vendors, including: 1) Xiaomi, Redmi and ZTE, who automatically download the entire file corresponding to the tampered URL without any validation; and 2) Samsung, who would check if the file matches the size declared in the XML (and send warnings to users of mismatch). While this validation mitigates DDoS and attacks intended to consume user data and storage space, it is still feasible for information leakage attack.

Root Cause. We consider this vulnerability to be a protocol design flaw. Upon scrutinizing the RCC.07 document [15], we found it only addresses confidentiality and integrity issues related to RCS services carrying file URIs, overlooking the potential scenario of a sender acting maliciously and sending XML with a harmful URL. In other words, the possibility of malicious XML modification by the sender has been ignored. Mitigating this security flaw is not straightforward. The operator must ensure the XML returned by the content server remains unaltered until it reaches the message server. Given the distinct functions of the content and message servers, this is a challenging task under the current protocol design. We'll provide several mitigation recommendations in Section 7.

5.3 V3. TLS Validation is Not Handled Correctly

Description. Our analysis also uncovered security issues when mobile devices use Transport Layer Security (TLS) for Wi-Fi connections. The TLS authentication process for certain devices is flawed, making them susceptible to man-in-the-middle attacks. Specifically, these devices only verify the certificate chain's issuance by a legitimate Certificate Authority (CA), neglecting hostname matching (i.e., if the certificate matches the domain name). Consequently, an attacker can use a TLS certificate from any domain they control to impersonate an RCS server and initiate a separate TLS connection to the actual RCS server, effectively establishing a man-in-the-middle attack.

Security Risks. In contrast to cellular access, while the deployment of TLS under Wi-Fi theoretically protects RCS traffic, the vulnerability of certificate validation makes the cryptographic protection broken again. By using a replaced certificate, a man-in-the-middle attacker can successfully establish an RCS connection to the flawed end device, enabling eavesdropping, interception, and even tampering with all RCS communications between the target device and the actual RCS server. This exposes victims to various potential risks, including identity theft, privacy invasion, and unauthorized access to personal information.

Impact. This is an implementation vulnerability on the client-side. We used `Let's Encrypt`[1], a public trusted certificate authority, to apply for a TLS certificate for a domain that we control, and tested the behaviors of the six mobile phones. The tests found that 3 of them were impacted, including Xiaomi, Redmi, and ZTE. They did not sufficiently verify the certificate, allowing the attacker to successfully establish a TLS connection and making them susceptible to an MITM attack. Besides, Huawei, Honor, and Samsung strictly verified the certificate and sent an RST packet to disconnect when the verification failed.

5.4 V4. Crash-inducing Security Issues

Description and Security Risks. We discovered a total of five security issues that could lead to crashes, four of which could be remotely triggered on the recipient's end without any interaction. The devices implicated include Samsung, Xiaomi, Redmi, and ZTE. We've categorized these issues as V4. It is noteworthy that the 5G messaging services tested on all devices were integrated within the native SMS application, implying that a crash in the 5G messaging service could likely affect the usability of the SMS application.

Impact. The impact of such crash vulnerabilities is closely tied to the end device's implementation. Therefore, we detail the vulnerabilities below separately.

1) Samsung restricts the size of an image thumbnail received via RCS to 1048576 bytes, i.e., 1MB. However, we discovered that the 5G message app on the phone crashes when the thumbnail size ranges between 504KB (approximately) and 1MB. This message triggering the crash will not be displayed on the phone, and messages being drafted at the crash time are not saved as drafts. Under normal circumstances, thumbnails are compressed by the operator's servers when images are sent via RCS, limiting an attacker's ability to exploit this issue as thumbnail size is not entirely controllable. However, when combined with the XML tampering issue (V2), attackers gain full control over the sent thumbnails, enabling them to easily remotely trigger a service crash on the target device.

2) When Xiaomi and Redmi phones receive a geolocation URI and the encoded string in `rcs-l` is malformed (e.g., *rcs-l=%E5%8C%9*), the SMS application will crash and exit. Similar to the behavior observed on Samsung phones, the message triggering the crash will not be displayed on the phone, and messages being drafted at the crash time would not be saved as drafts.

3) We identified three crash issues on ZTE mobile devices on the handling of malformed SIP messages, including: 1) When the device receives an audio file message and the `am:playing-length` value exceeds the range of `int` or is not a numeric string; 2) When the device receives a geolocation URI, and the `rcs-l` is malformed (similar to Xiaomi and Redmi); 3) When the `From` header in the received SIP message is malformed (for example, *⟨si:123⟩*), the messaging application would abruptly crash and exit.

