Insecurity of Operational IMS Call Systems: Vulnerabilities, Attacks, and Countermeasures

Yu-Han Lu, Sandy Hsin-Yu Hsiao, Chi-Yu Li, *Member, IEEE*, Yi-Chen Hsieh, Po-Yi Chou, Yao-Yu Li, Tian Xie, and Guan-Hua Tu, *Member, IEEE*

***Abstract*— IMS (IP Multimedia Subsystem) is an essential 4G/5G component to offer multimedia services. It is used worldwide to support two call services: VoLTE (Voice over LTE) and VoWiFi (Voice over WiFi). In this study, it is shown that the signaling and voice sessions of VoWiFi can both be hijacked by a malicious adversary. By hijacking the signaling session, s(he) gains the ability to make ghost calls to launch stealthy DoS (Denial of Service) or caller-ID spoofing attacks against specific cellular users. Such attacks can be carried out without any malware or network information, and require only the victim's phone number to be known. It is shown that phones vulnerable to the call DoS attacks can be detected at run time by exploiting a vulnerability of cellular network infrastructures referred to as call information leakage, which is exposed based on a machine learning method. Especially, the call DoS attacks can prevent victims from receiving incoming calls for up to 99.0% time without user awareness. Moreover, by hijacking the voice session, an adversary can launch stealthy free data transfer attacks based on phone numbers alone rather than IP addresses. The identified vulnerabilities/attacks are validated in the operational 4G networks of four top-tier carriers across Asia and North America with seven phone brands. The study concludes by presenting a suite of solutions to address them.**

***Index Terms*— IMS, VoWiFi, cellular security, 4G, 5G.**

I. Introduction

IMS (IP Multimedia Subsystem) is the core system for call services in the 4G/5G era and offers two basic services: VoLTE (Voice over LTE) and VoWiFi (Voice over WiFi). VoLTE is an essential voice solution for 4G LTE networks and supersedes the legacy 2G/3G call service. Meanwhile, VoWiFi complements VoLTE in areas with poor cellular signals by enabling telephony calls over WiFi networks. An Ericsson report [1] predicted that the number of their subscriptions will reach 6 billion in 2024, accounting for 90 percent of all 4G/5G subscriptions. Hence, it appears that IMS systems will inevitably play a decisive role for future call services.

VoWiFi extends the reach of the IMS call service, but, in doing so, enlarges the attack surface compared to conventional voice solutions. Its software-based framework is barely hardened by existing hardware-based security embedded in the telecom modem, and this has serious implications if an adversary succeeds in gaining full control over the phone OS (e.g., root access). In particular, there is a risk that vulnerabilities in VoWiFi may imperil the entire IMS ecosystem.

The software-based VoWiFi support motivates us to study potential vulnerabilities of the IMS call service; notably, the study is based on a responsible methodology that avoids harming any of IMS systems or users in operational cellular networks. Given the constraint of this methodology, we examine only the vulnerabilities that can be validated by our own phones serving as both the caller and the callee, and that do not impede normal operation of the IMS service for other cellular users. To explore the vulnerabilities with great impact, we focus on those which can cause the most dangerous security threats on the call service. They can be derived from the two major threat models which can be validated under the aforementioned constraint. First, the adversary as a cellular user attacks a specific cellular user on his call service; the most dangerous security threats are caller ID spoofing and call DoS (Denial of Service). Second, malicious caller and callee cooperate to attack their subscribed carrier with their call services; since the carrier's normal operation for other cellular users cannot be affected, the most dangerous security threat is to take advantage of the call services to get any unauthorized benefit (e.g., data service).

To launch the above attacks, the prerequisite is to allow the adversary to send fabricated messages to the IMS system; moreover, they shall be considered as valid messages so that the IMS system can react as normal operation. Since it can be extremely challenging to build security associations with the core network and the IMS system from scratch, the most viable way is to hijack the signaling/voice sessions which have been built by the VoWiFi service. It leads us to identify the first vulnerability, *no app-level data-origin authentication*, which can be exploited to enable the session hijacking. Alarmingly, this vulnerability allows the adversary to arbitrarily manipulate

Manuscript received 12 December 2021; revised 2 July 2022; accepted 16 August 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor Y. Zhang. This work was supported in part by the National Science and Technology Council (NSTC) in Taiwan under Grant 109-2628-E-009-001-MY3, Grant 110-2221-E-A49-031-MY3, Grant 111-2218-E-A49-023, Grant 111-2218-E-A49-013-MBK, and Grant 111-3114-E-A49-001; in part by the National Science Foundation (NSF) under Grant CNS-1815636 and Grant CNS-1814551; in part by the Center for Open Intelligent Connectivity from the Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan; and in part by National Center for High performance Computing (NCHC) for providing computational and storage resources. *(Corresponding author: Chi-Yu Li.)*

Yu-Han Lu, Sandy Hsin-Yu Hsiao, Chi-Yu Li, Yi-Chen Hsieh, Po-Yi Chou, and Yao-Yu Li are with the Department of Computer Science, National Yang Ming Chiao Tung University, Hsinchu 300, Taiwan (e-mail: chiyuli@cs.nctu.edu.tw).

Tian Xie and Guan-Hua Tu are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA.

Digital Object Identifier 10.1109/TNET.2022.3205183

TABLE I
IDENTIFIED SECURITY VULNERABILITIES AND ATTACKS IN OPERATIONAL IMS CALL SYSTEMS

Category	Vulnerability	Cause	Attack	Description	Can the vulnerability be applied? NA-I	NA-II	AS-I	AS-II
VoWiFi Signaling Session Hijacking	V1: No App-level Data-origin Authentication	Design Defect	Call DoS Attack	The access of the VoWiFi signaling session is not restricted to only the IMS app.	V	V	V	V
	V2: Caller ID Spoofing	Operational Flaw	Caller ID Spoofing Attack	The caller ID (i.e., phone number) of INVITE message can be arbitrarily specified, but is not verified by IMS.			V	
	V3: Abusing Reliability of Provisional Responses	Design Defect	Call DoS Attack	The acknowledgement of provisional responses can be abused to get the callee stuck in the proceeding state of a call session.	V	V	V	V
	V4: No Prohibition of Concurrent Call Attempts	Operational Flaw	Concurrent Attacks on Multiple Cellular Users	The caller is allowed to make concurrent call attempts.	V	V	V	
VoWiFi Voice Session Hijacking	V5: Data Smuggling over Voice Session	Design Defect	Stealthy Phone-number-based Free Data Transfer Attack	Non-voice data can be smuggled over voice session between two ends of a VoWiFi call through the IMS core.	V	V	V	V

the IMS call operation, and stems from the fact that the standard simply treats the device as one entity of the security associations in the Internet protocol security (IPsec) protection afforded to IMS services. The security parameters are stored within the phone itself, rather than in the IMS app running the VoWiFi session. Hence, if the attack phone is compromised, the security parameters may be easily leaked, thereby enabling the adversary to hijack the VoWiFi signaling/voice sessions and interact with the IMS system on a per-message basis.

By exploiting the session hijacking, we further explore the possibility of the aforementioned security threats, namely, caller ID spoofing, call DoS, and unauthorized service access. These security threats are possible and the exploration leads us to identify other four IMS vulnerabilities: *caller ID spoofing*, *abusing reliability of provisional responses*, *no prohibition of concurrent call attempts*, and *data smuggling over voice session*. Specifically, the first two vulnerabilities can result in the caller ID spoofing and call DoS attacks, respectively. Crucially, these attacks require no malware or network information, and need only a knowledge of the victim's phone number.

Moreover, the damage of the attacks can be aggravated by the vulnerability, no prohibition of concurrent call attempts, which allows a single smartphone to attack multiple cellular users simultaneously. The last vulnerability, data smuggling over voice session, enables unauthorized data service over voice session with free of charge. These vulnerabilities are rooted in either operational flaws of the carrier or design defects of the standard. Table I summarizes all the five vulnerabilities, and the corresponding root causes and attacks.[1]

Operationally, the call DoS attack works only for VoLTE and VoWiFi users located in the same carrier network as the adversary. Furthermore, for any target phone number, the phone may have been temporarily handed over from 4G/5G to 3G. Under these conditions, the phone may play the ringtone when subjected to a call DoS attack, thereby thwarting the desired stealthy nature of the attack. However, it is shown that a determined adversary can circumvent this obstacle by using a stealthy detection method to remotely detect attackable phones at run time (i.e., before the ringtone plays). In particular, a machine learning (ML) approach can be leveraged to explore the signaling message features available for runtime detection and these features can then be incorporated into the attack. The evaluation results show that such an approach enables the attacker to conduct stealthy attacks on the victim with call DoS up to 99.0% of the time.

[1]Notably, the identified vulnerabilities and attacks have been reported to GSMA; they had not been able to confirm a solution in the short term, and hence, moved the discussion to one of their standing working groups called the Fraud and Security Architecture Group, which would take a longer term look at those security issues.

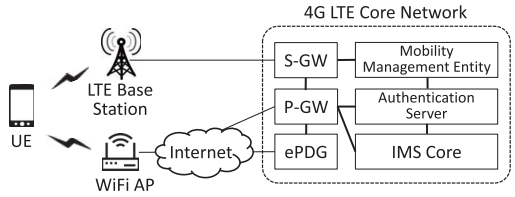


Fig. 1. 4G LTE network architecture with VoWiFi.

The identified vulnerabilities and attacks are validated by performing experimental trials in the operational 4G networks of four top-tier carriers across Asia and North America using seven phone brands. The experiments are conducted in a responsible manner such that no harm or disruption is caused to either carriers or cellular users. Specifically, no attempt is made to overwhelm the IMS system by flooding data traffic, or to crash it using malformed signaling messages. Furthermore, the researchers' phones are used as the victim devices in every case. Having validated the identified vulnerabilities and attacks, a suite of countermeasures is introduced.

The remainder of this paper is organized as follows. Section II presents the attack surface and model. Section III describes the details of the identified vulnerabilities. Section IV introduces the corresponding attacks. Sections V, VI, and VII present the proposed countermeasures, discussion, and related work, respectively. Finally, Section VIII concludes the paper.

II. VOWIFI ATTACK SURFACE

VoWiFi primer. VoWiFi is a cellular VoIP (Voice over IP) service [2] that enables cellular calls over WiFi networks. As shown in Figure 1, for a 4G LTE network architecture with VoWiFi support, the UE (User Equipment) consumes the VoWiFi service by connecting to the core network through the WiFi AP and Internet, while it consumes other services as normal through the LTE base station. The traffic flows of the two types of services reach the core network at the ePDG (evolved Packet Data Gateway) and S-GW (Serving Gateway), respectively. The ePDG enables untrusted non-3GPP access from the Internet and authenticates the UE through an authentication server before establishing an IPsec tunnel to the

UE [3], [4]. Within the core network, the P-GW (Public Data Network Gateway) then forwards the VoWiFi traffic between the ePDG and the IMS core. VoWiFi uses SIP (Session Initial Protocol) as its signaling protocol, but with some 3GPP-specific modifications [5], [6]. Specifically, it requires an IMS app installed at the UE to perform registration and mutual authentication prior to VoWiFi start-up based on the IMS-AKA (IMS Authentication and Key Agreement) protocol [7], [8]. The registration procedure derives IPsec ESP (Encapsulating Security Payload) [9] security associations between the IMS app and the core. While IPsec integrity protection over the SIP signaling is mandatory, the confidentiality is not [7].

