5GHOUL : Unleashing Chaos on 5G Edge Devices

VULNERABILITY DISCLOSURE REPORT

By

Matheus E. Garbelini (SUTD)
Zewen Shang (SUTD)
Shijie Luo (SUTD)
Sudipta Chattopadhyay (SUTD)
*Sumei Sun (I2R, A*STAR)*
*Ernest Kurniawan (I2R, A*STAR)*

Supported by

NRF NATIONAL SATELLITE OF EXCELLENCE IN DEST-SCI
FUTURE COMMUNICATIONS PROGRAM

December 07, 2023, UTC -12

5GHOUL : Unleashing Chaos on 5G Edge Devices

Singapore University of Technology and Design (SUTD) *I2R, A*STAR*

Updates

⇒ **29th December of 2023:** After additional severity assessment review from MediaTek, three vulnerabilities have been elevated from Medium to High severity rating. (CVE-2023-32842, CVE-2023-32841 and CVE-2023-32843).

⇒ **15th December of 2023:** Our study to estimate the affected smartphones considers phone models that employ the vulnerable 5G modems. However, certain phones may not enable the 5G functionality. We performed further analysis to estimate the number of smartphones that also offer 5G features and amended the study (see Section 4).

⇒ **7th December of 2023:** As of today, we disclose a total of 14 vulnerabilities (10 CVEs), of which 10 affect commercial off-the-shelf (COTS) edge devices employing 5G modems from *Qualcomm* and *MediaTek*. At least two other vulnerabilities are not disclosed yet due to confidentiality. This page will continue to be updated as our experiment progresses and the embargo is lifted. For more details on each disclosed vulnerability, please refer to Section 7. Feel free to contact us via contact@5ghoul.com.

⇒ **4th December of 2023:** Following a responsible disclosure period, the December security bulletin of MediaTek includes details of all MediaTek Chipsets affected by 5GHOUL vulnerabilities [10]. Likewise, all Qualcomm Chipsets affected by 5GHOUL vulnerabilities are detailed in December security bulletin of Qualcomm [13].

Summary

In this vulnerability disclosure report, we discuss details of 5GHOUL – a family of implementation-level 5G vulnerabilities. Such a family of vulnerabilities are present in the firmware implementation of 5G mobile network modems from major chipset vendors i.e., Qualcomm and MediaTek. Consequently, many 5G-capable commercial products such as smartphones, Customer-premises Equipment (CPE) routers and USB modems are potentially impacted due to the employment of vulnerable 5G modems in such products. *In total, we have found 12 new vulnerabilities (14 total), out of which 10 affect 5G modems from Qualcomm and MediaTek.* More importantly, *six* of these ten vulnerabilities are confirmed to have high severity. We also wrote a scraper to send crafted queries to https://www.kimovil.com/en/ and to have an estimate on the number of smartphone models affected due to these vulnerabilities. We found over *710 smartphone models that are currently in the market to employ the vulnerable modems. Further considering the 5G capability of these phones employing the vulnerable chipsets, we found over* 626 phone models *to be affected.* We emphasize that the actual number of affected models might be more, as firmware code is often shared across different modem versions. In this disclosure report, *we also demonstrate the exploitation of* 5GHOUL *vulnerabilities to drop and freeze 5G connection on smartphones and CPE routers. We also show downgrade attacks across multiple smartphones that result in downgrading the 5G connection to 4G.*

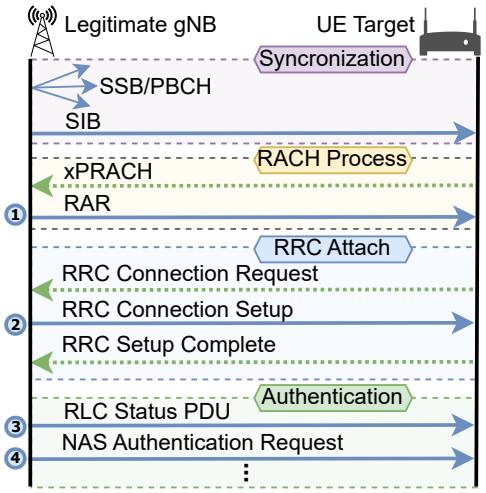


Figure 1: An Illustration of the 5G Standalone (SA) connection procedure between legitimate (i.e., benign) gNB and UE (e.g., 5G smartphone).

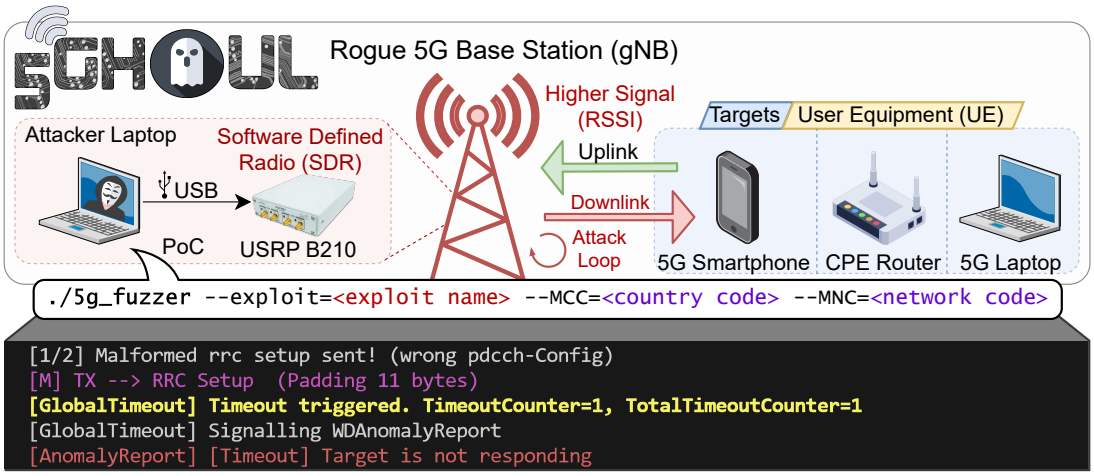


Figure 2: Illustration of 5GHOUL attack: Once the target connects to the rogue base station (gNB), the attacker simply launches the exploit script as shown in the command prompt with the Mobile Country Code (*MCC*) and the Mobile Network Code (*MNC*).

1 Introduction

Mobile communications are indispensable in our daily lives, and the advent of 5G networks has brought several new opportunities for low-latency communication in critical domains such as internet-of-things (IoT), virtual reality, and medical and automation industries. However, potential vulnerabilities in 5G networks can undermine trust in their security, necessitating comprehensive validation of 5G protocol stacks prior to deployment. In this context, 5GHOUL showcases implementation-level vulnerabilities existent in flagship 5G modems, thus highlighting the need for tools and technologies specifically targeted to test the robustness of 5G devices. Moreover, 5GHOUL pinpoints security vulnerabilities down to the earliest protocol decoding and handling, further emphasizing the requirement for validating the security of early-stage 5G communication down to data link protocols (OSI layer 2).

In the following, we provide a brief overview on the architecture of 5G system and procedures involved in 5G communication. This is to provide the necessary background knowledge for this disclosure report. Firstly, the 5G cellular network architecture consists of three key components: The gNodeB (gNB), User Equipment (UE), and Core Network. The gNB is also known as the base station in traditional cellular network. It serves as the access point for wireless communication between the UE and the 5G core network. The UE refers to end-user devices, such as smartphones and tablets (in general, any 5G-enabled device), that connect to the 5G network through the gNB. Lastly, the Core Network acts as the backbone of the 5G architecture by providing control and management functions, including authentication, security, mobility management, session establishment, and data routing between network entities.