[1]https://letsencrypt.org/

Table 4: MITM attacks validated

Network	Vulnerability	MITM Method			
		VPN	DNS Spoofing	ARP	Phishing Wi-Fi
Cellular	V1	✔	✔	-	-
Wi-Fi	V3	✔	✔	✔	✔

6 ATTACKS

To verify the feasibility of exploiting the identified vulnerabilities in real environments, we launched attack verification experiments and give the details in this section.

6.1 Man-In-The-Middle Attack

Both V1 and V3 lead to the lack of effective authentication of the identity of RCS servers, posing the risks of man-in-the-middle attacks. We conducted verification experiments of both these two scenarios.

Attack Procedure. The first step of the attack is to constitute a man-in-the-middle, i.e., make the victim's RCS traffic pass through the attacker. For the cellular scenario, although it may be challenging to perform a stable hijacking directly at the air interface [33], the attacker can (i) install malicious VPN software on the victim's mobile device or (ii) employ DNS spoofing to hijack the RCS traffic of the victim. The attack using method (i) does not necessarily require direct control of the VPN software. If the attacker controls the VPN server used by the victim or other nodes in the forwarding path, it is also possible to redirect traffic to the attacker and execute a man-in-the-middle attack. It is worth noting that even if a malicious node is unable to intercept traffic from a VPN server, it can still perform the attack of TCP injection. For the Wi-Fi scenario, (i) and (ii) are still valid. Besides, more attacking methods include (iii) deploying an ARP (Address Resolution Protocol) cache poisoning attack, (iv) using Dynamic Host Configuration Protocol (DHCP) spoofing, and (v) creating a phishing Wi-Fi are also effective. For example, when the attacker and the victim are located in the same Wi-Fi network, the man-in-the-middle can be achieved through ARP spoofing or DHCP spoofing, or by actually functioning as the Wi-Fi gateway (e.g., through the creation of a phishing Wi-Fi network). After successfully constructing an MITM, the attacker can perform a series of malicious operations, such as listening, intercepting, tampering or sending forged 5G messages.

Validation. The attack we have validated are shown in Table 4 with our developed tool, Sipano. Note that DNS spoofing needs to be executed based on actual situations, hence it is not included in Sipano's implementation. We validated DNS spoofing during our experiment by directly using a malicious (controlled by us) DNS server on the phone.

Attack Results. We successfully executed MITM attacks by V1 on Huawei, Honor, Xiaomi Redmi, and ZTE phones using cellular

access, and by V3 on Xiaomi, Redmi, and ZTE phones using Wi-Fi. All of these experiments showed the same attack result. We gained the ability to both impersonate the victim to send RCS messages to any recipient, and impersonate any sender to send RCS messages to the victim. We also gained the ability to intercept, eavesdrop on, and modify all RCS messages, leading to a significant breach of privacy and the exposure of sensitive communications. Additionally, we gathered specific information from the victim's devices, such as mobile model, phone number, International Mobile Subscriber Identity (IMSI) and system version, through these attacks. Note that the IMSI is carried in the SIP Authorization header during the registration process.

6.2 Zero-Click Remote Information Leakage

V2 leads a victim UE to access a malicious (attacker-controlled) server after receiving a tampered 5G message. One of the consequences is information leakage.

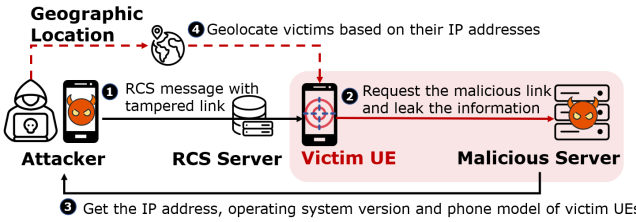

Figure 3: Process of a Zero-Click Remote Information Leakage Attack

Attack Procedure. Figure 3 illustrates the procedure of a zero-click remote information leakage attack. The attacker, equipped with only the target's phone number, sends an RCS message with a tampered thumbnail link. The victim's phone automatically accesses this link, resulting in an HTTP GET request to the server controlled by the attacker. The server can then log the originating IP address and User-Agent, revealing information including the victim's phone model and operating system version. The strong correlation between an individual and their phone number can also be exploited by an attacker to potentially track the geographic location of the target. This can be achieved by linking the recorded IP address with its corresponding geographic location[2] [3].

Moreover, when sending an RCS message containing XML with a tampered link to a phone that does not have the RCS service enabled, the phone will not receive any message, and the attacker's server will not receive an HTTP GET request. This characteristic can be used as a side channel to detect whether any arbitrary phone number has the RCS service enabled. This feature carries a certain degree of stealth — if the RCS service is not enabled, the target will not receive any prompt information.