Exposure of IMS potential vulnerabilities. VoWiFi has a larger attack surface than conventional cellular voice solutions since, whereas traditional IMS services hide all (e.g., CS-based) or part (e.g., VoLTE [10], [11]) of the operations and security functions within the hardware modem, VoWiFi keeps them in its software (including the IMS app and mobile OS). As a result, an adversary has the potential to learn the service operations from collected packet traces [12] and steal the security parameters (e.g., the security keys) from the software, or the delivery path from the SIM card to the IMS app using a sniffer such as SIMTrace [13]. These possibilities may allow the adversary to hijack the VoWiFi sessions. Having done so, s(he) can gain fine-grained interaction with the IMS core through the exchange of signaling messages. Any design defects of the call flow procedure or state machine can then be exploited to launch attacks on the IMS call service operations.

Attack model. In the experiments, the victims were mobile users with VoLTE or VoWiFi services. The attacks required only commodity smartphones without any remote access to the victim devices or malware installed on them. The attack phones carried SIM cards with VoWiFi services, and were rooted for full programmability and system data access. To maximize the attack impact, the WiFi environment was controlled to provide the attack phones with a strong WiFi signal with no interference. Moreover, the carrier networks were not controlled by the attacker and had no compromised facilities.

Experimental methodology. The experiments were conducted in the networks of four carriers: two from one country in North America and two from one country in Asia. The former two carriers, denoted as NA-I and NA-II, collectively account for more than 52.4% of the total market share in their country, while the latter two carriers, denoted as AS-I and AS-II, account for around 42.9% of the market share in their country. The full series of experiments was carried out in Carriers NA-I and AS-I. However, only the vulnerabilities were validated in Carriers NA-II and AS-II. In this study, 9 different phone models with Android versions 5.1.1 to 10.0.0 were used as attack phones; namely Samsung S5/S6/S8, Google Pixel XL/3a, hTC U11, Sony Xperia XA2, Essential PH-1, and Asus Zenfone 4. Meanwhile, the victim phones included 15 different models running on Android/iOS and were selected from 7 different brands, namely Samsung, Essential, Google Pixel, Asus, Apple, hTC, and Sony.

Responsible methodology. The experiments were conducted in a responsible fashion in order to avoid harming any of the carriers or cellular users. For the carriers, no attempt was made to overwhelm the cellular infrastructure or IMS core by flooding data traffic, or to crash the IMS using malformed SIP messages. The main focus was to validate the identified vulnerabilities, not to attack the carrier or cause any damage. Moreover, our own phones were used as the victim phones in order to avoid disrupting real-world cellular users. Notably, while the present tests focused only on attacks against phone devices, the exposed vulnerabilities may potentially open up even more powerful attacks against the IMS core itself.

III. Malicious Manipulation of IMS Call Service

In this section, we study the vulnerability of the software-based VoWiFi support and examine other IMS vulnerabilities that can be exposed by malicious manipulation of the IMS call service operation. By following the aforementioned responsible methodology, we explore the vulnerabilities that can be validated by our own phone devices and do not have impact on the IMS normal operation for other cellular users. To gain the ability for the malicious manipulation, we first examine whether the VoWiFi sessions can be hijacked. It leads us to identify the first vulnerability, no app-level data-origin authentication (V1), which allows the session hijacking. By hijacking the VoWiFi signaling and voice sessions, the adversary can obtain fine-grained control over the delivery of signaling and voice messages with the IMS system.

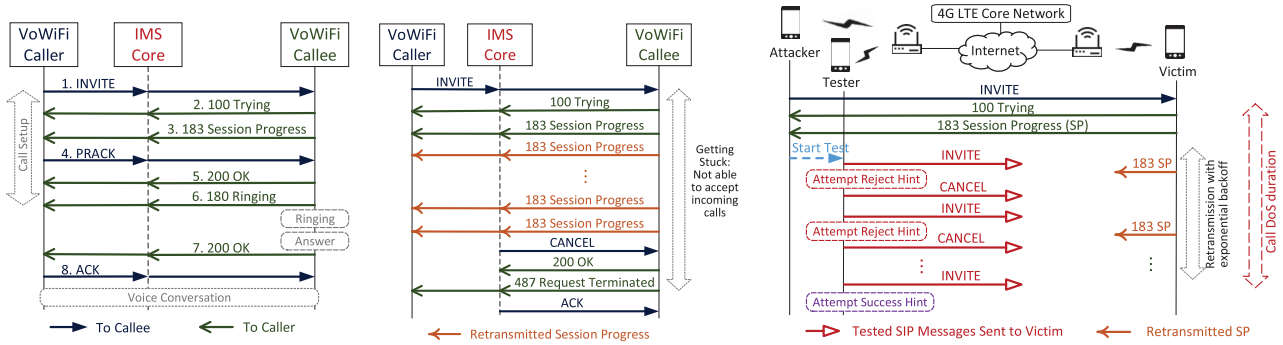
Furthermore, we focus on the vulnerabilities which can cause the most dangerous security threats on the call service. Given the threat model that the adversary, as a cellular user, attacks another cellular user on his call service, the major security threats are caller ID spoofing and call DoS; thus, two vulnerabilities are discovered to cause those two threats, respectively: caller ID spoofing (V2) and insecure reliability of provisional responses (V3). Since these two vulnerabilities are exploited based on the generation of call attempts, the feasibility of concurrent call attempts can aggravate attack damage by attacking multiple cellular users simultaneously from a single smartphone; it motivates us to discover another vulnerability, namely no prohibition of concurrent call attempts (V4).

Given the other threat model that adversaries serving as the caller and the callee in a call cooperate to attack their subscribed carrier; under the constraint of the responsible methodology, the security threat with great impact is to take advantage of the call service to get unauthorized services, where the data service can be the major concern. We then examine the existence of the security threat and identify the last vulnerability, data smuggling over voice session (V5).

In the following, we first present the VoWiFi session hijacking with the first vulnerability (V1), and then introduce the next three vulnerabilities (V2/V3/V4) and the last one (V5), which are explored from the hijacking of the VoWiFi signaling and voice sessions, respectively.

A. VoWiFi Session Hijacking

The IMS call service operates with two sessions: signaling and voice sessions. It relies on the signaling session to carry out the call control operation using SIP messages; during the



(a) VoWiFi call setup procedure. (b) VoWiFi callee gets stuck. (c) Gauging call DoS duration caused by an INVITE.

Fig. 2. Normal call setup procedure versus abuse scenario in which callee gets stuck without receiving PRACK.

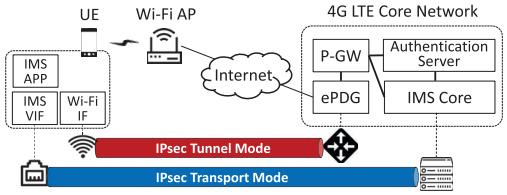


Fig. 3. Two-level IPsec protection over VoWiFi signaling session between IMS app and core.

call setup, the voice session is built for the voice delivery based on RTP packets. Figure 2(a) shows the VoWiFi call setup procedure, in which the IMS core mediates the delivery of the SIP messages between the caller and the callee. All of the messages except for the PRACK (Provisional Response ACKnowledgement) and 200 OK response messages are similar to those of conventional VoIP calls. The PRACK message is used to ensure the end-to-end reliability of the provisional responses employed to provide information on the progress of request processing (e.g., Session Progress) [14]. This reliability is essential for the IMS to provide carrier-grade voice services. After the call is accepted, the RTP packets are exchanged between two call ends for the voice conversation.

The VoWiFi signaling session between the IMS app and core is protected by two levels of security mechanisms in accordance with the standards [3], [4], [7], [15]. As shown in Figure 3, at the first level, an IPsec tunnel is built between the WiFi interface and the ePDG to facilitate untrusted access over non-3GPP networks [4]. When packets are sent via the IMS virtual interface (VIF), they are encapsulated into this tunnel and then delivered to the core network via the WiFi interface. At the second level, the integrity of the VoWiFi signaling session is protected by the IPsec transport mode, which is built between the IMS VIF and the core [15]. However, while such two-level security protection can defend against most outside attacks from non-3GPP networks, the signaling session is still vulnerable to threats originating from within the UE device itself, such as when the device is not trusted and a malicious app gains root access.

In contrast to the signaling session, the VoWiFi voice session is protected by only the IPsec tunnel at the first security level. When the voice packets belonging to a voice session are sent to the IMS core via the IMS VIF at one call end, they are delivered to the ePDG through the IPsec tunnel and then forwarded to the IMS core. The IMS associates the voice packets with the voice session based on the IP addresses and ports negotiated through the SDP in the preceding SIP messages, and finally forwards them to the other call end.

Note that since the security manner of the voice session is a subset of that of the signaling session, we focus on the hijacking of the signaling session and its success also indicates the feasibility of the voice session hijacking.

1) (V1) No App-Level Data-Origin Authentication: Since the VoWiFi signaling session has no app-level data-origin authentication, access is not restricted solely to the IMS app. Thus, when the IMS app relies on the mobile OS to carry out IPsec transport, it may be possible for an adversary to acquire the parameters of the IPsec security associations from the system, and then use these parameters to fabricate valid IPsec/SIP messages [12]. To hijack the session, two additional steps are required. First, the sequence numbers of the IPsec session should be tracked at run time, together with the corresponding TCP sequence numbers. Second, the default ESP padding algorithm [9] should be applied and the associated authentication data produced using the specified hash algorithm and keys. (Note that the HMAC-SHA-1-96 algorithm [16] is used by the carriers considered in this study.)

Experimental validation. Carriers NA-I and AS-I were indeed found to adopt the IPsec transport mode over VoWiFi signaling sessions. In addition, the initial REGISTER message sent by the IMS app included its capable security methods in the Security-Client field, such as the supported IPsec version IPsec-3gpp, the protocol esp, and the mode transport. However, the other two carriers did not enable this mandatory feature and were thus left unprotected.

The feasibility of VoWiFi session hijacking was examined by attempting to use fabricated SIP messages to make a VoWiFi call. Figure 4 shows a fabricated INVITE message (with the Session Name set to FORGED SIP), and the subsequent SIP messages. The responses of the Trying and Session Progress messages received from the IMS confirm that the forged INVITE message is considered to be valid. A forged PRACK message was then returned to the IMS. Thereafter, OK and Ringing messages were received and the callee phone started to ring. A similar outcome was obtained for Carrier NA-I. As for Carriers NA-II and AS-II, which do not have that

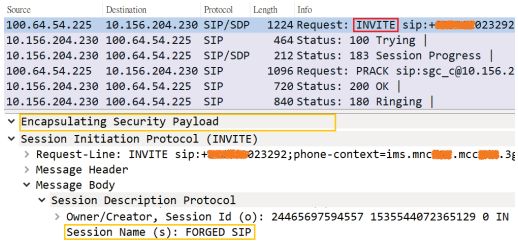
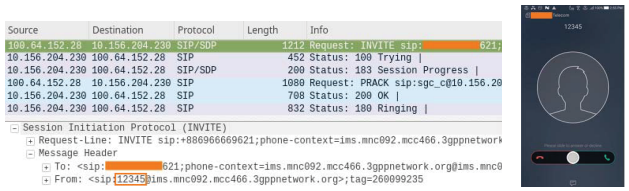


Fig. 4. VoWiFi call successfully made by forged `INVITE` and `PRACK` messages in AS-I network.



(a) Caller: fabricated SIP message with spoofed number. (b) Callee

Fig. 5. A VoWiFi user calls another user by spoofing his number as `12345`.

IPsec transport protection, the VoWiFi session can be easily hijacked.