Figure 1 illustrates a clean connection process between a 5G UE device (e.g., a smartphone) and a legitimate gNB. Multiple protocols including Radio Resource Control (RRC), Non-Access Stratum (NAS), Medium Access Control (MAC), Packet Data Convergence Protocol (PDCP) and Radio Link Control (RLC) from both network layer (OSI layer 3) and data link layer (OSI layer 2) are involved to ensure that the connection is established successfully and securely.

A notable feature of currently disclosed 5GHOUL vulnerabilities is that they can all be *fairly easy to exploit by an attacker, as no information about the victim's SIM card is required.* This is because 5GHOUL vulnerabilities can be exploited to launch an attack even before any NAS authentication procedure completes. The attacks exploiting 5GHOUL vulnerabilities will be illustrated in Section 4. We discuss the availability of the 5GHOUL PoC along with the necessary tooling in Section 5.

Why 5GHOUL ?

The code name 5GHOUL is a combination of two words: *(i)* 5G and *(ii)* Ghoul. While the connection of our work with 5G is clear, the word *Ghoul* is derived from Arabian mythology (https://www.britannica.com/topic/ghoul). In popular legend, *Ghoul* is a demon like creature which tries to distract travelers, and, upon successfully distracting them, the demon preys on them. Likewise, the 5GHOUL vulnerabilities, when exploited, try to distract the UEs (i.e., smartphones and other 5G-enabled devices) to connect to 5GHOUL created malicious base station (gNB). Once connected, 5GHOUL vulnerabilities may be exploited to continuously launch attacks to drop the connections, freeze the connection that involve manual reboot or downgrade the 5G connectivity to 4G.

2 Attack Scenario Overview

As illustrated in Figure 2, 5GHOUL makes use of an attacker model which mimics a *limited* Dolev-Yao adversary [5, 6, 8]. This is accomplished by exposing an *Adversary-Controlled Downlink channel* that can arbitrarily inject and/or modify 5G NR Downlink Packets generated from a real 5G stack implementation based on OpenAirInterface [11] (gNB) and Open5GS [1] (5G Core Network).

More importantly, *the attacker does not need to be aware of any secret information of the target UE e.g., SIM card details, to reach the beginning of the NAS network registration. The attacker only needs to impersonate the legitimate gNB using the known Cell Tower connection parameters (e.g., SSB ARFCN, Tracking Area Code, Physical Cell ID, Point A Frequency).* This can be easily accomplished using freely available applications like Cellular-Pro. Once the attacker is sufficiently close to the target UE and the *Received Signal Strength Indicator* (RSSI) of the adversarial gNB is higher than the legitimate gNB, the target UE will connect to the adversarial gNB. Then, the UE starts exchanging messages up to step 4 of Figure 1. Procedures that appear later are subjected to failure since key information from UE's SIM card is unknown. However, throughout the message exchanges, the adversarial gNB (see Figure 2) can freely manipulate downlink messages to the target UE, opening a window of opportunities to launch attacks at any step of the 5G NR procedures shown in Figure 1.

In practicality, 5GHOUL vulnerabilities can be easily exploited over-the-air by starting a malicious gNB within radio range of the target 5G UE device (see Figure 2). This is a practical setup which relies on using Software Defined Radio (SDR) to behave as a cloned gNB. While USRP B210 used in our setup could be recognized from afar, thus making the attack visually noticeable, such type of equipment has already been miniaturized to the size of a Raspberry Pi [15]. This, in turn, enables the use of SDR for visibly stealthy attacks.

3 Vulnerable 5G Modems and Products

In this section, we provide some salient characteristics of the found vulnerabilities and vulnerable devices.

Vulnerability Summary: The vulnerable targets used in our evaluation and their corresponding firmware version are outlined in Table 1. Additionally, Table 2 outlines all 5GHOUL vulnerabilities on 5G devices. In the first column, each vulnerability name is identified with prefix ***V***. While the first two vulnerabilities (***V1*** and ***V2***) affect an open-source UE implementation from *OpenAirInterface* project [11], the rest of the vulnerabilities were tested on many popular 5G USB Modems (***V3-V7***) or representative Smartphones employing *Qualcomm* or *MediaTek* modems (***V5-V14***).

It is worthwhile to note that our discovered vulnerabilities (***V3-V14***) are within the 5G baseband modem firmware. Thus, any product employing the respective Qualcomm and MediaTek modem firmware are affected (although the impact may vary depending on the type of product). Moreover, during our experiments, vulnerabilities V3 and V4 were discovered in certain older (before 2023) firmware versions employed in *SIMCOM* and *FIBOCOM* modems (see Table 2). We verified that such issues do not affect products (e.g., 5G USB modems and smartphones) that had firmware with build date in 2023. Finally, ***V5-V14 affect the latest modem firmware at the time of experiments and therefore, need patching for the CVEs illustrated in Table 2 to prevent exploitation.***

In our findings, *Qualcomm* 5G USB Modems and 5G-enabled Smartphones such as *Asus ROG Phone 5S* were impacted by a range of previously unknown vulnerabilities in the handling of *MAC/RLC*, *RRC* and *NAS* messages (see ***V5-V7*** in Table 2). *All these vulnerabilities were confirmed by Qualcomm to have High severity.* MediaTek modem enabled smartphones such as OnePlus Nord CE 2 are affected by previously unknown vulnerabilities ***V8-V14*** (see Table 2). In particular, vulnerabilities V10 and V11 capture memory access violations in terms of *null pointer dereference*. *While most vulnerabilities were confirmed to have medium severity, V11-V13 were upgraded to high severity after further security assessment review from MediaTek on 29th December 2023.*

Vulnerable Procedures: In summary, the *RRC Attach* and *Authentication* procedures (see Figure 1) exhibited all 5GHOUL vulnerabilities. In particular, *RRC Attach* procedure, which contains the *RRC Connection Setup* message, was focus of most vulnerabilities. Such findings are prevalent for *OnePlus Nord CE 2*. With such *UE* employing *MediaTek Dimensity 900 5G* modem, many asserts and memory-related crashes (see Table 2) were triggered during the exchange of *RRC Setup Connection*.

Lastly, all 5GHOUL vulnerabilities were found during the pre-authentication stage of the communication between UE and gNB. This means that attacks exploiting ***V1-V14*** do not require any secret information from the UE's SIM card to be successful. Such attacks can be launched by starting a malicious gNB with the same setup as shown in Section 2 (see Section 2). Overall, 5GHOUL not only captures 5G data link implementation vulnerabilities (***V4, V5***, ***V10***), but also demonstrates vulnerabilities at OSI Layer 3 and above (e.g., *RRC*, *NAS*) for COTS 5G UEs.

Triggering the Vulnerabilities: Overall, the vulnerabilities summarized in Table 2 are triggered by either sending a malformed *RRC Connection Setup* (albeit writing to different RRC fields as depicted in Figure 3) or sending a malformed *NAS Authentication Request*. In the particular case of ***V2***, the *NAS Authentication Request* is malformed such that the actual NAS message payload is empty. In contrast, only ***V4*** requires sending both an invalid *MAC Time Advance Command* and a malformed *RRC Connection Setup*.