Validation. For traffic control convenience, we first initiated an MITM attack on a mobile phone, thus controlling the RCS traffic from the phone. In addition, we set up `False Content Server` component to observe the specific content of the HTTP GET initiated by the victim. Next, we sent an RCS message to another phone (only its number was needed), the thumbnail URL in the XML of which was changed to *https://[hostname]:[port]*, where *hostname* and *port* are the domain name and listening port of `False Content Server`. Finally, we observed the content received on the `False Content Server`.

Attack Results. The Xiaomi and Redmi phones leaked the specific phone model, operating system version, and IP address. The ZTE phone revealed the specific phone model, Android version, and IP address. The Samsung phone divulged the specific phone model and IP address.

6.3 Phone Storage Occupancy and Mobile Data Consumption Attack

V2 could also be abused to consume the victim user's storage or data traffic charges.

Figure 4: Storage Occupancy Attack Process

Attack Procedure. As depicted in Figure 4, an attacker starts by sending an RCS message containing a tampered thumbnail URL to the victim UE. This altered URL leads to a download link for a large file. Consequently, the victim UE may automatically download the entire thumbnail, resulting in excessive occupation of phone storage and considerable consumption of mobile data.

Validation. First, we set up `False Content Server` and placed files of 1MB, 10MB, 100MB, and 1000MB sizes on it. Next, we controlled an attacker UE to send RCS messages with tampered thumbnail links to the victim UEs. The thumbnail links were tampered with to match the file download URL of `False Content Server`. Lastly, we observed if the victim UEs would completely download the large thumbnail.

Attack Results. The Xiaomi, Redmi, and ZTE phones automatically downloaded the entire thumbnail without any user interaction, making them vulnerable to this attack. In contrast, the Samsung phone restricts thumbnail size to 1MB, rendering it unaffected. The Huawei and Honor phones were not susceptible to this attack since they did not access the server specified by the attacker.

6.4 Reflection Amplification DDoS Attack towards an Arbitrary Server

V2 can also be abused to implement a reflection amplification attack, where a group of victim UEs form intermediate devices, download large files on the target server in parallel, and consume the target server's network bandwidth.

Attack Procedure. Figure 5 illustrates the process of initiating a reflection amplification DDoS attack using vulnerability V2. This attack is similar to the attack described in section 6.3, except that this attack targets the HTTP server rather than UE. The attacker begins by sending RCS messages, with tampered thumbnail links in the XML, to multiple UEs. These UEs, in turn, send HTTP requests

[2] https://www.iplocation.net/ip-lookup
[3] https://www.geolocation.com/

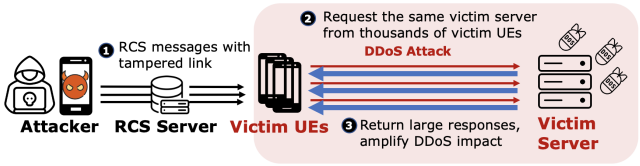

Figure 5: Process of a Reflection Amplification DDoS Attack

to the server corresponding to the link to download the file. It's important to note that the attacker only needs to send an RCS message with XML, which is typically small (about 3KB). In contrast, the UEs download the entire file, creating a reflection amplification attack. Although not all mobile devices can serve as reflectors to facilitate a DDoS attack, the attacker can initially enumerate phone numbers and collect information about devices that can be used for a DDoS attack through an information leakage attack.

Validation. Due to ethical considerations and the large number of mobile devices needed for such an attack, we could only use the mobile devices available in our lab for testing. We used Xiaomi, Redmi, and ZTE phones as reflectors to perform the attack and observe the amplification factor.

Attack Results. The total TCP traffic sent and received for sending the constructed 5G message is approximately 10 KB. By conservatively placing a 1,000 MB file on the HTTP server, we observed an amplification factor of approximately 100,000 times. It is important to note that this is not the maximum potential amplification factor. The factor could potentially be larger, depending on the file size hosted on the target server.

6.5 DoS Attack on the Messaging Application

Attackers can abuse V4 (crash issues) to attack Samsung, Xiaomi, Redmi, and ZTE cell phones, resulting in denial of service effects. The prerequisite for these attacks is only to obtain the cellphone number of the target victim, and all of them can be launched remotely. As the causes of crashes are related to specific phone models, we describe them separately below.

Case-I: Samsung, Xiaomi, and Redmi. Assume that the attacker controls a cellphone with 5G messaging enabled and has the ability to modify its traffic. Then the attacker sets up a malicious HTTP server and places a bmp image of size 523,966 bytes on it. For Samsung, the attacker sent an RCS message with a tampered thumbnail link to the target victim phone every 10 seconds. The thumbnail link was altered to the bmp image link on the malicious HTTP server. For Xiaomi and Redmi, the attacker sent a malformed geolocation URI text message (specifically, *geo:40,30;rcs-l=%E5%8C%9*) every 10 seconds. We then observed the attack results on the target phone: its messaging app crashed every 10 seconds. Along with the loss of the edited draft when the application crashes, it also disabled the target phone's ability to send text messages normally.