Root causes and lessons. Vulnerability V1 can be attributed mainly to a *design defect* whereby the standard stipulates only device-level IPsec protection. This approach is reasonable for conventional voice solutions, such as VoLTE and 3G CS-based, since they hide (either fully or partially), the signaling operation in the device modem, which is protected by hardware-based security; notably, for the interaction with the mobile OS, the modem has proprietary interfaces, which can increase the difficulty of hacking modem. However, VoWiFi carries out its signaling operation using software. When the security parameters are passed to the mobile OS, they are at risk of being stolen. This inherent weakness of VoWiFi suggests that the standard should be updated to implement app-level data-origin authentication for the IMS system.

Moreover, V1 can be attributed in part to an *operational flaw* of IMS in the sense that Carriers NA-II and AS-II do not even enable mandatory IPsec protection. The absence of such protection may reflect an assumption on the part of the carrier that the signaling messages are already adequately protected by the first-level IPsec tunnel, and are thus robust toward outside network threats. However, such thinking ignores the very real risk of threats originating from inside the phone itself.

B. Signaling Session Hijacking: IMS Call Setup Manipulation

Once the SIP signaling session has been hijacked, the adversary can interact with the IMS core on a per-message basis. If the IMS core is not properly hardened against security threats, it may not only suffer service or system disruptions, but may also propagate threats to the cellular users. An examination of the practical IMS call service operation reveals three potential vulnerabilities below.

1) (V2) Caller ID Spoofing: Given vulnerability V1, the caller ID of a forged `INVITE` message can be arbitrarily specified. If the IMS system does not verify this ID, the `INVITE` with a spoofed caller ID can be forwarded to the callee and a spoofed call made accordingly.

Experimental validation. The experimental results showed that AS-I did not prohibit caller ID spoofing, but the other three carriers do. To validate this vulnerability, an `INVITE` message with a spoofed caller ID was fabricated and sent to the callee to make a call. Specifically, the caller ID in the front portion of the SIP ID in the From field was modified to `12345`, as shown in the red rectangle in Figure 5(a). The spoofed call attempt was successful, with proper SIP message exchange for call establishment from `INVITE` to `Ringing` (see Figure 5(a)) and the spoofed caller ID presented on the phone call GUI at the callee (see Figure 5(b)).

Root causes and lessons. The caller ID spoofing vulnerability is rooted in an *operational flaw* of the carriers since it can be prevented based on the existing information at the IMS. In particular, the IMS core maintains an established IPsec session in the transport mode with each VoWiFi user, and hence it knows the user identity and call ID. Consequently, it can, in theory, check whether the actual call ID is consistent with the caller ID claimed in the `INVITE` message. However, AS-I does not have such a function and simply forwards the `INVITE` messages without first checking the caller ID.

Importantly, an adversary can exploit this vulnerability to launch caller ID spoofing attacks against any cellular user, irrespective of the carrier to which they belong. For example, when a carrier with such a vulnerability is trusted by other carriers on account of its general reputation and size, they will most likely accept any call attempt originating from it on the assumption that it must be genuine. Notably, AS-I is the largest carrier in the Asia country, and hence if it is vulnerable to V2, it seems probable that other carriers may be vulnerable too.

2) (V3): Abusing Reliability of Provisional Responses: The establishment of an IMS call may fail in the absence of sufficient resources. However, the callee may have been alerted to the call in the meantime. To eliminate this annoying case, a mechanism known as precondition [17] has been introduced to enable resource reservation during the call setup process [15]. This mechanism relies on SIP provisional responses (e.g., `Session Progress`) and requires the support of a reliability mechanism that acknowledges the responses in order to confirm the reservation. The precondition mechanism is not widely used in Internet VoIP applications, but the 3GPP standard suggests its support for IMS call services [15] in order to maintain a carrier-grade call quality.

To enable the precondition mechanism, the caller sets an option-tag `precondition` in the `Supported` header field of the `INVITE` message, together with another option-tag, `100rel`, which indicates the reliability. As shown in Figure 2(a), the callee replies to the `INVITE` with a provisional response, `Session Progress`. In this response, the callee confirms a set of service requirements (e.g., the port and session parameters) that are specified in the `INVITE` SDP (Session Description Protocol), and sets the `precondition` option-tag. In addition, it commences resource reservation

based on the requirements and waits for a reliable alerting indication (i.e., the PRACK message) to alert the user. On receiving the Session Progress response from the callee, the caller also reserves resource at its side and acknowledges it with a PRACK message. After receiving this message, the callee device starts to ring.

However, the reliability mechanism of provisional responses may be abused in order to cause the callee to become stuck in the *proceeding* state of a call session [18]. In this state, the callee can neither accept other incoming calls, nor leave the session, until the PRACK message, which acknowledges the Session Progress, is received, or the session is canceled from the caller end. Thus, for reliability purpose, the callee retransmits the Session Progress message with an exponential backoff timer. When the number of retransmission attempts reaches a certain maximum number, the IMS cancels the session by sending a CANCEL message to the callee. Both the maximum number of retransmission attempts and the initial retransmission timeout are carrier-specific.

A caller can abuse this vulnerability to prevent the callee from receiving incoming calls without any awareness. As shown in Figure 2(b), the caller can send the INVITE to the callee without a PRACK, thereby keeping the callee in the proceeding state and preventing the callee device from ringing. Although the stuck state is maintained for only a short period of time, it can be exploited by an adversary as a building block to launch a long-time call DoS attack on the callee.

Experimental validation. The vulnerability was tested using three phones, namely an attacker, a tester, and a victim, where both the attacker and the tester were controlled to send SIP messages. As shown in Figure 2(c), the attacker sent the victim a single INVITE message without an answering PRACK, thereby causing the victim to repeatedly retransmit Session Progress messages. The DoS duration caused by the single INVITE was then gauged. Meanwhile, the tester continuously sent INVITE messages to the victim. Based on the last failed INVITE, the DoS durations were determined to be at least 14.5 s and 32.4 s for Carriers NA-I and AS-I, respectively. The callees in the two carriers sent 4 and 5 Session Progress messages to the attacker under the exponential backoff mechanism, respectively, before finally receiving a CANCEL message from the IMS core. Similar trends were observed for the other two carriers.

The experimental results reveal two important findings. First, vulnerability V3 also exists at the VoLTE callee for all of the considered carriers and test phones since VoLTE is supported by the IMS core with a similar call operation. Second, the callee is prohibited from making any outgoing calls during the DoS duration. For example, when using the GUI to dial a call at the callee, the GUI becomes stuck at the dialing page until the DoS duration ends. This negative impact happens for most test phones and is vendor-specific.

Root causes and lessons. The root cause of this vulnerability is a *design defect* wherein the standard fails to account for the possible negative impacts of the reliability mechanism. Nevertheless, it is reasonable to enable such a mechanism for two reasons. First, cellular resource is costly compared with that of the Internet. Second, the essential call service must be carrier-grade for the cellular network, and hence it is unacceptable to allow an invalid call to make the phone ring. It appears that the 3GPP standard [15] does not carefully review it in terms of security, and this security vulnerability is also not disclosed in the IETF standard. [17].

TABLE II
MAXIMUM POSSIBLE NUMBER OF CONCURRENT CALL ATTEMPTS

Carrier	Max number of concurrent call attempts	Provisional response	Failure status
NA-I	3	Yes	603 Decline
NA-II	3	Yes	403 Forbidden
AS-I	5	Yes	606 Not Acceptable
AS-II	1	No	N/A

3) (V4) No Prohibition of Concurrent Call Attempts: A caller is allowed to make successive calls to speak over a call while holding the other(s), or to have a conference call [19]. However, concurrent call attempts are prohibited by the system's GUI or call API. In other words, a new call attempt can be issued only when the current one has been answered. Notably, the caller can have concurrent call sessions in the conference call service, but s(he) must make them one by one, and add each callee separately to the conference call.

Seemingly, only one call attempt can be made at a time. However, a closer inspection reveals that this may in fact not be the case. For example, if the prohibition is fulfilled only at the end device (i.e., not at the IMS), once the system fence has been bypassed (via V1, for example), it may be possible to generate concurrent call attempts successfully. That is, the adversary may send out multiple INVITE messages concurrently and maintain a session state for each one.

Experimental validation. An experiment was performed to confirm whether or not carriers do in fact prohibit concurrent call attempts. Two concurrent call attempts were initiated from a single caller towards two different callees. The results showed that the SIP messages were properly handled at the caller and resulted in a Ringing status at both callees. A further test was made to determine the maximum number of possible concurrent call attempts from a caller to different callees for each carrier network. Table II presents the corresponding results. It is seen that the carriers differ not only in terms of the number of INVITE sessions they can maintain, but also the response messages they provide in the case that an INVITE is not accepted. For example, Carriers NA-I, NA-II and AS-I reply to an unaccepted INVITE with a provisional response including a failure status, whereas Carrier AS-II simply does not respond.

Root causes and lessons. Vulnerability V4 stems from an *operational flaw* of the carriers, and suggests that they not only enable concurrent call sessions in order to support conference calls or other services, but may also set number limits. As a result, concurrent call attempts are permitted at the IMS since the acceptance of a valid call attempt (i.e., an INVITE) leads to the initialization of a call session. Even through such an operation is not actually used in practice, and is even prohibited in the call API of the device, it nonetheless represents a possible opportunity for an adversary to abuse

the IMS call service. Thus, to mitigate this vulnerability, the IMS needs to differentiate call attempts from established call sessions, and then set appropriate limits on them.

C. Voice Session Hijacking: Unauthorized Data Service

IMS voice data are carried in RTP packets [7] and encoded with either an AMR (Adaptive Multi-Rate) [20] or AMR-WB (AMR Wideband) speech codec [21]. However, if the IMS does not carefully validate the voice packets, the voice session may be hijacked by an adversary and used to transport non-voice data instead. Although the IMS can check the RTP format of the data, it can be challenging to validate whether the RTP payload indeed carries voice data.

1) (V5) Data Smuggling Over Voice Session: Preliminary experimental trials revealed that non-voice data can be smuggled over voice sessions between the two ends of a VoWiFi call through the IMS core. In particular, given the IP addresses and ports of the voice session, an adversary can fabricate RTP packets with a payload containing non-voice data and send them out via the IMS VIF at one call end. The RTP packets are then forwarded to the other call end by the IMS core in the usual manner. This vulnerability enables two cellular users to exchange data over a VoWiFi call. This should clearly be prohibited by the carrier since non-voice data are charged based on volume rather than time, as for cellular voice traffic. Given an unlimited service plan for cellular voice data, an adversary can potentially abuse this plan to carry out non-voice data transfer free of charge.

At first glance, there seems little reason for an attacker to exploit this vulnerability to carry out data transmission between two device ends since such a capability is already available within many other Internet applications anyway. Conventionally, there are two major methods for the data transmission. One is to have an Internet server as an intermediary and require both transmission ends to log on to it, whereas the other is to set up a connection between two device ends based on their IP addresses; the latter method requires public IP addresses for the devices to reach each other, but the devices in most WiFi and cellular networks are assigned private IP addresses, and even for those with public IP addresses in some cellular networks, they cannot be reached due to the firewall deployed at the cellular network gateways.

However, given the above vulnerability, the adversary requires only phone numbers for the data transmission; neither the login of an Internet server nor obtaining public IP addresses is required. Moreover, the IP addresses used by mobile devices may change with locations, but the phone numbers do not change, even when using the VoWiFi service aboard. In addition, some carriers have service plans with free intra-network calls, so the data transmission can be free of charge. This vulnerability can thus offer a more convenient way for free data transmissions between mobile devices.