Table 1: Devices and Monitoring used for evaluation of 5GHOUL vulnerabilities. The firmware/software is provided by vendor or through an OS update. *OpenAirInterface UE* is a software solution, not employing a 5G modem (N.A).

Vendor / Product	5G Modem	Type	Monitor	Firmware/Software Version
OpenAirInterface UE	N.A	Software	ProcessMonitor	2023.w03
Quectel RM500Q-GL	Qualcomm X55	USB Modem	ModemManager	Aug 03 2021
Simcom SIM8202G	Qualcomm X55	USB Modem	ModemManager	SIM8202G-M2_V1.2
Fibocom FM150-AE	Qualcomm X55	USB Modem	ModemManager	89602.1000.00.04.07.20
Telit FT980m	Qualcomm X55	USB Modem	ModemManager	38.23.001-B001-P0H.000640
OnePlus Nord CE 2 5G	Dimensity 900 5G	Smartphone	ADB	M_V3_P10
Xiaomi Redmi K40	Dimensity 1200 5G	Smartphone	ADB	MOLY.NR15.R3.TC8.PR2.SP.V2.1.P70
Asus ROG Phone 5s	Qualcomm X60	Smartphone	ADB	M3.13.24.73-Anakin2
Samsung S22 (5G)	Qualcomm X65	Smartphone	ADB	S901EXXU4CWCE

Table 2: Summary of 5G Implementation Vulnerabilities and affected software or products.

Implementation Vulnerability	Affected Modems/Smartphones	Protocols	Impact	CVE Status
V1 - Invalid PUSCH Resource Allocation	OAI UE	RRC	DoS	Pending
V2 - Empty RRC dedicatedNAS-Message	OAI UE	RRC, NAS	DoS	Pending
V3 - Invalid RRC Setup	Fibocom FM150-AE Simcom SIM8202G	RRC	DoS	Patched
V4 - Invalid RRC Reconfiguration	Simcom SIM8202G	MAC, RRC	DoS	Patched
(7.1) V5 - Invalid MAC/RLC PDU	Telit FT980m Simcom SIM8202G Asus ROG Phone 5s	MAC, RLC	DoS	CVE-2023-33043
(7.2) V6 - NAS Unknown PDU	Telit FT980m Fibocom FM150-AE Asus ROG Phone 5s	NAS	DoS	CVE-2023-33044
(7.3) V7 - Disabling 5G / Downgrade via RRC	Telit FT980m Asus ROG Phone 5s Simcom SIM8202G Fibocom FM150-AE Quectel RM500Q-GL	RRC	Hang/Downgrade	CVE-2023-33042
(7.4) V8 - Invalid RRC Setup spCellConfig	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32842
(7.5) V9 - Invalid RRC pucch CSIReportConfig	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32844
(7.6) V10 - Invalid RLC Data Sequence	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RLC	DoS	CVE-2023-20702
(7.7) V11 - Truncated RRC physicalCellGroupConfig	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32846
(7.8) V12 - Invalid RRC searchSpacesToAddModList	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32841
(7.9) V13 - Invalid RRC Uplink Config Element	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32843
(7.10) V14 - Null RRC Uplink Config Element	OnePlus Nord CE 2 5G Xiaomi Redmi K40	RRC	DoS	CVE-2023-32845

Patch Status: Patching 5G modem vulnerabilities involves a complex process until the patches become available to end users. We reflect on this process in more detail in Section 4.5. The patches for vulnerabilities *V5-V14* have taken reasonable time to be available to end users. In particular, for Android-based smartphones, the patches are expected to be available in ***December 2023***. For Apple-based iPhones, however, the patch schedule is not aligned with the December timeline. Nonetheless, *Qualcomm has confirmed that patches were available to their customers (i.e., device makers) starting in August 2023. Likewise, MediaTek has also stated that the patches were available to their OEM partners two months before the December bulletin [10].* Therefore, end users are advised to apply respective security updates as they become available from device makers.

4 Exploitation of 5GHOUL vulnerabilities

We created different concrete attacks leveraging the 5GHOUL vulnerabilities. In the following, we discuss attacks that cause Denial of Service (DoS) or connectivity Downgrade on affected target devices.

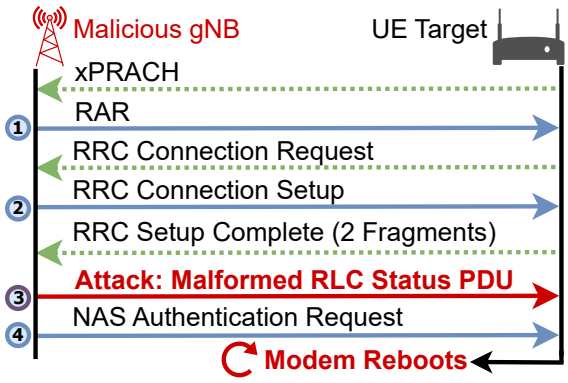


Figure 3: Illustration ***V5*** - *"Invalid MAC/RLC PDU"*

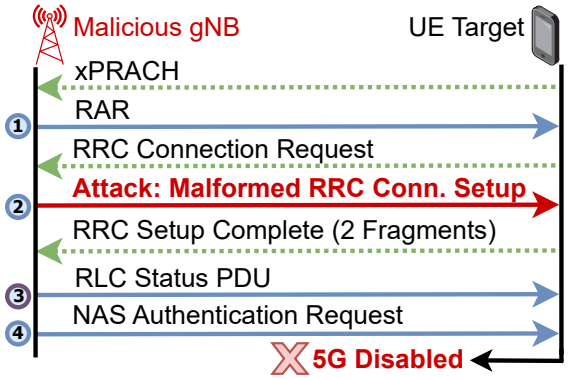


Figure 4: Illustration ***V7*** - *"RRC Invalid pdcch-Config"*

4.1 Exploitation on Mobile Devices

To test the impact of 5G vulnerabilities on mobile devices and hence user experience, we exploit vulnerabilities *V5* to *V10* against *Asus ROG Phone 5S (ARP5s)*, which uses Qualcomm Modem and *OnePlus Nord CE 2 (OnePlus)*, which uses MediaTek Modem (see Table 1). First, when vulnerability **V5** (see Figure 3) or **V6** is triggered on *ARP5s*, its 5G modem immediately reboots and automatically recovers connection to gNB in few seconds (temporary DoS). Hence, an attacker exploiting ***V5*** and ***V6*** must continuously launch the attacks if the intention is to completely disrupt mobile network connectivity of the user. Additionally, 3G and 4G communication are also disrupted upon modem reboot since the modem handles all *3GPP* related communication.

More surprisingly, vulnerability ***V7*** (see Figure 4) can prevent *ARP5s* to connect to any 5G network, while keeping 4G/3G connectivity intact (*i.e., downgrade attack*). Nevertheless, such behavior of ***V7*** highlights that the mobile device modem enters an erratic state such that the user needs to manually reboot the phone, thus power cycling the modem to fully restore 5G connectivity. As illustrated in the Video 1, the impact of this attack (i.e., freezing 5G connectivity) persists even after the attack stops. Moreover, in certain cases, manually rebooting the phone does not restore the connectivity either, instead, the SIM card requires to be taken out manually and then re-inserted to ensure a hard reset on the modem.