Case-II: ZTE. For ZTE, the attacker can send either malformed geolocation URI or malformed audio file XML with audio duration set to values exceeding the `int` range or non-numeric strings, similarly sent every 10 seconds. Following this, we observed the attack results on the target phone: its messaging app immediately crashed once, and upon receiving malformed messages subsequently, it wouldn't crash immediately but would do so intermittently. The target phone would not be able to receive any RCS messages, and the status would remain as "sending" when trying to send RCS messages (the recipient could receive the messages normally at this point).

7 COUNTERMEASURE AND DISCUSSIONS

7.1 Countermeasures

V1. Unprotected RCS Traffic Using Cellular Access. To address V1, operators should follow the recommendations and requirements by 3GPP and GSMA on protecting the interface between UE and P-CSCF over 3GPP radio access network using IPsec ESP. However, one reason that IPsec is not used may be due to the fact that not all end devices support IPsec for accessing IMS services. Therefore, a short-term solution is to use TLS, as GSMA responded that it is important to secure the Gm interface regardless of whether TLS or IPsec is used.

V2. URL Tampering When Sending Files. The current short-term mitigation strategy is strictly limiting the thumbnail size on the UE side. The proposed solution by OP-I is to rigorously verify on the RCS Server that the URL in the XML comes from the current operator. File interoperability between different operators can be achieved by pulling files through the operators' servers. We believe this approach, although more resource-intensive when operating across different operators, to be a feasible long-term solution. We have reached out to GSMA and they have sent a Liaison Statement (LS) to the GSMA member working group NG to discuss the solution.

V3. Incorrect Handling of TLS Verification and V4. Crash-inducing Security Issues. To mitigate V3, V4, and any future vulnerabilities, security specifications need to be clear on requirements and the quality of codes, particularly security-related codes, needs to be improved. For example, certificate validation needs to be clearly specified and implemented rigorously, including validating the hostname in a requeset URI against the identity in a server certificate (e.g., Subject Alternative Name field). Based on our findings and the recommendation from GSMA, 3GPP has updated TS 33.203 [3] by explicitly requiring UE to validate the P-CSCF server name against its TLS certificate [4]. More robust handling of exceptions is also important in avoiding potential security vulnerabilities.

7.2 Disclosure and Ethic Considerations

Vulnerability Disclosure. We reported the two security issues related to OP-I, OP-II and OP-III to the GSMA. GSMA recognized the vulnerability we found, assigned a CVD number (CVD 2023-0075), and is working on potential mitigation. GSMA also informed 3GPP of V3, and 3GPP updated TS 33.203 accordingly [4]. Both GSMA and 3GPP thanked our contribution on enhancing mobile network security. We also reported V1 and V2 to CNVD, and they accepted the vulnerabilities and assigned two CNVD numbers (CNVD-2023-95756 and CNVD-2023-95757). We also informed the smartphone manufacturers Xiaomi, Huawei, Honor, Samsung, and ZTE about the relevant security issues. Xiaomi recognized all the vulnerabilities we reported (V1, V2, V3 and V4) and gave us a bonus of 6,500 RMB. Huawei has recognized V1 and gave us a bonus of 5,000 RMB. Honor gave an initial response and is analyzing the vulnerabilities.

Samsung recognized V4 and gave us a bonus of 290 US dollars. ZTE confirmed all four vulnerabilities.

Ethical Considerations. The primary ethical concern of our work is transmitting altered SIP packets to actual 5G messaging services. We referenced existing papers [28, 46] on security testing of cellular networks, as well as authoritative guidance such as the Menlo Report [6], to design an ethically compliant experiment. First, we only tested our controlled smartphones to prevent potential disruption to other mobile users. Second, for messages destined for RCS servers or forwarded by them, we meticulously managed the message modification strategies. All altered fields are theoretically processed solely by the receiving UE, with the server merely forwarding them. We also ensured the absence of malformed fields that could lead to parsing errors in case the server attempted to parse them. Particularly during file download URL testing, we managed our content server to return files exclusively to the receiving UE's IP address, thereby avoiding security or performance impacts on the operators' servers due to incorrect downloads. Furthermore, we strictly regulated the message sending rate to a minimum of 10-second intervals, to avoid undue impact on servers. Another ethical concern involves the potential for Sipano to be maliciously used to launch actual attacks. Therefore, we provide Sipano only under the condition of verified identities and upon request, to mitigate the risk of its malicious use.

8 RELATED WORK

This paper focuses on the security of 5G messaging. In this section, we primarily discuss the known vulnerabilities of SMS and RCS, along with other 5G security issues reported in prior studies.