Experimental validation. The vulnerability was investigated by fabricating voice RTP packets and embedding marks in them. The packets were fabricated using the IP addresses and ports of the voice session and the RTP header information obtained from normal voice packets, as observed immediately after call establishment. The fabricated RTP packets were sent to the IMS VIF at one call end and a check was then made as to whether they were subsequently received at the other call end. The results showed that none of the four carriers prohibited data smuggling over VoWiFi voice sessions.

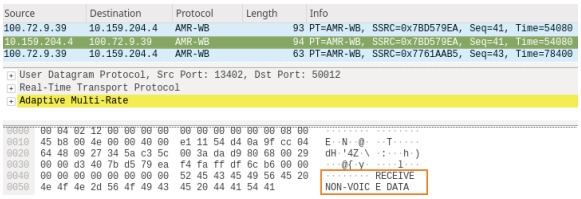


Fig. 6. Caller sends and receives non-voice data using RTP payload in Carrier AS-I.

We here take the result of Carrier AS-I as an example. Figure 6 shows successful data smuggling over a VoWiFi voice session from one call end as the sender to the other end as the receiver. Specifically, it is seen that the sender successfully sends an RTP/AMR-WB packet with `INJECT NON-VOICE DATA` out to the receiver and subsequently receives another RTP/AMR-WB packet with `RECEIVE NON-VOICE DATA` as an acknowledgement; only the details of the acknowledgement packet are shown due to space limit. Notably, the fabricated voice packets are successfully forwarded to the receiver only when the corresponding VoWiFi call is ongoing; moreover, those packets need to be formatted in the RTP format for the AMR speech codec [22].

It was observed that while RTP packets are fixed to several sizes, they vary with different carriers. For example, Carrier AS-I uses RTP packets with a size of 63 and 117 bytes, while Carrier NA-I uses packet sizes of 67 and 93 bytes. Thus, further experiments were performed using fabricated voice packets with sizes ranging from as small as 63 bytes to an MTU size of 1500 bytes. The packets of each size were sent out 5 times during an ongoing call. It was found that not all of the packets could pass through the IMS core. For example, the maximum permitted sizes in Carriers AS-I and NA-I were 1296 and 1336 bytes, respectively. It should be noted that these packet sizes (obtained from Wireshark) include the Linux Cooked Capture header with a size of 16 bytes, and hence the maximum permitted sizes of the voice IP packets for the two carriers are actually 1280 and 1320 bytes, respectively.

Root causes and lessons. As vulnerability V1, vulnerability V5 stems from the absence of app-level data-origin authentication from the IMS core, i.e., it arises from a fundamental *design flaw* of the standards. Notably, the voice session is not even protected by the IPsec transport mode at the second security level (see Figure 3) since this protection is not mandatory. Even though the voice session is protected against outside attacks by the IPsec tunnel at the first security level, the voice session can still be hijacked at the call ends to transport non-voice data. Since the content of voice traffic does not affect the call operation, and the IMS core simply forwards the packets between the two call ends, the potential damage of any adversarial attack on the voice session may be considered to be negligible, with the result that no additional security defense is deployed besides IPsec tunnel protection.

Validating voice traffic can be considered as a remedy for the weak access control of the voice session, but it can be challenging in differentiating between voice and non-voice traffic. The adversary can embed non-voice traffic in the RTP payload and fabricate RTP packets with the same size as those used normally. Differentiation can only be achieved by checking the decoded RTP payload at run time, which not only incurs a high overhead, but also presents a significant challenging in attempting to confirm whether the decoded audio signals are truly voice signals generated by the call ends.

IV. Attacks on IMS Call Service

By exploiting the above five vulnerabilities, we devise three major attacks, namely stealthy call DoS, caller ID spoofing, and stealthy phone-number-based data transfer attacks.

A. Ghost Calls: Stealthy Call DoS

A stealthy call DoS attack was devised against telephony users by generating ghost calls. Given only the victim's phone number, the attack prevented the victim's phone from both receiving incoming calls and making outgoing calls. Moreover, the attack was stealthy and did not cause the device to ring or attract the victim's attention in any other way. The details of the attack are described below.

1) Stealthy Call DoS Attack: The attack uses V3 as a building block and works only for callee phones using VoWiFi or VoLTE and subscribing to the same carrier as the attack phone. For simplicity, it is assumed here that the target phones are always attackable. It is noted that this assumption may not hold in real-world networks. Thus, in practical attack scenarios, the attacker must first detect whether or not the phone is actually attackable. An ML-assisted stealthy detection approach for achieving this is presented below in Section IV-B.

The reason why this attack works only when the two call ends belong to the same carrier is that current IMS systems from different carriers do not communicate with each other directly through the SIP protocol. Instead, they rely on the traditional PSTN (Public Switched Telephone Network) network. Thus, even when two call ends from different carriers both use VoWiFi/VoLTE, their call setup involves translations between the SIP and PSTN protocols, and hence the attacker has no means of manipulating the victim's call state machine.

Static DoS attack. An attack app was installed on the attack phone to initiate a call DoS duration on the victim phone by sending it an `INVITE` message without acknowledging any provisional responses. On receiving the `CANCEL` from the IMS, the attack app simply sent another `INVITE` message to the callee to initiate a follow-on DoS phase for a long-duration attack (see the upper panel in Figure 7).

There inevitably exists a non-DoS window period between the adjacent call DoS phases, during which the `INVITE` message from a normal call may arrive at the victim phone and cause the next DoS phase to fail. Thus, to shorten this non-DoS window, the attack phone was enabled to actively cancel the current DoS phase such that the `INVITE` for the following DoS phase arrived immediately after the `CANCEL` message (see middle panel in Figure 7). However, the experimental

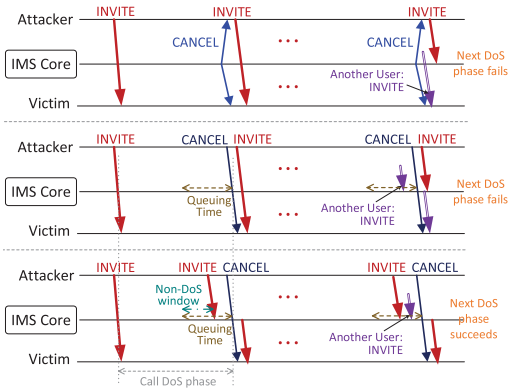


Fig. 7. Stealthy call DoS attack scenarios.

results revealed that the `INVITE` message sent by a non-victim before the attack phone sends out this `CANCEL` message may still successfully arrive at the victim and prevent the next attack `INVITE` from the attack phone being forwarded by the IMS core. This finding implies that the IMS core queues `INVITE` messages for a while before denying them.

Given the existence of `INVITE` queuing at the IMS core, only the first `INVITE` arriving within the queuing period prior to the `CANCEL` arrival is considered to be valid and accepted (as shown in the lower panel of Figure 7). Thus, to ensure the success of the follow-on DoS phase, the attack `INVITE` must be the first `INVITE` to arrive. In this case, the non-DoS window becomes the time interval between the start time of queuing and the arrival of the attack `INVITE`. To minimize the size of this non-DoS window, it is desirable to maximize the *attack interval*, i.e., the elapsed duration between the sending times of the `INVITE` and `CANCEL` messages at the attacker, respectively, given that the `INVITE` is accepted.

Static attack interval. In practice, the maximum valid attack interval that can reliably start a new DoS phase depends on the network conditions of the WiFi network, Internet, and cellular network, since varying wireless channel and network congestion affects the arrival times of the SIP messages. However, we discover that carriers generally prioritize VoWiFi traffic to ensure its low-latency delivery and service quality. They utilize differentiated services code point (DSCP) in IP networks and the 802.11e high-priority access category (AC) in WiFi networks. The resulting low-latency delivery not only minimizes the impact of network dynamics on the message arrival times, but also reduces the attack interval.

Experiments were thus performed to gauge the maximum valid attack intervals for Carriers AS-I and NA-I. For each carrier, the attack interval was varied from 0 ms to 600 ms in intervals of 10 ms. The success ratio of each interval was evaluated over 20 runs. The results showed that the maximum values of the attack intervals with a 100% success rate (i.e., valid attack intervals) were 100 ms in AS-I and 50 ms in NA-I, while the minimum ones of those with a 100% failure rate were 490 ms and 290 ms, respectively.

2) Adaptive Multilayer DoS Attack: Based on the experimental findings above, an adaptive multi-layer DoS attack was designed to dynamically approach the maximum valid

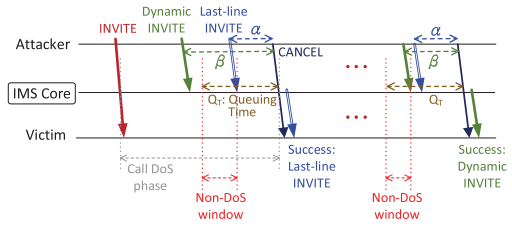


Fig. 8. The adaptive multi-layer DoS attack.

attack interval over time by exploiting two INVITE messages. The first INVITE was used to determine the maximum attack interval, and was sent out at varying times depending on the success of failure ratio of consecutive trials. In the event that the first INVITE message failed, the second INVITE was used as the last line of attack to ensure that the next DoS phase could be successfully launched. Note that the attack interval of the last-line INVITE was chosen as the interval which always succeeded for the particular carrier concerned.

Figure 8 illustrates the adaptive multi-layer DoS attack. The first INVITE initiates the first call DoS phase, and the attacker sends out a session CANCEL message after a specified DoS duration. However, before it sends out this CANCEL message, it sends out two INVITE messages for the next DoS phase. The first message is sent based on a dynamic attack interval β, while the last-line message is sent after a subsequent fixed interval α. (Note that β is adjusted dynamically with a granularity a ms based on b consecutive rounds of successes and failures.) For the former case shown in Figure 8, the dynamic INVITE message fails, while the last-line INVITE succeeds. Consequently, the non-DoS window at the IMS is the interval between the start time of queuing and the arrival of the last-line INVITE. In the latter case where the dynamic INVITE succeeds, and the last-line INVITE is invalid, the non-DoS window becomes shorter. Note that this adaptive attack requires three concurrent call attempts, i.e., three outgoing uncanceled INVITE messages: the INVITE of the current DoS phase, and the dynamic and last-line INVITE messages of the next phase.

3) Attack Prototype and Evaluation: The adaptive DoS attack was implemented on an attack phone and the DoS time was evaluated over a one-hour attack. Since it was hardly to know the exact DoS time at the IMS core, its lower bound was estimated as follows. Another test phone was used to send the victim an INVITE at a time when it would certainly fail. Accordingly, based on the experimental results presented in Section IV-A.1, the time was chosen as 490 ms and 290 ms before the attacker's CANCEL for Carriers AS-I and NA-I, respectively. The interval between the sending time of this invalid INVITE and that of the valid INVITE sent from the attack phone was taken as the upper bound of the non-DoS window, and hence the lower bound of the DoS time. The fixed and initial dynamic intervals (α, β) for the two carriers were set as (100 ms, 280 ms) and (50 ms, 200 ms), respectively. In addition, the call attack period, defined as the interval between two adjacent CANCEL messages, was set as 30 s and 12 s for the two carriers, respectively, based on the corresponding DoS durations caused by an INVITE. For the one-hour attack, the attack thus required 120 and 300 rounds of the attack period for Carriers AS-I and NA-I, respectively.