Video 1: Demonstration of *"Disabling 5G connection via Invalid RRC pdcch-Config"* (CVE-2023-33042). Youtube Video URL: https://youtu.be/DiGqwgBCSXM

As demonstrated in Video 1, initially, *ARP5s* is connected to a legitimate gNB (shown as ***SUTD 00101*** before the attack). We then start the malicious gNB and *V7* is triggered by starting 5GHOUL framework (exploit `mac_sch_rrc_setup_crash_var`). *Although the video illustrates the attack using a Software Defined Radio (SDR) for legitimate gNB, we successfully launched the attack even when the smartphone was connected to a commercial 5G base station.* To this end, we used the commercial base station provided by Quanta Cloud Technology (QCT) in the FCCLab (https://fcp.sutd.edu.sg/fcclab-details/) under Future Communications Program. The attack results in manually rebooting affected smartphones to recover any 5G connectivity, even if connected to QCT 5G network.

Lastly, vulnerabilities ***V8-V14*** trigger crashes on *OnePlus*, which employs *MediaTek Dimensity 900 5G Modem*. More specifically, ***V8, V9*** and ***V12-V14*** trigger reachable asserts in the internal microcontroller and DSP of the 5G modem. Likewise, ***V10*** produces an invalid memory access exception due to a *null pointer dereference*. For all cases, the modem immediately reboots and takes a few seconds to recover 5G connectivity. Similar to vulnerabilities ***V5*** and ***V6***, an attacker can continuously launch the attacks to keep disrupting all 3G/4G/5G communications on *OnePlus*.

4.2 Exploitation on Specialized 5G Products

We note that vulnerabilities ***V5-V14*** (Table 2) affect 5G devices employing Qualcomm and MediaTek modems. Thus, ***V5-V14*** affect not only smartphones and USB modems, but also appliances that rely on low-latency communication.

To assess the practical impact of 5GHOUL vulnerabilities

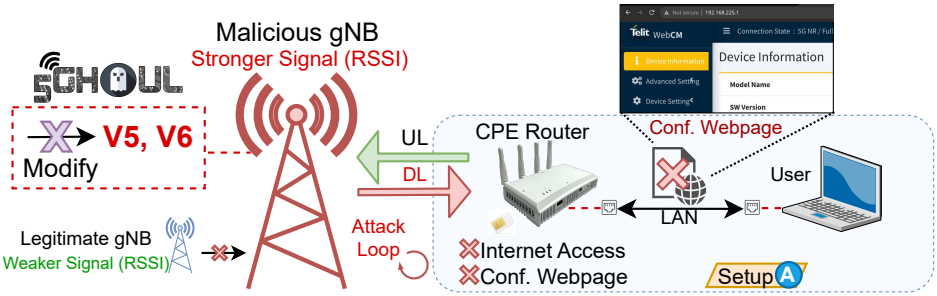


Figure 5: Impact of exploiting ***V5, V6*** on *Telit FT980m* operating as CPE Router (**Setup A**)

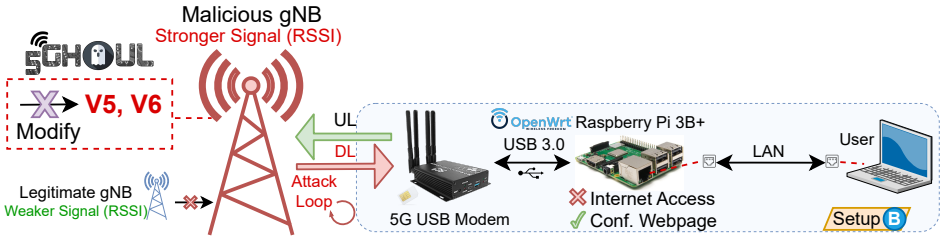
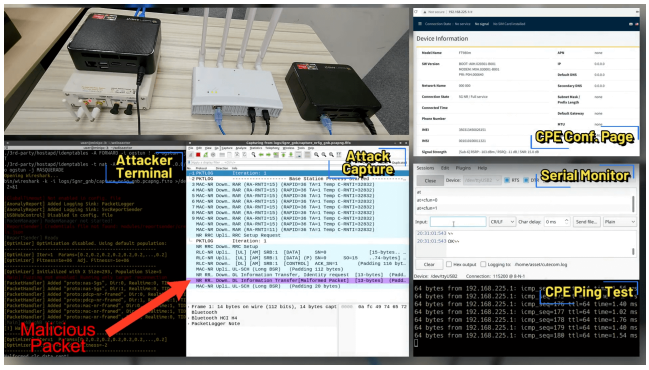


Figure 6: Impact of exploiting ***V5, V6*** on *RM500Q-GL* modem operating as CPE Router (**Setup B**)

beyond smartphones, we conducted exploit tests on specific 5G UE Modems listed in Table 1 and analyzed the impact on applications relying on these modems. Two Qualcomm-based Modems were configured as Customer Premises Equipment (CPE) to evaluate the impact of **V5** to **V7**. The exploitation setup is illustrated in Figure 5. Specifically, in ***Setup A*** (see Figure 5), the Telit *FT980m* modem (see Table 2) within the *FT980WW* platform [14] was tested. Such a platform provides 5G internet connectivity through its LAN port and hosts a *Configuration Webpage*. We launched continuous attacks within radio range, exploiting vulnerabilities ***V5*** and ***V6*** from a malicious gNB. ***This resulted in a complete loss of internet connectivity for devices connected to the FT980WW LAN port, and the Configuration Webpage became inaccessible due to modem reboots caused by attacks exploiting V5 and V6.*** A demonstration of this attack is showcased in Video 2.

In ***Setup B*** (shown in Figure 6), a Raspberry Pi 3B+ with *OpenWRT 22.03.4* [12] and *RM500Q-GL* 5G USB Modem (see Table 1) were used. This setup offered better isolation against attacks exploiting ***V5*** and ***V6*** since the *Configuration Webpage* was hosted by the *Raspberry Pi 3B+* processor instead of the modem itself. This allowed the user to remotely attempt to steer the CPE device by selecting a different mobile network to connect.

During our tests, the *FT980-WW* CPE exhibited more proactive attempts to recover 5G connectivity after attacks exploiting ***V5*** and ***V6***, while the *Raspberry Pi 3B+ with OpenWRT* was less proactive, often requiring manual reconnection of the 5G Modem via *OpenWrt's Webpage*. Such a reconnection instability involves manually rebooting via OpenWrt's Webpage and restore proper 5G connectivity when ***V5-V6*** stopped. On the contrary, since *FT980-WW* runs the web application within its 5G Modem, the user was unable to access the *Webpage*. As a result, the user cannot steer the CPE from the malicious gNB while the attack was still ongoing. This means the user intervention does not apply when trying to recover from the attack.



Video 2: Demonstration of *"NAS unknown PDU"* vulnerability on a Telit CPE Router (CVE-2023-33043). Youtube Video URL: https://youtu.be/_g4P7iSuFNk

For attack ***V7***, its effects were immediate, causing the targeted modem to disconnect from any 5G mobile network and fallback to 4G networks, exposing potential vulnerabilities in the 4G domain [9]. This made the CPE router to likewise

downgrade its service and access the Configuration Webpage to be available only in 4G speed.