SMS Security Threats. As one of the most popular communication tools, SMS has garnered significant attention from both cyber attackers and the security community. Numerous studies have been published focusing on the detection of spam SMS [20, 21, 26, 31] and the behavior analysis of the criminal groups [25, 32, 45]. Besides spam, several studies also explored security threats stemming from vulnerabilities in SMS-related protocols and implementations. As early as 2011, Mulliner et al. [24] conducted fuzzing test on SMS applications of feature phones (non-smartphones) using controlled GSM base stations established on open-source software. They discovered that GSM phones have numerous implementation flaws, thus maliciously crafted messages can easily cause the phone to crash, restart, or even become unusable. Then with the emergence of smartphones, Schrittwieser et al. [35] studied the security issues in new SMS applications in 2012, primarily identifying errors at the business logic level (e.g., the authentication process) rather than the protocol layer. With the advent of 4G, SMS services have shifted from the traditional circuit-switched domain to the packet-switched domain, facilitated by IMS (IP Multimedia Subsystem). In 2016, Tu et al. [42] explored the security threats of IMS-based SMS, such as plaintext transmission and identity verification flaws.

In 5G, the text messaging service has undergone a new transition to RCS, which also implies the emergence of new potential threats. Zhao et al. [46] conducted the first systematical security analysis of the 5G messaging of 4 US operators, revealing OTP (One Time Password) authentication flaws that allow remote identity impersonation and RCS service hijacking. However, the three operators examined in this study utilize AKA (Authentication and Key Agreement) rather than OTP for authentication, which could potentially introduce new security issues that warrant further investigation.

5G Security Issues. While still in its infancy, 5G security research has gradually become an emerging focus in the academic community in recent years. Most existed works have delved into the formal analysis and verification of 5G protocols [7, 17, 23, 44]. Among them, the 5G-AKA authentication protocol is most analyzed and has been found to have several design flaws. For example, Borgaonkar et al. [8] found that the sequence number used in AKA was not sufficiently randomized, allowing attackers to continuously track users via fake base stations.

In addition to protocols related to security research, several works also explored security issues introduced by 5G features. To accommodate a variety of terminal devices, 4G and 5G allow terminals to declare their supported parameters through "device capabilities". However, Shaik et al. [36] confirmed that such features could be leveraged by attackers to identify active terminals within a network. Besides, Hussain et al. [16] found that the fixed paging occasions in 4G and 5G could be exploited by attackers to identify mobile users for privacy-related attackers such as location tracking. In response to these security flaws, some protective measures based on digital certificates and signature verification mechanisms have been proposed [9, 39], but they have not yet been implemented. In comparison, there are currently few papers studying the security issues of the real-world 5G network and application implementations. Most of them concentrated on 5G performance such as coverage and quality in commercial deployment. [27, 43]. Therefore, this paper could supplement the research gap in the field of 5G application security.

9 CONCLUSION

SMS has evolved significantly over three decades, notably with the advent of RCS. Along with the promising functionalities and global adoption of RCS, the evolution of such services also introduces new security risks. Our research provides a comprehensive security analysis of 5G messaging services, focusing on both carrier-side deployment and phone-side implementations. We developed a semi-automated tool, Sipano, which helps identify potential vulnerabilities in 5G messaging services. Our study, which tested three popular operators and six popular 5G messaging-enabled devices, unveiled numerous unreported security flaws from noncompliance with security specifications to software implementation vulnerabilities. We hope our work will raise awareness of 5G messaging security and spur research on mobile network safety.

ACKNOWLEDGMENTS

We extend our gratitude to the anonymous reviewers for their constructive suggestions for revision. Additionally, we thank GSMA, 3GPP and CNVD for acknowledging our work, as well as to all the involved vendors for their positive feedback and bounties. We also extend our appreciation to GeekPwn 2022 for their support and reward. This work is supported by the Taishan Scholars Program. Yiming Zhang is partially supported by the Shuimu Tsinghua Scholar Program. Jianjun Chen is in part supported by the NSFC #62272265.

REFERENCES

[1] 3GPP. 2021. *Network architecture.* Technical Standard (TS) 23.002. 3rd Generation Partnership Project (3GPP). Version 17.0.0.

[2] 3GPP. 2022. *IP Multimedia Subsystem (IMS).* Technical Standard (TS) 23.228. 3rd Generation Partnership Project (3GPP). Version 17.3.0.