TABLE III

DoS Times in Percentage of One Hour for Various Attack Cases. Multi-Victim Attack Results Are in Average

Attack mode Number of victims	Adaptive 1	Static 1	Multi-victim 2	Multi-victim 4
AS-I (one attacker)	99.00%	98.70%	97.38%	96.80%
NA-I (one attacker)	98.41%	98.00%	93.60%	N/A

The results showed that, based on the always-failure case, the upper bounds of the aggregate non-DoS windows were 1.00% and 1.59% time, respectively. Moreover, comparing the adaptive attack with the static attack, in which only the last-line INVITE message was sent, and applying the upper bounds of the static attack (1.30% and 2.00% time, respectively), the adaptive attack was found to perform better with 23.08% and 20.50% gains on the lengths of the aggregate non-DoS durations, respectively. In other words, the adaptive attack caused the victim phone to suffer from call DoS for at least 99.00% (AS-I) and 98.41% (NA-I) of the time. Note that the victim phone did not ring during the experiment, thereby confirming the stealthy nature of the attack.

Multi-victim attack. A further attack was conducted based on the requirement of only one call attempt at a time. In this case, the attack phone sent out a new INVITE only after the existing call session was canceled. We used the phone to launch this simple attack against multiple victims concurrently, where the maximum number of victims depended on the maximum number of allowable concurrent call attempts in the particular carrier network (see Table II). Table III summarizes the DoS times for the various attack cases.

B. ML-Assisted Call DoS Attack

Before launching the call DoS attacks, the attacker needs to remotely detect attackable phones, i.e., phones that are using VoWiFi or VoLTE and are located in the same carrier network. Accordingly, an ML approach was developed to identify the SIP message features the attacker can use to carry out such a detection process. It was assumed that the attacker would perform ML-based identification for each interested carrier based on the call SIP traces collected before launching the attack, and would then perform detection based on the identified features through the course of the attack. The remote detection process should be stealthy (i.e., not cause the target phones to ring) and should also support real-time operation during attacks. It would allow the attack app to detect when the victim phone underwent handoff from VoWiFi/VoLTE to the 3G call service and, if so, to stop the attack immediately.

In general, the attack app needs to know the result of each attack INVITE such that it can take the appropriate action. The result depends on the call state of the target at the moment the INVITE arrives. Three call states are possible; idle, calling and talking, where these states indicate no proceeding of call setup or talking, proceeding with a call setup, and talking in a call, respectively. The attack INVITE succeeds (i.e., is accepted) in the idle and talking states, but

fails in the calling state. Thus, to ensure its success, the attack app should detect the call technology and state of the target phone at run time. To be stealthy, the attack app can only rely on the initial SIP messages that arrive at the attack phone before the PRACK delivery. Experimental trials showed that the content of the SIP messages varied with both the carriers and the phone models. Given a particular carrier, it is necessary to determine the specific set of features that can be used to classify the call technology and state at the callee. Moreover, the method used to do so must be independent of the phone model such that it works for all possible phones in the same carrier network.

In practice, it is very labor intensive to manually extract the classification features from the SIP traces of different phones for each carrier since the SIP messages contain a lot of information and their contents may vary with the phones. Specifically, the messages contain many fields, each of which has various values, and variances exist both in the message flow and the message interval. Thus, an ML-based classification method was developed to automatically identify the particular classification features for each carrier.

1) ML-Based Call Information Leakage: A preliminary experiment revealed the feasibility of using an attack app to cause a remote phone to leak its call technology and state from the SIP messages in response to silent calls. In this experiment, we collected the traces of the initial SIP messages from various cases with different combinations of call technology/state and carrier. Although many pieces of information from the SIP messages could be extracted as features for the classification of difference cases, not all of them were effective. We then sought to identify an effective feature set which can give the highest classification accuracy from potential features, which were determined empirically based on their attributes probably relating to the call technology/state; thus, the selected feature set can be used for the detection of the remote phone's call technology and state at run time. It thus called for an ML method to evaluate the classification accuracy for each potential feature set.

The support-vector machine (SVM) method [23], [24] was then chosen due to the following two reasons. First, the number of the potential features was as many as more than 10; such high-dimensional feature space can be supported by the SVM with non-linear classification. Second, we searched for the effective feature set by examining all the different combinations of potential features (here, there were 14 features and thus more than 16K combinations ($2^{14}-1$) were considered); for each combination, an ML model was trained and tested for the classification accuracy. Given such large number of required ML models with high-dimensional feature space in part, the SVM can be efficient.

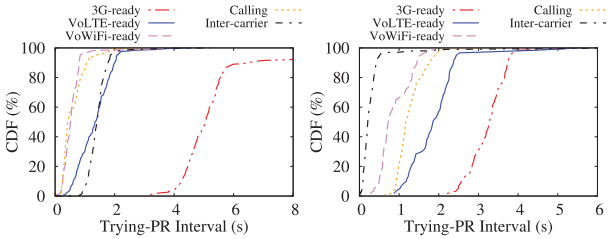
Trace collection. Three different call technologies were considered, namely 3G, VoLTE and VoWiFi, each with three possible call states: idle, calling and talking. By covering both intra-carrier and inter-carrier calls with different combinations of caller/callee phones and carriers, SIP message traces were obtained from more than 5,000 call attempts relating to 10 different phone models, 7 different brands, and 4 carriers. For each combination, the traces of 10 call attempts were collected. The collection process was performed using a semi-automatic tool, which for each callee setting (including the call technology/state and carrier) automatically went through the three states with 10 call attempts each time.

Categorization. The ultimate aim of the trace collection process was to detect attackable phones at run time and to obtain the results of each attack INVITE message at the attack app. It was deemed unnecessary to differentiate among all 18 possible combinations of call technologies (3G/VoWiFi/VoLTE), call states (idle/calling/talking), and carrier cases (intra-carrier/inter-carrier). Thus, we can group two sets of the combinations without affecting the need of our goal achievement. Specifically, all of the inter-carrier cases, for which the call DoS attack is not applicable, were grouped into one category designated as "inter-carrier", while the idle and talking states, both of which allow the INVITE to succeed, for each technology were grouped into the other category "ready". The callee in these two states treats new call attempts as incoming calls without difference. Thus, after the grouping process, only 7 categories remained, namely inter-carrier, 3G-ready, 3G-calling, VoWiFi-ready, VoWiFi-calling, VoLTE-ready, and VoLTE-calling.

Methodology. 14 features were empirically considered in the SVM feature space, consisting of 10 features extracted from the SIP message content and 4 features which were defined from the patterns of SIP messages. The former features included P-Early-Media, Allow, Session_Name, Bandwidth, etc., and were mainly carried by the non-100 SIP messages, e.g., Session Progress and Ringing. Meanwhile, the latter set of features comprised Trying-PR interval, Message_Flow, etc. The Trying-PR interval indicates the interval between the arrival time of the Trying message and that of its subsequent provisional response (Session Progress or Ringing) at the caller. The underlying rationale for this feature is that the Trying message is always returned immediately by the IMS, whereas the delivery of the provisional response can be triggered by different entities, e.g., the IMS, the SIP/PSTN translation gateway, and the inter-carrier gateway. It may thus result in different values for different call technologies.

To process the features, we converted string values into numerical values to form an input vector using the methods of one-hot encoding [25] and feature hashing [26], whereas the output was the index of those aforementioned 7 categories. We focused on the analysis of Carriers AS-I and NA-I, which have 2400 and 1600 collected traces, respectively; notably, similar findings can be also observed from the small set of traces from the other two carriers. The traces are split into 60% as the training dataset and 40% as the testing dataset.

Since not all the features were useful for the classification, we sought to find out the dominant ones which can give the highest classification accuracy. We did training and testing on a per-carrier basis, because there could be many carrier-specific parameters and operations. We tried all the different combinations of the possible features. For each combination, we trained an SVM model and tested its classification accuracy. Finally, the feature sets with the highest accuracy can be used for the detection.



(a) Carrier AS-I. (b) Carrier NA-I.

Fig. 9. The `Trying-PR` intervals vary with call technologies and states.

TABLE IV
CONFUSION MATRIX OF CALL CLASSIFICATION

Carrier AS-I				
Actual / Predicted	VoWiFi/VoLTE-ready	3G-ready	Calling	Inter-carrier
VoWiFi/VoLTE-ready	100%	0%	0%	0%
3G-ready	0%	100%	0%	0%
VoWiFi/VoLTE/3G-calling	0%	0%	100%	0%
Inter-carrier	0%	0%	0%	100%
Carrier NA-I				
Actual / Predicted	VoWiFi/VoLTE-ready	3G-ready	Calling	Inter-carrier
VoWiFi/VoLTE-ready	100%	0%	0%	0%
3G-ready	0%	95.8%	4.2%	0%
VoWiFi/VoLTE/3G-calling	0%	0%	98.6%	1.4%
Inter-carrier	0%	0%	1.4%	98.6%

Findings. For both carriers, the findings were as below:

- The VoWiFi-ready and VoLTE-ready cases cannot be clearly differentiated. However, since they both belong to attackable cases, they can be grouped together for detection purposes anyway.
- The three calling cases with different technologies cannot be separated. Hence, they can be also grouped into a single category. Note that the calling state is very short, and thus the call technology can be detected after it ends.
- The combined case of VoWiFi-ready and VoLTE-ready can be distinguished from that of the calling case.
- The 3G-ready case results in much larger `Trying-PR` values than the other cases (see Figure 9).

Table IV shows the SVM classification results for Carriers AS-I and NA-I; they are obtained from the feature sets which can result in the highest classification accuracy based on the testing dataset. Those chosen feature sets are different in the two carriers. For Carrier AS-I, all four categories can be clearly differentiated. Furthermore, there are eight 2-feature sets which achieve a 100% detection accuracy. One of these sets contains the `Session_Name` and `Message_Flow` features, for example, where the combination of these features yields different string values for each of the four categories. Notably, none of the feature sets contain the `Trying-PR` feature, which overlaps the different categories, as shown in Figure 9(a). However, the feature is still needed for stealthy detection, which requires the differentiation of the 3G-ready case from the other cases (see Section IV-B.2).

For Carrier NA-I, the 2-feature set consisting of `Allow` and `Trying-PR` yields the highest accuracy. Using this feature set, most of the data of the four categories can be separated; however, there are few exceptions. Specifically, 4.17% 3G-ready, 1.39% calling, and 1.39% inter-carrier data are mistakenly classified as calling, inter-carrier, and calling cases, respectively. This confusion can be attributed to the `Trying-PR` feature, which as shown in Figure 9(b), overlaps the different categories slightly. Notably, even though the overlap portion between the VoLTE-ready and 3G-ready cases is not small, the two cases can still be reliably differentiated based on the `Allow` feature.

TABLE V
TWO-PHASE STEALTHY DETECTION METHODS OF PHONE STATUS FOR CARRIERS AS-I AND NA-I

Phase	AS-I		NA-I	
	Action	Classification	Action	Classification
I	Single call: stop right after receiving `SP`	Check `SN` and `MF`	Multi-call: stop right after receiving `SP`	Check `Allow` and Interval (inter-carrier: $[0.01, 0.57]$)
II	Single call: stop at 3.0 s after receiving `Trying`	No non-100 provisional resp: 3G-ready; Otherwise, check `SN` and `MF`	Multi-call: stop at 2.2 s after receiving `Trying`	Check `Allow` in a non-100 provisional resp if any; otherwise, 3G-ready after n calls

The few exceptions in the detection reliability for Carrier NA-I can be avoided by applying a judgement based on multiple trials. For example, the inter-carrier, calling, and 3G-ready cases have 97% data in $[0.01, 0.57]$, 98% data in $[0.61, 2.06]$, and 100% data in $[2.21, 5.64]$, respectively, in terms of the `Trying-PR` feature. Thus, by assuming that each case has a probability ρ of falling within a given range, a threshold setting of θ can be used to exclude the possibility of one case. In particular, at the nth detection trial with m times not in the range, the case can be excluded when $(1-\rho)^m \rho^{n-m} < \theta$.