4.3 Downgrade Attacks

While 5GHOUL reveals a set of previously unknown implementation vulnerabilities which cause denial of service (i.e., crash or hang), we showcase the capability of 5GHOUL exploitation tool to perform downgrade attacks. Such a class of vulnerability is unique in its effect on modems. This is because downgrade attacks often lead the connectivity of the user to forcibly change, specifically from a newer network technology (e.g., 5G NR) to an older one (e.g., LTE or below). Thus, downgrade attacks potentially expose the user to a different set of design or implementation issues inherently to a network technology (e.g., 2G, 3G and 4G). In this context, vulnerability *V7* (7.3) is understood as a downgrade attack because its exploitation prevents affected devices from connecting to any 5G network, while connection to older network technologies such as 4G is still possible.

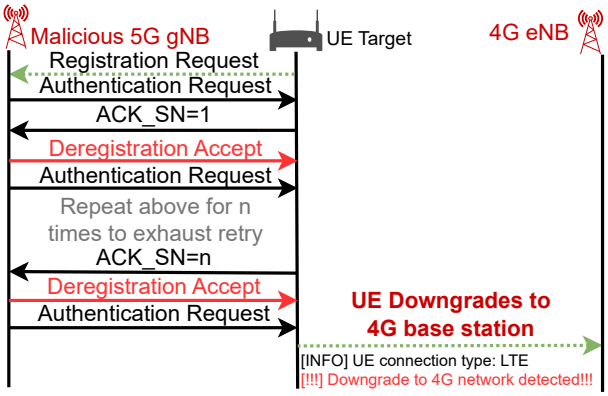


Figure 7: Downgrade via *NAS flooding*.

Nevertheless, *V7* is an implementation vulnerability which only affects devices employing Qualcomm X55/X60 modems. However, there are "general" downgrade attacks that can potentially affect all 5G devices due to the vulnerable components existing in the 3GPP standard itself [2]. During our exploration of downgrade attacks, we were able to trigger known general downgrade attacks [3]. Moreover, ***we exploited a new downgrade attack vector (DA1), which affects smartphones employing 5G baseband modems from multiple vendors including Qualcomm and MediaTek.*** Specifically, we successfully launched the new downgrade attacks on Xiaomi Redmi K40 (MediaTek Dimensity 1200), OnePlus Nord CE 2 (MediaTek Dimensity 900 and MediaTek Dimensity 1200), Samsung Galaxy S22 (Qualcomm X65) and Asus ROG Phone 5s (Qualcomm X60).

Our downgrade attack vector **DA1** is illustrated in Figure 7 and in Video 3. In this attack, the target device is kept in a *NAS*

Video 3: Demonstration Video of Downgrade Attack via NAS flooding (see Figure 7). Youtube Video URL: https://youtu.be/84DDy_4ofxw

Authentication procedure loop by receiving many (i.e., flooding) *NAS Authentication Request*, followed by *NAS Deregistration Accept* many times. After the target device goes through an arbitrary number of retries in this scenario, it falls back to connect to the nearest 4G base station. *Notably, the UE cannot re-establish the 5G connection by merely toggling the airplane mode; it necessitates initiating a new connection process with the base station.* Such downgrade attack highlights the capability of 5GHOUL exploitation to be used as a valuable tool for 5G NR security testing beyond DoS vulnerabilities. Such tooling capabilities of launching both attacks and test cases are detailed in Section 5.

A demonstration of 5GHOUL exploitation tool triggering the new downgrade attack is show in Video 3. In summary, an attacker can use the same setup as discussed in Section 2. The execution of this attack requires both a legitimate 5G and 4G base stations (*gNB* and *eNB* respectively). To this end, we setup, three SDRs in total. two to simulate the legitimate base stations and another to simulate an attacker within radio range of the target UE (OnePlus Nord CE 2 5G).

4.4 Estimating the reach of 5GHOUL

To estimate the potential number of 5G smartphones affected by 5GHOUL , we search for the model of all the smartphones that employ vulnerable 5G modems from Qualcomm and MediaTek. Notably, we relied on web scraping to automate the process of finding phone models matching specific processor chipsets. This is because mobile processor vendors usually provide to the smartphone vendor a chipset platform that already integrates a CPU, 5G modem, GPU, amongst other peripherals. For example, smartphones employing a Qualcomm processor contains a ***Snapdragon 8XX Mobile Platform***, while MediaTek-based 5G phones contains a ***Dimensity XXXX*** chipset.

To find potentially affected smartphone models, we based our search on Kimovil website, which makes it easy to filter all released smartphone models by a given processor chipset name. The models of affected chipsets, in turn, can be found in the vendor December 2023 security bulletins [10, 13]. Nonetheless, the security bulletin of Qualcomm [13] lists other affected (IoT) chipset platforms than used in smartphones. Thus, we only extract the chipset platforms (e.g., Snapdragon 8XX) from this bulletin. Unfortunately, it is not possible for us to map the chipset names released in MediaTek security bulletin [10] with the *Dimensity XXXX* chipset platform. Therefore, we only search for MediaTek-based phones employing the exact chipset names used in our evaluation (*Dimensity 900/1200*) to avoid false positives. Consequently, our computed set of smartphone models, which are affected by MediaTek modem vulnerabilities, is an underestimation.

Once we have all the chipset names listed, we use a scraping script to automate the search for all smartphone models employing the affected chipsets. Since many results presented in Kimovil web page are duplicated, we filter out smartphone entries that contain duplicated names, but variate only slightly in their specification (e.g., RAM and storage size). The summarized number of affected smartphones models across the market is shown via the pie chart of Figure 8. Notably, while most of the smartphone models e.g., ***Samsung, OnePlus, Oppo, Vivo, Xiaomi etc.*** in our estimation use chipsets from Qualcomm (670 out of 714 i.e., **94%**), we were only able to match a few phones using MediaTek's *Dimensity 900* and *Dimensity 1200*. As discussed in the preceding paragraph, the set of MediaTek-based affected smartphones is quite an underapproximation due to the limited search. The detailed list of smartphone models is found in the following URL: 5GHOUL affected smartphones. We also checked whether these phones use at least one 5G band for the attacks to be feasible to launch. We found over 626 phones amongst the listed smartphones in Figure 8 to support 5G band.

Finally, it is worthwhile to mention that the reach of 5GHOUL vulnerabilities goes beyond smartphones. This is because the affected 5G modems are also used in other 5G-enabled applications. Indeed, as detailed in Qualcomm Security Bulletin [13], ***5GHOUL vulnerabilities potentially affect Industrial IoT solutions such as "315 5G IoT" as well as other platforms e.g., "AR8035" used in diverse applications including home appliances, IP Cameras etc.***

4.5 The Challenge of Delivering 5G Patches to the End-user

It is critical to ensure that the modem software development kit (SDK) is well tested and devoid of serious vulnerabilities before being released downstream. Otherwise, attackers may exploit a modem failure for a prolonged period and before the end user can actually pull the relevant security updates.