[3] 3GPP. 2022. TS 33.203. 3G security; Access security for IP-based services. https://www.3gpp.org/ftp/Specs/archive/33_series/33.203/33203-h10.zip

[4] 3GPP. 2024. S3-240894. LS on GSMA CVD-2023-0075 – Certificate validation on IMS access interface. https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_115_Athens/docs/S3-240894.zip

[5] François Audet. 2009. The Use of the SIPS URI Scheme in the Session Initiation Protocol (SIP). *RFC* 5630 (2009), 1–56. https://doi.org/10.17487/RFC5630

[6] Michael Bailey, David Dittrich, Erin Kenneally, and Douglas Maughan. 2012. The Menlo Report. *IEEE Secur. Priv.* 10, 2 (2012), 71–75. https://doi.org/10.1109/MSP.2012.52

[7] David A. Basin, Jannik Dreier, Lucca Hirschi, Sasa Radomirovic, Ralf Sasse, and Vincent Stettler. 2018. A Formal Analysis of 5G Authentication. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018*, David Lie, Mohammad Mannan, Michael Backes, and XiaoFeng Wang (Eds.). ACM, 1383–1396. https://doi.org/10.1145/3243734.3243846

[8] Ravishankar Borgaonkar, Lucca Hirschi, Shinjo Park, and Altaf Shaik. 2019. New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols. *Proc. Priv. Enhancing Technol.* 2019, 3 (2019), 108–127. https://doi.org/10.2478/popets-2019-0039

[9] Hui Gao, Yiming Zhang, Tao Wan, Jia Zhang, and Haixin Duan. 2021. On Evaluating Delegated Digital Signing of Broadcasting Messages in 5G. In *IEEE Global Communications Conference, GLOBECOM 2021, Madrid, Spain, December 7-11, 2021*. IEEE, 1–7. https://doi.org/10.1109/GLOBECOM46510.2021.9685173

[10] GSMA. [n. d.]. Global forecast for RCS growth. https://www.gsma.com/futurenetworks/rcs/global-launches/. Accessed on May 7, 2023.

[11] GSMA. [n. d.]. GSMA RCS - Future Networks. https://www.gsma.com/futurenetworks/rcs/. Accessed on May 7, 2023.

[12] GSMA. 2019. RCS Universal Profile Service Definition Document Version 2.4. https://www.gsma.com/futurenetworks/wp-content/uploads/2019/10/RCC.71-v2.4.pdf

[13] GSMA. 2020. Chinese operators make major RCS commitment: Whitepaper. https://www.gsma.com/futurenetworks/latest-news/china-operators-make-major-rcs-commitment-whitepaper/. Accessed on May 7, 2023.

[14] GSMA. 2022. Rich Communication Suite Endorsement of OMA CPM 2.2 Conversation Functions. https://www.gsma.com/newsroom/wp-content/uploads//RCC.11-v11.0-2.pdf

[15] GSMA. 2022. Rich Communication Suite – Advanced Communications Services and Client Specification v13.0. https://www.gsma.com/newsroom/wp-content/uploads//RCC.07-v13.0-1.pdf

[16] Syed Rafiul Hussain, Mitziu Echeverria, Omar Chowdhury, Ninghui Li, and Elisa Bertino. 2019. 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. https://www.ndss-symposium.org/ndss-paper/privacy-attacks-to-the-4g-and-5g-cellular-paging-protocols-using-side-channel-information/

[17] Syed Rafiul Hussain, Mitziu Echeverria, Imtiaz Karim, Omar Chowdhury, and Elisa Bertino. 2019. 5GReasoner: A Property-Directed Security and Privacy Analysis Framework for 5G Cellular Network Protocol. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019*, Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz (Eds.). ACM, 669–684. https://doi.org/10.1145/3319535.3354263

[18] Pijus Jauniskis. [n. d.]. VPN statistics: Users, markets, and legality. https://surfshark.com/blog/vpn-users. Accessed on Mar 7, 2022.

[19] Cullen Fluffy Jennings, Ben Campbell, and Rohan Mahy. 2007. The Message Session Relay Protocol (MSRP). RFC 4975. https://doi.org/10.17487/RFC4975

[20] Nan Jiang, Yu Jin, Ann Skudlark, and Zhi-Li Zhang. 2013. Greystar: Fast and Accurate Detection of SMS Spam Numbers in Large Cellular Networks Using Gray Phone Space. In *Proceedings of the 22th USENIX Security Symposium, Washington, DC, USA, August 14-16, 2013*, Samuel T. King (Ed.). USENIX Association, 1–16. https://www.usenix.org/conference/usenixsecurity13/technical-sessions/presentation/jiang

[21] Zhenhua Li, Weiwei Wang, Christo Wilson, Jian Chen, Chen Qian, Taeho Jung, Lan Zhang, Kebin Liu, Xiangyang Li, and Yunhao Liu. 2017. 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. https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/fbs-radar-uncovering-fake-base-stations-scale-wild/