2) Stealthy Detection of Phone Status: Based on the findings above for the call information leakage, a stealthy attack was devised for detecting the status of the target phone by sending an `INVITE` to the target phone and then observing the response. To be stealthy, the attack must prevent the target phone from ringing during the detection process. The absence of a `PRACK` message in V3 does not suppress the ringtone in the inter-carrier and 3G-ready cases; however, the inter-carrier callee does not ring when the caller cancels its call attempt right after it receives the provisional response. Similarly, for the 3G-ready callee, the ringtone does not sound if the caller cancels its call attempt before receiving the provisional response since the long `Trying-PR` interval allows the caller to differentiate it from the other cases. Thus, the following two-phase stealthy detection method was devised: (1) inter-carrier determination; and (2) call status classification, which detects one of the other three intra-carrier cases. The first phase allows an attacker to exclude inter-carrier phones from the potential attack targets, while the second phase detects the status of the victims at run time during the attack. The two-phase stealthy detection methods for Carriers AS-I and NA-I are summarized in Table V.[2]

Evaluation. The stealthy detection performance of the developed app was evaluated for both carriers. In each run, the app sent an `INVITE` to the target phone and then detected the phone status at run time. Seven scenarios were considered: 3G-ready/calling, VoWiFi-ready/calling, VoLTE-ready/calling,

[2]In the action field, an `INVITE` is sent for each call, and the stop is done by sending `CANCEL`. Interval, `SP`, `SN`, and `MF` stand for `Trying-PR` interval, `Session Progress`, `Session_Name`, and `Message_Flow`, respectively.

and inter-carrier. For each carrier, 25 runs were conducted for each of the first 6 scenarios. In addition, 25 runs were conducted for each carrier in the inter-carrier case. For each run, both the detection output of the app and the given scenario were collected. The results showed that, for both carriers, the app accurately classified the cases into four categories (VoWiFi/VoLTE-ready, 3G-ready, calling, and inter-carrier) with 50, 25, 75, 100 runs, respectively.

3) Application of Stealthy Detection Into Call DoS: An adversary can apply the two-phase detection method to launch stealthy call DoS attacks against a set of valid phone numbers. For example, given several cellular accounts with Carriers AS-I and NA-I, (s)he can use the first phase of the detection process to identify which phone numbers belong to which carrier. For each phone number belonging to one of the carriers, (s)he can then launch a detection-enabled call DoS attack by applying the second phase of the detection method. The attack operates in two modes, attack and probing, for each potential victim. In the attack mode, the attack app launches the call DoS attack against the victim while continuing to detect its status. It persists with the attack until the victim status becomes 3G-ready, at which point, it switches to the probing mode and periodically probes the victim's status. If the victim switches back to VoLTE or VoWiFi, the attack returns to the attack mode. Note that the calling state does not trigger the mode switch since the call technology cannot be determined.

For evaluation purposes, the second phase of the detection method was integrated into the call DoS attack. For Carrier AS-I, the detection process can be accomplished by a single call attempt, and hence it was enabled for each attack `INVITE`. Specifically, any attack `INVITE` which did not produce a non-100 provisional response within 3.0 s after `Trying` was canceled, indicating that the victim phone was detected to be in a 3G-ready status. For Carrier NA-I, the detection process relies on multiple call attempts. In particular, for each call DoS phase, the attack app used three `INVITE` messages to perform detection. As discussed earlier, the adaptive attack involves two different types of attack `INVITE` messages: dynamic and last-line (see Figure 8). To avoid impeding the attack operation, the attack app sent an additional `INVITE` specific for detection purposes before the dynamic one. The `INVITE` was sent so early that it was sure to be canceled successfully before the delivery of the last-line `INVITE` (here, 3 s earlier than the last-line `INVITE`) since the maximum number of concurrent `INVITE` messages is 3. In performing the detection-enabled attack, each `INVITE` which did not generate a non-100 provisional response within 2.2 s after `Trying` was canceled. In the event that none of the three `INVITE` messages generated a provisional response, the victim phone was considered to be in a 3G-ready status.

Evaluation. Table VI shows the DoS times of the various detection-enabled attack methods. It is observed that enabling detection in the attacks incurs a very small overhead, with only up to 1.60% reduction in the DoS time.

TABLE VI

DoS Times in Percentage of One Hour for Various Detection-Enabled Attacks

Attack mode (detection-enabled) Number of victims	Adaptive 1	Multi-victim 2	Multi-victim 4
AS-I (one attacker)	98.86%	97.20%	95.80%
NA-I (one attacker)	98.20%	92.00%	N/A

C. Caller ID Spoofing Attacks

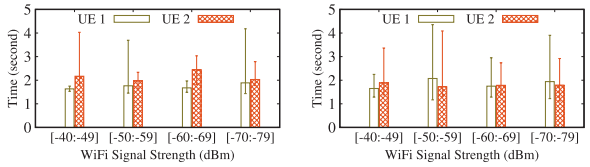
Adversaries can exploit vulnerability V2 to launch social engineering attacks through caller ID spoofing, in which they pretend to be officers from government agencies, or employees from financial institutions, for example, in order to lure victims to transfer money to them, or hand over credential information, such as account passwords. A validation experiment was thus performed in which call ID spoofing on the phone numbers of 10 government agencies and 10 financial institutions was conducted by making VoWiFi calls from a smartphone in the AS-I network. The results showed that all of the phone numbers could be successfully spoofed, even though a mobile account was used in the attack and the phone numbers of all the organizations were landlines. Notably, the adversaries are charged for the calls of the caller ID spoofing attacks.

D. Stealthy Phone-Number-Based Data Transfer Attack

An adversary can exploit vulnerability V5 to carry out a persistent stealthy phone-number-based data transfer attack on a VoWiFi voice session. To carry out such an attack, the initiator of the data transfer needs only the responder's phone number. The bidirectional data transfer is stealthy without carrier awareness. Thus, if attackers can have a cellular plan with unlimited voice service, they can perform stealthy data transfer free of charge. Even if the attackers are traveling outside of their countries, they can still use phone numbers to perform data transfer if they can have WiFi access and enable VoWiFi. Such stealthy data transfer is not only convenient for the attackers, but also highly secure since the data transfer is protected by the IPsec tunnel built for the IMS services. Even through the available throughput of the stealthy data transfer is not large (i.e., several tens of Kbps), it is still sufficient for the delivery of important text documents.

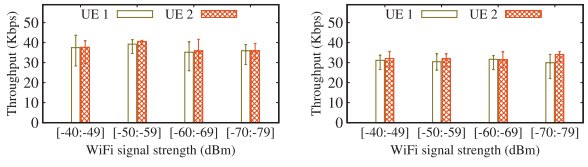
The stealthy data transfer process described above differs from the conventional SMS (Short Messaging Service) and Internet data/messaging transfer services in two key regards. First, while SMS also requires only the phone number to effect data transfer, it allows only one-time unidirectional delivery with a small amount of text per request and may not be free of charge. Second, Internet data/messaging transfer requires an Internet server to make a rendezvous between the two communication ends and requests them to login with their user credentials. Moreover, data/messaging transfer proceeds based on the IP address of the two ends, which may change over the course of the communication between them, whereas the call end phone numbers remain always the same.

To explore the stealthy phone-number-based data transfer attack further, an app was developed which allowed for the input of a responder's phone number and then selected a file to be delivered to the responder at that number. Since the attack has to be launched during an ongoing call, the app also initiated a VoWiFi call attempt with the responder using fabricated SIP messages based on vulnerability V1. The app at the responder end replied to the call attempt by forging SIP messages and then accepted the call based on a



(a) Carrier AS-I (b) Carrier NA-I

Fig. 10. Maximum/average/minimum initialization delays for stealthy data channel to be ready for transmission for various WiFi signal strengths.



(a) Carrier AS-I (b) Carrier NA-I

Fig. 11. Maximum/average/minimum attack throughput for stealthy data transfer service for various WiFi signal strengths.

list of recognized phone numbers specifically configured for the attack. Once the call was accepted, the initiator delivered the selected file to the responder by fragmenting the file and embedding it within the payloads of multiple voice packets.

The performance of the stealthy data transfer attack was evaluated from three perspectives: the attack initialization delay, the throughput, and the duration. Experiments were conducted for Carriers AS-I and NA-I using two different devices each, namely Samsung S6/S8, Google Pixel 3a, and hTC U11. The impact of the WiFi signal strength was also taken into account by classifying it into four cases: [-40∼-49], [-50∼-59], [-60∼-69], and [-70∼-79] dBm. (Note that if the WiFi signal strength is not larger than -80 dBm, a call handover from VoWiFi to VoLTE may be triggered.) Each device was tested with 15 runs in every case.

Initialization Delay. The initialization delay indicates how fast the stealthy data channel becomes available for transmission upon making a request. In the experiments, the initialization delay was measured as the duration between the time at which the `INVITE` message was sent out and that at which an `OK` message was received from the responder indicating its willingness to accept the call. Once the `OK` message was received, stealthy data transmission commenced. Figures 10(a) and 10(b) show the max/avg/min initialization delays for Carriers AS-I and NA-I, respectively. For AS-I, the average delays of the two tested devices range from 1.63 s to 2.44 s, while the maximum delay is 4.18 s. In NA-I, the average delays range from 1.65 s to 2.07 s, and the maximum delay is 4.36 s. Notably, the delays do not decrease with an increasing WiFi signal strength since the VoWiFi voice packets are sent by default with the 802.11e high-priority AC protocol. Together with the DSCP in IP networks, the impact of network dynamics can thus be minimized. Consequently, the variation in the initialization delays shown in Figure 10 can be attributed mainly to the processing times of the IMS system and UEs.

Attack Throughput. In conducting the attack, the non-voice data were carried using the maximum RTP message size, namely 1280 and 1320 bytes for Carriers AS-I and NA-I, respectively. For each device, the attack comprised 15 runs of 1-minute duration each. The transmission rate of the non-voice data was set to be larger than 50 Kbps, and hence exceeded the capacity of the voice sessions. Figures 11(a) and 11(b) show the experimental results for the max/avg/min attack throughputs for Carriers AS-I and NA-I, respectively, given different ranges of the WiFi signal strength. The average throughputs for the two carriers over the considered WiFi signal range are 37.23 Kbps and 31.58 Kbps, respectively. Moreover, the maximum throughputs are 43.69 Kbps and 35.62 Kbps, respectively. Overall, the results suggest that the available bandwidth of the voice session for each call is constrained by the IMS core. In addition, variations in the WiFi signal strength have very little impact on the attack throughput, which suggests that the bottleneck between the two communication ends is located at the IMS core.

Attack Duration. A final experiment was performed to examine the duration for which the stealthy data transfer attack could be sustained without being interrupted as a result of an abnormally large volume of voice sessions for IMS calls. As in the previous experiment, the RTP message size was set to its maximum value for each carrier. For each test device, the attack was continued for one hour. For both carriers, the attack continued for the full one hour without being interrupted, and there was no sign of any time constraint or abnormal detection warning from either carrier. During the one-hour experiment, 14.38 MB and 15.05 MB of non-voice data were successfully delivered over Carriers AS-I and NA-I, respectively.