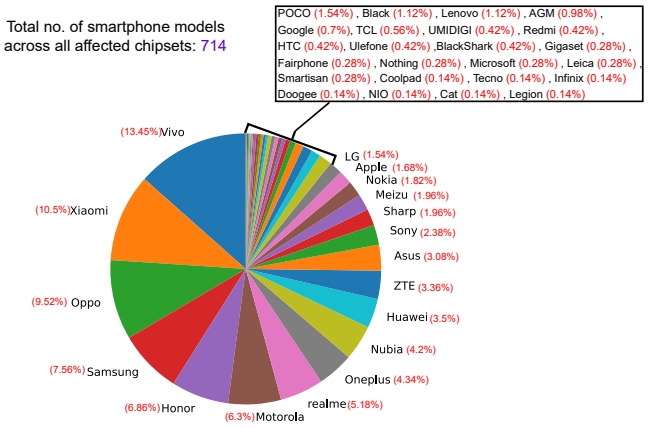


Figure 8: Distribution of smartphone models potentially affected by *5Ghoul*. Data based on Qualcomm and MediaTek security disclosures [10, 13] and Kimovil smartphone listings.

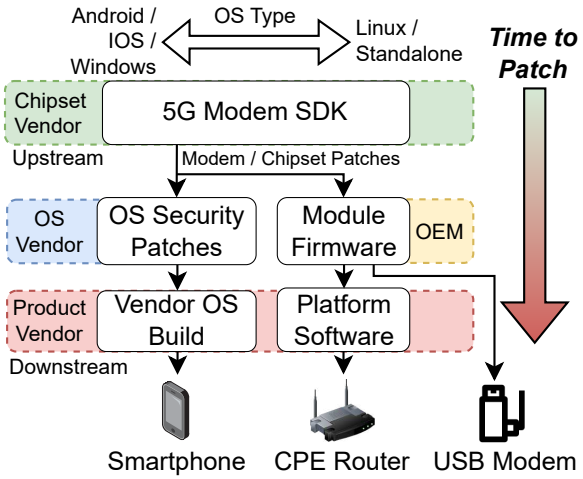


Figure 9: 5G UE Software Supply Ecosystem

Figure 9 depicts the complexity of the software supply chain of a 5G modem. In summary, finding issues in the implementation of the 5G modem vendor heavily impacts product vendors downstream. This is because the software dependency of product vendors on the *Modem / Chipset Vendor* adds complexity and hence delays to the process of producing and distributing patches to the end-user. For example, each iteration that the upstream 5G modem software goes through, carrier recertification must be performed by the chipset vendor so that the updated modem firmware can be integrated into OS security patches by the smartphone OS vendor on a fixed release schedule (i.e., Google for Android and Apple for IOS). Next, such security patches ought to be manually built into the smartphone OS image by the *product vendor*. Therefore, it can often take six or more months for 5G security patches to finally reach the end-user via an OTA update (final downstream node in Figure 9).

The chain of software dependencies are similarly applied to CPE routers or USB Modems. However, such type of products have a shorter time to distribute patches to the end-user since adhering to the release schedule of OS vendors is not required. Instead, the module vendor (i.e., OEM of Figure 9) directly builds modem patches into the module firmware and distribute it downstream via private channels. Therefore, the CPE product vendor can directly apply the patches from upstream to its platform software, which usually includes a customized Linux OS.

5 Exploit Scripts and Fuzzing Tool

The 5GHOUL proof-of-concept (PoC) and fuzzing tool is available open-source at our github repository:

https://github.com/asset-group/5ghoul-5g-nr-attacks

5GHOUL PoC is built on top of an over-the-air fuzzing framework that provides 5G NR protocol control to create test cases targeting 5G-capable Android smartphones or Qualcomm USB based modems. Nevertheless, using 5GHOUL involves a software defined radio such as USRP B210 to bring up the physical rogue base station. Once the 5GHOUL PoC is started, it automates the configuration process of the smartphone or modem to connect to the rogue base station for evaluating the attacks. Such PoC can also be used as a standalone over-the-air 5G SA fuzzer.

Our existing setup is quite portable when paired with an inexpensive mini PC (5GHOUL Mini PC in Figure 10). Currently, 5GHOUL PoC contains exploit scripts to launch attacks exploiting ***V3-V14*** (see Table 2). Scripts for other attacks exploiting ***V1-2*** and ***NAS Flooding Downgrade*** (see Figure 7) will be made available in the upcoming weeks.

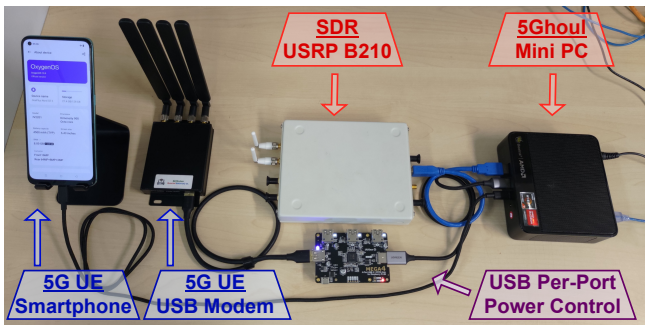


Figure 10: Hardware Setup for 5GHOUL PoC testing and fuzzer evaluation.

6 Reflection

In this section, we outline some of the key challenges we faced over the past few years, especially along the line of automated security testing of COTS 5G devices. We also discuss our personal experiences during the disclosure process that sheds light on the downstream patch propagation complexity and timeline.

Identification of Crashes: As our focus is on testing COTS devices, we faced significant challenges in precisely identifying device crashes. In particular, due to the nature of our adversarial testing (e.g., communicating malformed packets), the targeted 5G UEs may face arbitrary connection timeouts. Such connection-timeouts are normal and should not be confused with crashes (i.e., denial of service). Our initial research involved going through many message logs and trying to identify messages that may reliably indicate a crash. Indeed, at the early stage of our research, we had several message logs suspected to have crashes and only to be ruled out later as normal connection timeouts.

In particular, 5G USB modems contain a combination of proprietary and common AT commands for their configuration, which respond differently depending on the vendor [7]. To solve the crash detection problem in a generic fashion for 5G USB modems, we hand over control of the UE modem to Freedesktop ModemManager process (MM) [4]. MM process communicates with the UE mostly via the standard Qualcomm MSM Interface (QMI) and falls back to AT when necessary. In particular, whenever the modem is configured and ready to be used, MM informs our test process. Moreover, MM alerts modem crashes by informing QMI interface hangs or USB detachment caused by modem software reset. As per 5G smartphones, the device crash is indicated in the message log. Nonetheless, such a message varies with respect to modem vendor (e.g., Qualcomm, Mediatek etc.). Thus, the detection of 5G modem crashes in smartphones still involves minor vendor-specific engineering effort.

Debugging Support for Triage: Due to the closed nature of 5G UE stack, there is little support for us to triage crashes. This holds even after we confirm and report the vulnerability to vendor. While the message logs and Wireshark capture can show the modified packet(s) and crash indication, it still requires significant effort to locate the crash at the firmware code. During our disclosure period, one of the vendors (*Telit*) helped to provide better logging tool for debugging. Specifically, such a tool generated a comprehensive HTML report containing both core dump and backtrace summary, including several other information needed for Qualcomm's action. In particular, such a report could be generated for Qualcomm for vulnerabilities **V5** and **V6** (see Section 7), both of which led to firmware crashes (reachable assert). However, this was not the case for vulnerability **V7** that leads to a *hang*. Thus, no backtrace report could be produced to accelerate the triage effort for **V7**.