[22] Aniss Maghsoudlou, Lukas Vermeulen, Ingmar Poese, and Oliver Gasser. 2023. Characterizing the VPN Ecosystem in the Wild. In *Passive and Active Measurement - 24th International Conference, PAM 2023, Virtual Event, March 21-23, 2023, Proceedings (Lecture Notes in Computer Science, Vol. 13882)*, Anna Brunström, Marcel Flores, and Marco Fiore (Eds.). Springer, 18–45. https://doi.org/10.1007/978-3-031-28486-1_2

[23] Rhys Miller, Ioana Boureanu, Stephan Wesemeyer, and Christopher J. P. Newton. 2022. The 5G Key-Establishment Stack: In-Depth Formal Verification and Experimentation. In *ASIA CCS '22: ACM Asia Conference on Computer and Communications Security, Nagasaki, Japan, 30 May 2022 - 3 June 2022*, Yuji Suga, Kouichi Sakurai, Xuhua Ding, and Kazue Sako (Eds.). ACM, 237–251. https://doi.org/10.1145/3488932.3517421

[24] Collin Mulliner, Nico Golde, and Jean-Pierre Seifert. 2011. SMS of Death: From Analyzing to Attacking Mobile Phones on a Large Scale. In *20th USENIX Security Symposium, San Francisco, CA, USA, August 8-12, 2011, Proceedings*. USENIX Association. http://static.usenix.org/events/sec11/tech/full_papers/Mulliner.pdf

[25] Ilona Murynets and Roger Piqueras Jover. 2012. Crime scene investigation: SMS spam data analysis. In *Proceedings of the 12th ACM SIGCOMM Internet Measurement Conference, IMC '12, Boston, MA, USA, November 14-16, 2012*, John W. Byers, Jim Kurose, Ratul Mahajan, and Alex C. Snoeren (Eds.). ACM, 441–452. https://doi.org/10.1145/2398776.2398822

[26] Akshay Narayan and Prateek Saxena. 2013. The curse of 140 characters: evaluating the efficacy of SMS spam detection on android. In *SPSM'13, Proceedings of the 2013 ACM Workshop on Security and Privacy in Smartphones and Mobile Devices, Co-located with CCS 2013, November 8, 2013, Berlin, Germany*, William Enck, Adrienne Porter Felt, and N. Asokan (Eds.). ACM, 33–42. https://doi.org/10.1145/2516760.2516772

[27] Arvind Narayanan, Xumiao Zhang, Ruiyang Zhu, Ahmad Hassan, Shuowei Jin, Xiao Zhu, Xiaoxuan Zhang, Denis Rybkin, Zhengxuan Yang, Zhuoqing Morley Mao, Feng Qian, and Zhi-Li Zhang. 2021. A variegated look at 5G in the wild: performance, power, and QoE implications. In *ACM SIGCOMM 2021 Conference, Virtual Event, USA, August 23-27, 2021*, Fernando A. Kuipers and Matthew C. Caesar (Eds.). ACM, 610–625. https://doi.org/10.1145/3452296.3472923

[28] Shiyue Nie, Yiming Zhang, Tao Wan, Haixin Duan, and Song Li. 2022. Measuring the Deployment of 5G Security Enhancement. In *WiSec '22: 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks, San Antonio, TX, USA, May 16 - 19, 2022*, Murtuza Jadliwala, Yongdae Kim, and Alexandra Dmitrienko (Eds.). ACM, 169–174. https://doi.org/10.1145/3507657.3528559

[29] Jon Postel. 1980. User Datagram Protocol. *RFC* 768 (1980), 1–3. https://doi.org/10.17487/RFC0768

[30] Jon Postel. 1981. Transmission Control Protocol. *RFC* 793 (1981), 1–91. https://doi.org/10.17487/RFC0793

[31] Bradley Reaves, Logan Blue, Dave Tian, Patrick Traynor, and Kevin R. B. Butler. 2016. Detecting SMS Spam in the Age of Legitimate Bulk Messaging. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WISEC 2016, Darmstadt, Germany, July 18-22, 2016*, Matthias Hollick, Panos Papadimitratos, and William Enck (Eds.). ACM, 165–170. https://doi.org/10.1145/2939918.2939937

[32] Bradley Reaves, Nolen Scaife, Dave Tian, Logan Blue, Patrick Traynor, and Kevin R. B. Butler. 2016. Sending Out an SMS: Characterizing the Security of the SMS Ecosystem with Public Gateways. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*. IEEE Computer Society, 339–356. https://doi.org/10.1109/SP.2016.28

[33] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. 2019. Breaking LTE on Layer Two. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*. IEEE, 1121–1136. https://doi.org/10.1109/SP.2019.00006