Note that it was observed that the data transfer could cause normal voice messages to be dropped so that the voice quality could be affected. However, it should not affect the call quality of other devices in the same carrier network, since the bandwidth of each voice session is limited by the IMS.

V. Solution

In this section, we proposed a suite of short-term remedies to address the vulnerabilities described above, and analyzed their overhead. The proposed remedies are standard compliant, so they allow carriers and vendors to deploy them in the current IMS systems. Note that the proposed solutions are not intended to be long-term fixes for the identified vulnerabilities; the development of such solutions requires the concerted effort of carriers, network/phone vendors, and the cellular standard community based on their practical concerns.

App-level data-origin authentication. Any entity which exchanges messages with the IMS shall be a legitimate IMS app so that vulnerability V1 can be addressed; it can also address V5 by preventing the voice session from being hijacked. Such data-origin authentication can be achieved using the current IPsec transport-mode security mechanism, which is mandatory for the signaling session and stipulated in the standard [7]; however, the entity of the IPsec security associations at the end device shall be the IMS app. Moreover, to prevent IMS session hijacking, the IMS keys used by the IPsec shall not be leaked outside the IMS app and SIM card.

This component requires two security measures. First, the IMS app shall embed the IPsec implementation without relying

on the mobile OS such that it can keep the IMS keys safe inside itself. Second, the IMS app shall be authenticated by the SIM card such that it can securely obtain the IMS keys, which are generated by the ISIM (IMS Subscriber Identity Module) module of the card. Having being authenticated, the app can build security associations with the SIM card to effect secure delivery of the IMS keys, thereby preventing the adversary from extracting them with an SIM card sniffer. Note that the SIM card is assumed to be trusted with hardware-based security, and hence the IMS app can still be authenticated even in the event of a compromised or rooted OS.

The major overhead is that the IMS app builds IPsec transport-mode security associations by itself instead of the mobile OS. Specifically, the proposed security measure deals with the IPsec operation in the user space, whereas the current implementation is in the kernel space; however, the IMS app, a system app deployed by the phone vendor, can be given high priority on the resource usage so that the performance of its IPsec operation cannot be sacrificed. Although new security associations between the IMS app and the SIM card are needed, they are built whenever the IMS app starts to run; thus, the call attempt or establishment would not be delayed. Notably, the embedded IPsec implementation can increase the size of the IMS app and its memory usage.

Caller ID verification. The IMS core shall be enabled to verify the caller ID in the SIP messages at run time and block spoofed IDs to address vulnerability V2. For each VoWiFi user, the IMS core has an established IPsec session based on the user's profile, and hence it knows the user identity, e.g., IP Multimedia Public Identity. It can then query the home subscriber server (HSS) using this identity to obtain the user's caller ID. By considering overhead, this component needs to query the caller ID and verify it for only the SIP `INVITE` message. It may add delay overhead to the delivery of each `INVITE` message, and the overhead depends on how fast the above two actions can be performed.

Delay call binding. A delay call binding mechanism is further proposed to address V3 by delaying call binding to the arrival of the `PRACK` rather than of the `INVITE`. Even though many attack `INVITE` messages may arrive at the callee, the mechanism can bind the call to the earliest `INVITE` which returns the `PRACK` and then start to play the ringtone. In this way, it can prevent the callee from becoming stuck with a specific `INVITE`. In general, both the callee and the IMS core consider sessions without a `PRACK` to be pending ones. Thus, when seeing an `INVITE` without any pending sessions, they reserve resource for a call, but do not bind it to the `INVITE` session. The callee follows the same call setup procedure to serve it. Thereafter, no new resources are allocated for further `INVITE` messages. When a `PRACK` message is subsequently returned, both the callee and the IMS core bind the call resource to the corresponding session and dismiss the other pending sessions. The call resource for the callee will be released once no pending sessions exist.

The major overhead of this component lies in the maintenance of multiple call attempt sessions. For the call setup resource, it is similar to the current call service operation, where only the resource needed for a single call is required at any time. Since the callee reacts to each call attempt session individually, no obvious delay can be observed if the callee can afford the processing of concurrent sessions. When the number of concurrent sessions increases to a certain threshold that can deteriorate the processing delay of the callee, it can adopt some policies (e.g., randomly dropping a session) to keep the session number below that threshold.

Call limit decoupling. The limit number of established call sessions shall be decoupled from that of call attempts for each phone device. Due to conventional phone design and usage practice, a phone can only make one call attempt at a time, though keeping multiple concurrent sessions with established calls is allowed. Herein, it is proposed that the IMS should consider them differently, instead of treating them as the same and causing vulnerability V4. In particular, it is suggested that the carriers can retain the same limit on the number of concurrent call sessions as currently used, but restrict the number of call attempts made by each phone to just one.

The overhead of this component is lightweight and can have little impact on the call service performance, since the IMS just needs to maintain two separate counters to restrict the numbers of concurrent call attempts and sessions. Notably, maintaining concurrent call sessions has been allowed in the current IMS system, so it is not considered as the overhead.

VI. Discussion

Roaming Impact on IMS Vulnerabilities. Vulnerability V1 exists only in the VoWiFi call service, so it cannot be exploited when a mobile device roams to use the other cellular voice solutions (e.g., VoLTE). However, it is not affected by the roaming between different WiFi networks, since a mobile device always connects to its home IMS system no matter which WiFi network it connects. To exploit another vulnerabilities V2/V3/V4, the adversary as the caller has to use the VoWiFi call service, similar to V1. For a roaming callee, which may roam to a visited network with an IMS system different from its home IMS or to the legacy CS call system, V2 and V4 can still take effect, since the IMS system of the caller, where the vulnerabilities are, can forward malicious call attempts to any roaming call system of the callee. For the exploitation of V3, the SIP messages generated by the adversary need to reach the callee device without conversion between different IMS systems or between the IMS and the legacy CS call systems, so the adversary as the caller needs to connect with the same IMS system as the callee; otherwise, the V3 cannot be exploited successfully. To use V5, both the caller and the callee have to connect with the same IMS system using the VoWiFi call service; they are allowed to roam between different WiFi networks.

Launching Attacks from VoLTE. There are two potential avenues: one is to hijack the VoLTE sessions established by the phone modem, whereas the other is to established them based on a customized UE with the software-defined radio. For the session hijacking, it is almost impossible from two aspects. First, if the adversary attempts to do session hijacking outside the modem, the corresponding security parameters are required, but they are hidden in the modem and hardly leaked out. Second, if the adversary seeks to take control of the modem for the session hijacking, some vulnerabilities

of the modem need to be discovered. For the second avenue with the customized UE, there shall be an IMS client to be developed on the UE; it needs to have the implementation of the IMS authentication procedure [8], [15] with the IMS-AKA and IPsec, as well as the communication with the ISIM module for the authentication and the generation of IPsec/SIP messages.

VII. Related Work

Cellular network security. Cellular network security is an active research area. Broadly speaking, the related studies can be classified into three main categories, besides IMS-related ones. First, several studies focus on security issues of cellular-specific network protocols and operations, such as LTE access networks with rogue base stations [27], layer-two protocols [28], misconfiguration [29], temporary identifier relocation [30], charging functions [31], and GSM encryption [32]. Second, some studies investigate security threats caused by Internet technologies and malicious traffic in the cellular network. Typical topics include middleboxes [33], malicious Internet traffic [34], [35], and botnets [36]. Third, many studies examine security issues of 3G services including CS-based calls [37], SMS [32], [38], [39], [40], [41], [42], and MMS (Multimedia Messaging Service) [43]. In contrast to these studies, the present work focuses on the problem of IMS security.

IMS security. Many studies have investigated security issues of IMS services, such as IMS-based SMS [12], VoLTE [10], [11], and VoWiFi [13], [44], [45], [46]. The study in [12] showed the feasibility of IMS-based SMS spoofing and its potential threats, whereas those in [10] and [11] investigated the possible resource abuse of VoLTE bearers in 4G networks. However, none of these studies explored the vulnerabilities of the IMS call system. Among the studies on VoWiFi, that in [44] examined the issue of man-in-the-middle attacks over VoWiFi, while that in [13] demonstrated the feasibility of stealing the IPsec keys used for VoWiFi using an SIM sniffer. In addition, the studies in [45] and [46] disclosed user privacy and launched DoS attacks by intercepting VoWiFi packets en route to/from the Internet. However, the attack models used in these prior studies assume that the adversary can intercept the VoWiFi packets sent by the victim's phone through virtue of being located in the same local area network. By contrast, the present study does not have such limitation.

Caller ID spoofing. Nowadays, caller ID spoofing is easily performed using third-party services [47], [48]. However, this study has identified a new vulnerability (V2) stemming from the IMS core, which allows an adversary to spoof the caller ID without the assistance of any third-party service by fabricating IPsec/SIP messages based purely on their attack phone. Many studies on caller ID spoofing have been performed. For example, the studies in [49], [50], and [51] propose methods for preventing spoofing by authenticating the caller ID using either a third-party entity, designated as the Phone Call Authority, or a cryptographic encryption and public key infrastructure (PKI) [50], [51]. By contrast, the studies in [52], [53], and [54] focus on the problem of detecting caller ID spoofing. Mustafa *et al.* [52] used a challenge-response mechanism performed in a cover channel accessible to only the call parties, while Deng and Peng [53] established a callback session upon each incoming call and then compared the call states of the outgoing and incoming calls, respectively. Finally, Sheoran *et al.* [54] leveraged the subscription data shared between the EPC and the IMS to carry out spoofing detection. However, despite the contributions of these studies, they are not used in practice due to their inconvenience.

SIP and VoIP security. Various security issues relating to the SIP protocol have been identified [18], [55], [56], [57], including eavesdropping, session hijacking, impersonation, message tampering, and DoS attacks. Most of these issues arise as the result of an absence of adequate authentication, confidentiality, or/and integrity functions. While several VoIP detection systems have been proposed to protect against intrusion [58] and DoS attacks [59], none of them provide defense against the vulnerabilities uncovered in this study.

VIII. Conclusion

Carriers have deployed the IMS system ever since the launch of VoLTE. The vulnerability of IMS has seldom been questioned since its access by a phone device is protected by hardware-based security. However, VoWiFi removes this security barrier due to its inherent design. This study has thus examined the vulnerabilities of VoWiFi and the corresponding security implications for IMS. It has been shown that the VoWiFi sessions can be hijacked by an adversary and then used to maliciously manipulate the IMS call operation. For example, the adversary can make ghost calls to launch stealthy call DoS attacks against cellular users given only a knowledge of their phone numbers. It has further been shown that a ML-assisted call DoS attack can be used to detect attackable phones at run time without gaining the attention of either the victim or the unattackable phones. Crucially, the security threats identified in this study apply to four top-tier carriers distributed across North America and Asia, respectively, and seven well-known phone brands. As a result, they call for immediate attention from global carriers, device vendors, and the cellular standard community.