Time to Patch Downstream: In Figure 9, we have illustrated the typical 5G UE software supply chain. In line with the ecosystem shown in Figure 9, we share our experience with

the time involved in 5G modem patches becoming available to end users (e.g., smartphone users). We reported first vulnerabilities (i.e., **V5** and **V6** in Table 2) to Telit (USB modem vendor) on *17th April 2023*. Telit kindly provided a logging tool, which, in turn generated a comprehensive HTML report containing both core dump and backtrace summary. Such a summary clearly showed that the root cause of the crash/vulnerability resides in Qualcomm firmware code, seizing possibility of patches to be produced by Telit. This also confirmed that other products (e.g., smartphones) using the same Qualcomm 5G modems would also be affected in a similar fashion. After sending the comprehensive report to Qualcomm on *21st April 2023*, the crash triage was completed on *9th May 2023*. Eventually the patch for these vulnerabilities were provided by Qualcomm internally to their clients on *August 2023*. Finally, Android patches are expected to be available on *December 2023* for (smartphone) end users to repair our reported vulnerabilities.

In summary, it took approximately seven (7) months for the patch to be available to end users since it was reproduced by the vendor and approximately eight (8) months since its discovery. This shows the complexity of patch propagation to end users in 5G UE software supply chain. For vulnerabilities that did not have any accompanied crash report e.g., **V7** (see Figure 13), it took slightly longer time to triage (as expected due to the absence of any crash report): we reported on *11th May 2023* and the triage was complete on *13th June 2023*.

Software/Hardware Support for Testing at Scale: Our approach involves sending malformed packets from a malicious base station to COTS 5G UE. Such also needs to be done arbitrarily many times and automatically to uncover deeply rooted vulnerabilities in modem firmware. At an early stage of our research, we faced significant technical challenges to simply keep a stable over-the-air connection between open-source base station (i.e., OpenAirInterface [11]) and COTS 5G UE (e.g., commercial smartphones). This, in turn, resulted significant roadblocks in achieving practical results via over-the-air fuzzing. Moreover, due to the nature of fuzzing, we discovered that USB 5G modems frequently becomes unstable, which, in turn makes the fuzzing difficult to progress. To address this, we added a USB power switch to automatically restart USB modems. For Android smartphones, similar restart mechanism can be achieved via sending *reboot* command through Android Debug Bridge (ADB).

While 5G comes with a lot of potential for supporting complex applications, we foresee the need for deeper research in automatically and comprehensively hunting for implementation-level vulnerabilities in 5G software systems. Considering the multi-layer, complex implementation of 5G network stack and its closed nature, the research in 5G security testing has a long way ahead. This is evident from the discovery of 5GHOUL vulnerabilities in major chipset vendors, who, despite having all the resources for comprehensively testing their own 5G stack, are still susceptible to 5GHOUL vulnerabilities.

7 Vulnerabilities Description

In this section, we provide a detailed description of each vulnerability, attack vector, the affected 5G modem chipsets and/or the firmware version where applicable. We only include implementation vulnerabilities that were unknown at the time of discovery (i.e., **V5**-**V14** in Table 2). Some vulnerabilities were discovered when testing modems while others were detected by testing smartphones.

7.1 V5: Invalid MAC/RLC PDU (CVE-2023-33043)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the ***Qualcomm's X55/X60*** modem firmware by means of sending invalid downlink MAC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. After the *RRC Attach Procedure* is complete, the attacker (malicious gNB) sends a malformed *RLC Status PDU*, changing the MAC header (first byte) of the downlink transport block from 0x41 (DL-SCH) to ***0xB5 (MAC TCI States Activation/Deactivation)***. We highlight this byte in red in Figure 11. After the invalid frame is sent to the 5G UE, the Qualcomm X55 modem fails and reboots, indicating a firmware assert in the coredump log (QCAP). More specifically, the modem temporarily hangs and performs a reboot in ***5 to 10 seconds***. This delay is noticeable in products using such modems. For example, *Telit FT980-WW* Web-based application becomes accessible only after the platform reboots due to the following firmware coredump message:

> [ASSERT] nr5g_ml1_mdb.c:12636 Assert serving_cell_ptr->cell_data.nr5g.configured_bwp_bmask & (1< <bwp_idx) failed: gNB beam update for BWP not configured 0x1

Impact: *An attacker within X55/X60 modem-based UE radio range can block 3GPP Modem connectivity by continuously sending an invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

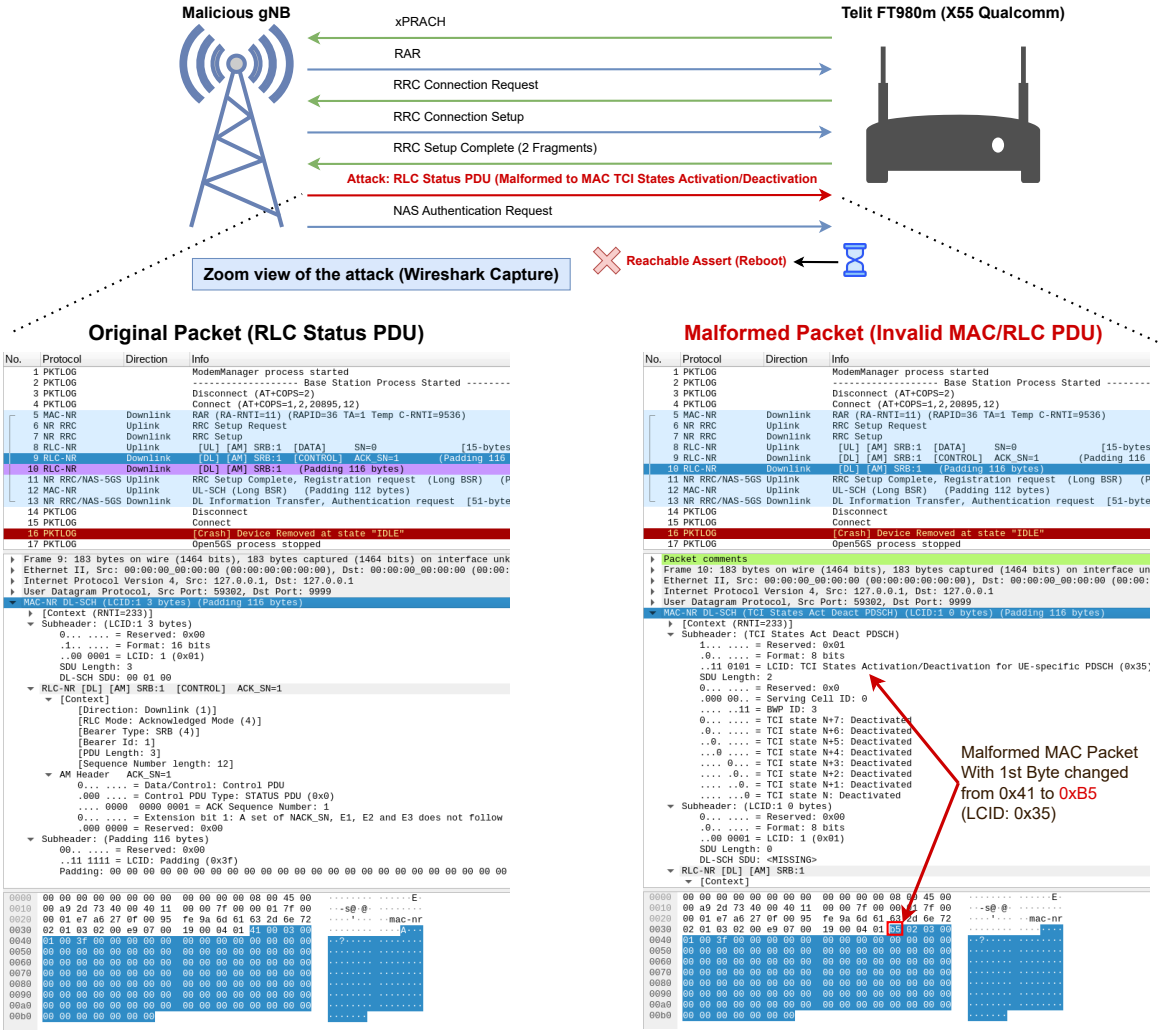


Figure 11: Invalid MAC/RLC PDU.