[34] Eve Schooler, Jonathan Rosenberg, Henning Schulzrinne, Alan Johnston, Gonzalo Camarillo, Jon Peterson, Robert Sparks, and Mark J. Handley. 2002. SIP: Session Initiation Protocol. RFC 3261. https://doi.org/10.17487/RFC3261

[35] Sebastian Schrittwieser, Peter Frühwirt, Peter Kieseberg, Manuel Leithner, Martin Mulazzani, Markus Huber, and Edgar R. Weippl. 2012. Guess Who's Texting You? Evaluating the Security of Smartphone Messaging Applications. In *19th Annual Network and Distributed System Security Symposium, NDSS 2012, San Diego, California, USA, February 5-8, 2012*. The Internet Society. https://www.ndss-symposium.org/ndss2012/guess-whos-texting-you-evaluating-security-smartphone-messaging-applications

[36] Altaf Shaik, Ravishankar Borgaonkar, Shinjo Park, and Jean-Pierre Seifert. 2019. New vulnerabilities in 4G and 5G cellular access network protocols: exposing device capabilities. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, WiSec 2019, Miami, Florida, USA, May 15-17, 2019*. ACM, 221–231. https://doi.org/10.1145/3317549.3319728

[37] Rifaat Shekh-Yusef. 2020. The Session Initiation Protocol (SIP) Digest Access Authentication Scheme. *RFC* 8760 (2020), 1–9. https://doi.org/10.17487/RFC8760

[38] Rifaat Shekh-Yusef, David Ahrens, and Sophie Bremer. 2015. HTTP Digest Access Authentication. *RFC* 7616 (2015), 1–32. https://doi.org/10.17487/RFC7616

[39] Ankush Singla, Rouzbeh Behnia, Syed Rafiul Hussain, Attila A. Yavuz, and Elisa Bertino. 2021. Look Before You Leap: Secure Connection Bootstrapping for 5G Networks to Defend Against Fake Base-Stations. In *ASIA CCS '21: ACM Asia Conference on Computer and Communications Security, Virtual Event, Hong Kong, June 7-11, 2021*, Jiannong Cao, Man Ho Au, Zhiqiang Lin, and Moti Yung (Eds.). ACM, 501–515. https://doi.org/10.1145/3433210.3453082

[40] Christian Spanring and Alexander Mayrhofer. 2010. A Uniform Resource Identifier for Geographic Locations ('geo' URI). RFC 5870. https://doi.org/10.17487/RFC5870

[41] Randall R. Stewart, Qiaobing Xie, Ken Morneault, Chip Sharp, Hanns Juergen Schwarzbauer, Tom Taylor, Ian Rytina, Malleswar Kalla, Lixia Zhang, and Vern Paxson. 2000. Stream Control Transmission Protocol. *RFC* 2960 (2000), 1–134. https://doi.org/10.17487/RFC2960

[42] Guan-Hua Tu, Chi-Yu Li, Chunyi Peng, Yuanjie Li, and Songwu Lu. 2016. New Security Threats Caused by IMS-based SMS Service in 4G LTE Networks. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi (Eds.). ACM, 1118–1130. https://doi.org/10.1145/2976749.2978393

[43] Dongzhu Xu, Anfu Zhou, Xinyu Zhang, Guixian Wang, Xi Liu, Congkai An, Yiming Shi, Liang Liu, and Huadong Ma. 2020. Understanding Operational 5G: A First Measurement Study on Its Coverage, Performance and Energy Consumption. In *SIGCOMM '20: Proceedings of the 2020 Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication, Virtual Event, USA, August 10-14, 2020*, Henning Schulzrinne and Vishal Misra (Eds.). ACM, 479–494. https://doi.org/10.1145/3387514.3405882

[44] Jingjing Zhang, Lin Yang, Weipeng Cao, and Qiang Wang. 2020. Formal Analysis of 5G EAP-TLS Authentication Protocol Using Proverif. *IEEE Access* 8 (2020), 23674–23688. https://doi.org/10.1109/ACCESS.2020.2969474

[45] Yiming Zhang, Baojun Liu, Chaoyi Lu, Zhou Li, Haixin Duan, Shuang Hao, Mingxuan Liu, Ying Liu, Dong Wang, and Qiang Li. 2020. Lies in the Air: Characterizing Fake-base-station Spam Ecosystem in China. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna (Eds.). ACM, 521–534. https://doi.org/10.1145/3372297.3417257

[46] Jinghao Zhao, Qianru Li, Zengwen Yuan, Zhehui Zhang, and Songwu Lu. 2022. 5G Messaging: System Insecurity and Defenses. In *10th IEEE Conference on Communications and Network Security, CNS 2022, Austin, TX, USA, October 3-5, 2022*. IEEE, 37–45. https://doi.org/10.1109/CNS56114.2022.9947238