References

[1] P. Cerwall, *Ericsson Mobility Report*. Stockholm, Sweden: Ericsson, 2018.
[2] *IP Multimedia Subsystem (IMS); Stage 2 (Release 15)*, Document TS23.228, V15.4.0, 3GPP, 2019.
[3] GSM Association, *IMS Profile for Voice, Video and SMS Over Untrusted Wi-Fi Access*, GSMA Official, Document IR.51, Version 5.0, 2017.
[4] *3GPP System Architecture Evolution (SAE); Security Aspects of Non-3GPP Accesses (Release 15)*, Document TS33.402, V15.1.0, 3GPP, 2018.
[5] M. Garcia-Martin, *Input 3rd-Generation Partnership Project (3GPP) Release 5 Requirements on the Session Initiation Protocol (SIP)*, Document IETF RFC 4083, 3GPP, 2005.
[6] R. Jesske, K. Drage, and C. Holmberg, *Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3GPP*, Document IETF RFC 7315, 3GPP, 2014.
[7] GSM Association, *IMS Profile for Voice and SMS*, GSMA Official Document IR.92, Version 11.0, 2017.
[8] *3G Security; Access Security for IP-based Services (Release 15)*, Document TS33.203, V15.1.0, 3GPP, 2018.
[9] S. Kent, *IP Encapsulating Security Payload (ESP)*, Document IETF RFC 4303, 2005.

[10] C.-Y. Li *et al.*, "Insecurity of voice solution VoLTE in LTE mobile networks," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, Denver, CO, USA, Oct. 2015, pp. 316–327.

[11] H. Kim *et al.*, "Breaking and fixing VoLTE: Exploiting hidden data channels and mis-implementations," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, Denver, CO, USA, Oct. 2015, pp. 328–339.

[12] G.-H. Tu, C.-Y. Li, C. Peng, Y. Li, and S. Lu, "New security threats caused by IMS-based SMS service in 4G LTE networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Vienna, Austria, Oct. 2016, pp. 1118–1130.

[13] S. Chalakkal, "How secure are your VoLTE and VoWiFi calls?" in *Proc. DeepSec Depth Secur. Conf. Eur. (IDSC)*, Vienna, Austria, Nov. 2017, pp. 1–62.

[14] J. Rosenberg and H. Schulzrinne, *Reliability of Provisional Responses in the Session Initiation Protocol (SIP)*, Document IETF RFC 3262, 2002.

[15] *IP Multimedia Call Control Protocol Based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3 (Release 16)*, Document TS24.229, V16.0.0, 3GPP, 2018.

[16] C. Madson and R. Glenn, *The Use of HMAC-SHA-1-96 Within ESP and AH*, Document IETF RFC 2404, 1998.

[17] G. Camarillo, W. Marshall, and J. Rosenberg, *Integration of Resource Management and Session Initiation Protocol (SIP)*, Document IETF RFC 3312, 2002.

[18] J. Rosenberg *et al.*, *SIP: Session Initiation Protocol*, Document IETF RFC 3261, 2002.

[19] (2022). *Verizon Conference Calling Services*. [Online]. Available: https://www.verizonwireless.com/support/calling-services/conference-calling/

[20] *Mandatory Speech CODEC Speech Processing Functions; AMR Speech CODEC; General Description*, Document TS26.071, V16.0.0, 3GPP, 2020.

[21] *Speech Codec Speech Processing Functions; Adaptive Multi-Rate-Wideband (AMR-WB) Speech CODEC; General Description*, Document TS26.171, V16.0.0, 3GPP, 2020.

[22] J. Sjoberg, M. Westerlund, A. Lakaniemi, and Q. Xie, *Real-Time Transport Protocol (RTP) Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs*, Document IETF RFC 3267, 2002.

[23] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, Apr. 1995.

[24] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA, USA: MIT Press, 2012, ch. 5, pp. 89–120.

[25] D. Harris and S. Harris, *Digital Design and Computer Architecture*, 2nd ed. Burlington, MA, USA: Morgan Kaufmann, 2012.

[26] K. Weinberger, A. Dasgupta, J. Langford, A. Smola, and J. Attenberg, "Feature hashing for large scale multitask learning," in *Proc. 26th Annu. Int. Conf. Mach. Learn. (ICML)*, Montreal, QC, Canada, 2009, pp. 1113–1120.

[27] A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J.-P. Seifert, "Practical attacks against privacy and availability in 4G/LTE mobile communication systems," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, 2016, pp. 1–16.

[28] D. Rupprecht, K. Kohls, T. Holz, and C. Popper, "Breaking LTE on layer two," in *Proc. IEEE Symp. Secur. Privacy (SP)*, Oakland, CA, USA, May 2019, pp. 1121–1136.

[29] M. Chlosta, D. Rupprecht, T. Holz, and C. Pöpper, "LTE security disabled: Misconfiguration in commercial networks," in *Proc. 12th Conf. Secur. Privacy Wireless Mobile Netw.*, Miami, FL, USA, May 2019, pp. 261–266.

[30] M. Arapinis, L. I. Mancini, E. Ritter, and M. Ryan, "Privacy through pseudonymity in mobile telephony systems," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, 2014, pp. 1–14.

[31] H. Hong *et al.*, "Pay as you want: Bypassing charging system in operational cellular networks," in *Proc. World Conf. Inf. Secur. Appl. (WISA)*, Jeju Island, (South) Korea, Aug. 2016, pp. 148–160.

[32] U. Meyer and S. Wetzel, "On the impact of GSM encryption and man-in-the-middle attacks on the security of interoperating GSM/UMTS networks," in *Proc. IEEE 15th Int. Symp. Pers., Indoor Mobile Radio Commun.*, Barcelona, Spain, Sep. 2004, pp. 2876–2883.

[33] Z. Wang, Z. Qian, Q. Xu, Z. Mao, and M. Zhang, "An untold story of middleboxes in cellular networks," in *Proc. ACM SIGCOMM Conf.*, Toronto, ONT, Canada, 2011.

[34] C. Lever, M. Antonakakis, B. Reaves, P. Traynor, and W. Lee, "The core of the matter: Analyzing malicious traffic in cellular carriers," in *Proc. IEEE Netw. Distrib. Syst. Secur. Symp. (NDSS)*, San Diego, CA, USA, Feb. 2013, pp. 1–16.

[35] P. Traynor *et al.*, "On attack causality in internet-connected cellular networks," in *Proc. USENIX Conf. Secur. (SEC)*, Boston, MA, USA, Aug. 2007, pp. 1–21.

[36] P. Traynor *et al.*, "On cellular botnets: Measuring the impact of malicious devices on a cellular network core," in *Proc. 16th ACM Conf. Comput. Commun. Secur. (CCS)*, Chicago, IL, USA, 2009, pp. 223–234.

[37] Y. Zheng, L. Huang, H. Shan, J. Li, Q. Yang, and W. Xu, "Ghost telephonist impersonates you: Vulnerability in 4G LTE CS fallback," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Las Vegas, NV, USA, Oct. 2017, pp. 1–9.

[38] B. Reaves, N. Scaife, D. Tian, L. Blue, P. Traynor, and K. R. B. Butler, "Sending out an SMS: Characterizing the security of the SMS ecosystem with public gateways," in *Proc. IEEE Symp. Secur. Privacy (SP)*, San Jose, CA, USA, May 2016, pp. 339–356.

[39] P. Traynor, W. Enck, P. McDaniel, and T. La Porta, "Mitigating attacks on open functionality in SMS-capable cellular networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 1, pp. 40–53, Feb. 2009.

[40] N. Saxena and N. S. Chaudhari, "EasySMS: A protocol for end-to-end secure transmission of SMS," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1157–1168, Jul. 2014.

[41] W. Enck, P. Traynor, P. McDaniel, and T. La Porta, "Exploiting open functionality in SMS-capable cellular networks," in *Proc. 12th ACM Conf. Comput. Commun. Secur. (CCS)*, Alexandria, VA, USA, 2005, pp. 393–404.

[42] A. Bose, X. Hu, K. G. Shin, and T. Park, "Behavioral detection of malware on mobile handsets," in *Proc. 6th Int. Conf. Mobile Syst., Appl., Services*, Breckenridge, CO, USA, 2008, pp. 225–238.

[43] R. Racic, D. Ma, and H. Chen, "Exploiting MMS vulnerabilities to stealthily exhaust mobile phone's battery," in *Proc. Securecomm Workshops*, Baltimore, MD, USA, Aug. 2006, pp. 1–10.

[44] J. G. Beekman and C. Thompson, "Breaking cell phone authentication: Vulnerabilities in AKA, IMS and android," in *Proc. 7th USENIX Workshop Offensive Technol. (WOOT)*, Washington, DC, USA, Aug. 2013, pp. 1–10.

[45] T. Xie, G.-H. Tu, C.-Y. Li, C. Peng, J. Li, and M. Zhang, "The dark side of operational Wi-Fi calling services," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Beijing, China, May 2018, pp. 1–9.

[46] J. Baek *et al.*, "Wi not calling: Practical privacy and availability attacks in Wi-Fi calling," in *Proc. ACM Annu. Comput. Secur. Appl. Conf. (ACSAC)*, San Juan, PR, USA, Dec. 2018, pp. 278–288.

[47] (2022). *SpoofTel*. [Online]. Available: https://www.spooftel.com/

[48] (2022). *SpoofCard*. [Online]. Available: https://www.spoofcard.com/

[49] J. Li, F. Faria, J. Chen, and D. Liang, "A mechanism to authenticate caller ID," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, 2017, pp. 745–753.

[50] B. Reaves *et al.*, "AuthentiCall: Efficient identity and content authentication for phone calls," in *Proc. USENIX Secur. Symp.*, Vancouver, BC, Canada, Aug. 2017, pp. 575–592.

[51] H. Tu, A. Doupe, Z. Zhao, and G.-J. Ahn, "Toward standardization of authenticated caller ID transmission," *IEEE Commun. Standards Mag.*, vol. 1, no. 3, pp. 30–36, Sep. 2017.

[52] H. Mustafa, W. Xu, A.-R. Sadeghi, and S. Schulz, "End-to-end detection of caller ID spoofing attacks," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 3, pp. 423–436, May/Jun. 2016.

[53] H. Deng and C. Peng, "Combating caller ID spoofing on 4G phones via CEIVE," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, New York, NY, USA, Oct. 2018, pp. 846–848.

[54] A. Sheoran, S. Fahmy, C. Peng, and N. Modi, "NASCENT: Tackling caller-ID spoofing in 4G networks via efficient network-assisted validation," in *Proc. INFOCOM IEEE Conf. Comput. Commun.*, Paris, France, Apr. 2019, pp. 676–684.

[55] S. El Sawda and P. Urien, "SIP security attacks and solutions: A state-of-the-art review," in *Proc. 2nd Int. Conf. Inf. Commun. Technol.*, Damascus, Syria, 2006, pp. 3187–3191.

[56] D. Sisalem, J. Kuthan, and S. Ehlert, "Denial of service attacks targeting a SIP VoIP infrastructure: Attack scenarios and prevention mechanisms," *IEEE Netw.*, vol. 20, no. 5, pp. 26–31, Sep. 2006.

[57] W. Werapun, A. A. El Kalam, B. Paillassa, and J. Fasson, "Solution analysis for SIP security threats," in *Proc. Int. Conf. Multimedia Comput. Syst.*, Ouarzazate, Morocco, Apr. 2009, pp. 174–180.

[58] H. Sengar, D. Wijesekera, H. Wang, and S. Jajodia, "VoIP intrusion detection through interacting protocol state machines," in *Proc. Int. Conf. Dependable Syst. Netw. (DSN)*, Philadelphia, PA, USA, 2006, pp. 393–402.

[59] H. Sengar, H. Wang, D. Wijesekera, and S. Jajodia, "Fast detection of denial-of-service attacks on IP telephony," in *Proc. 14th IEEE Int. Workshop Quality Service*, New Haven, CT, USA, Jun. 2006, pp. 199–208.