7.2 V6: NAS Unknown PDU (CVE-2023-33044)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the ***Qualcomm's X55/X60*** modem firmware by means of sending invalid downlink MAC frame to the target 5G UE (smartphone, CPE router) from a nearby malicious gNB. After the *RRC Attach Procedure* is complete and before NAS network registration is complete, the attacker (malicious gNB) sends an invalid NAS PDU, changing the third byte of the ***RRC dlInformationTransfer*** from *0xC0* (NAS Authentication Request) to ***0xF4 (NAS Unknown PDU)*** (See this byte highlighted in red in the Figure 12). After the invalid frame is sent to the 5G UE, the Qualcomm X55/X60 modem fails and reboots, indicating a firmware assert in the coredump log (QCAP). More specifically, the modem temporarily hangs and performs a reboot in ***5 to 10 seconds***. This delay is noticeable in applications. For example, *Telit FT980-WW* Web-based application becomes accessible only after the platform reboots due to the following firmware coredump message:

```
[ASSERT] ds_3gpp_tlb_ctrl.c:2088 Assertion msg_size == 1 failed
```

Impact: *An attacker within X55/X60 UE radio range can block 3GPP Modem connectivity by continuously sending an invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *early* in the NAS registration procedure, which happens before any NAS authentication procedure completes.

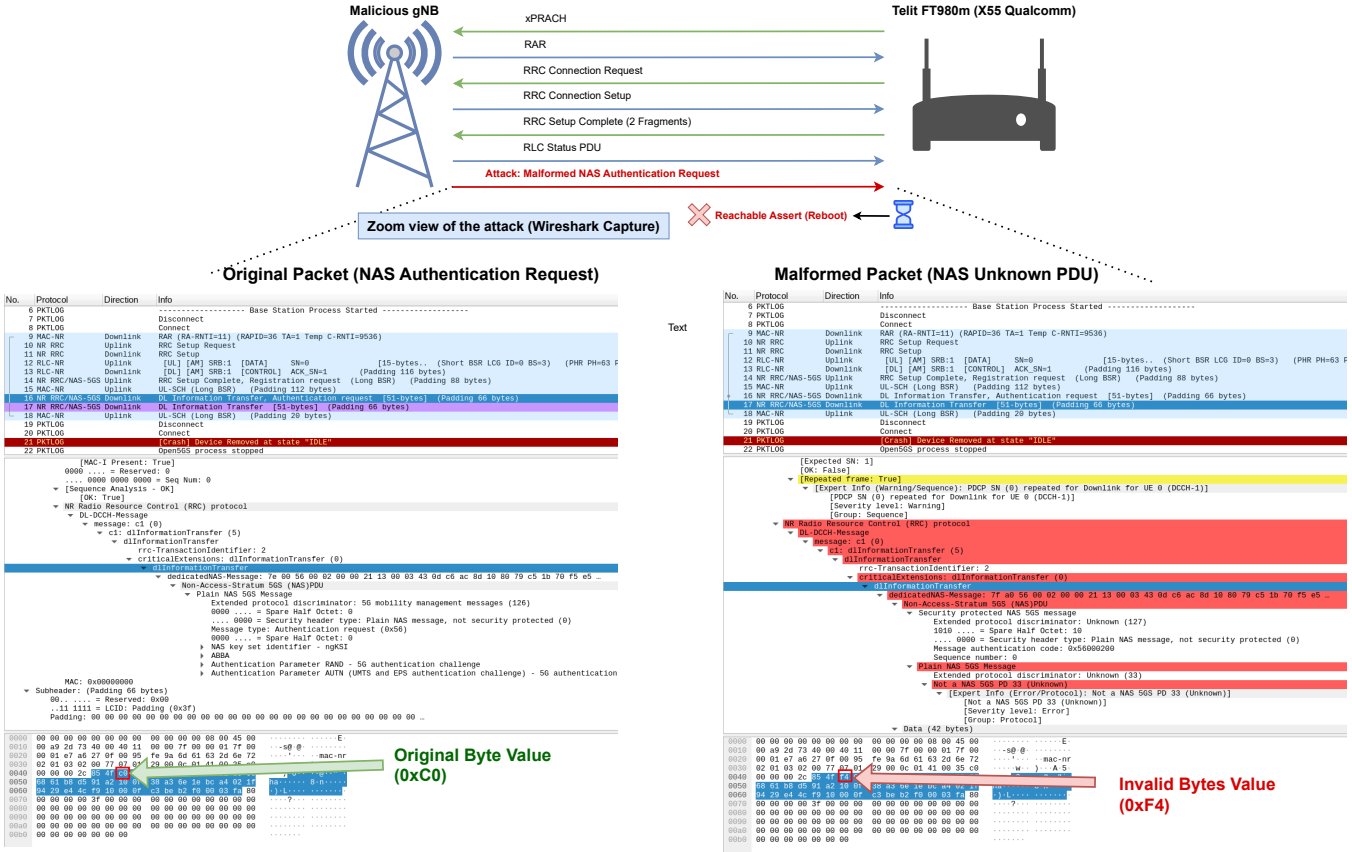


Figure 12: NAS Unknown PDU.

7.3 V7: Disabling 5G / Downgrade via Invalid RRC pdcch-Config (CVE-2023-33042)

An attacker within radio range can trigger a 5G connectivity downgrade or denial of service within the ***Qualcomm's X55/X60*** modem firmware by means of sending malformed RRC frame to the target 5G UE (smartphone, CPE router) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed NAS PDU, changing the *30th* byte of the ***RRC dlInformationTransfer*** payload from *0x04* to ***0x9C***. This modifies optional bits of the ***pdcch-Config*** such as ***tpc-PUCCH*** and ***Sequence-Of Length*** (See these bytes highlighted in red in the Figure 12). After the malformed message is sent to the 5G UE, connection to any 5G network fails. For example, USB Modems or CPE Routers are often not able to scan 5G networks via the serial command *AT+COPS=?* and hence connection to any gNB (including legitimate ones) fails. Moreover, attempting to connect to any listed 5G network does not succeed either. This behavior was verified to occur in all of our 5G-capable Qualcomm devices such as *Asus Rog Phone 5S*, *Quectel RM500Q-GL* etc (see Table 2). Once the attack is launched, the X55/X60 modem remains affected until the user reboots the phone. Simply switching airplane mode on and off does not restore 5G communication.

Impact: *An attacker within Qualcomm X55/X60 UE radio range can deny or downgrade 5G connectivity by sending a single malformed packet. The user needs to manually reboot the phone to recover 5G connectivity*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

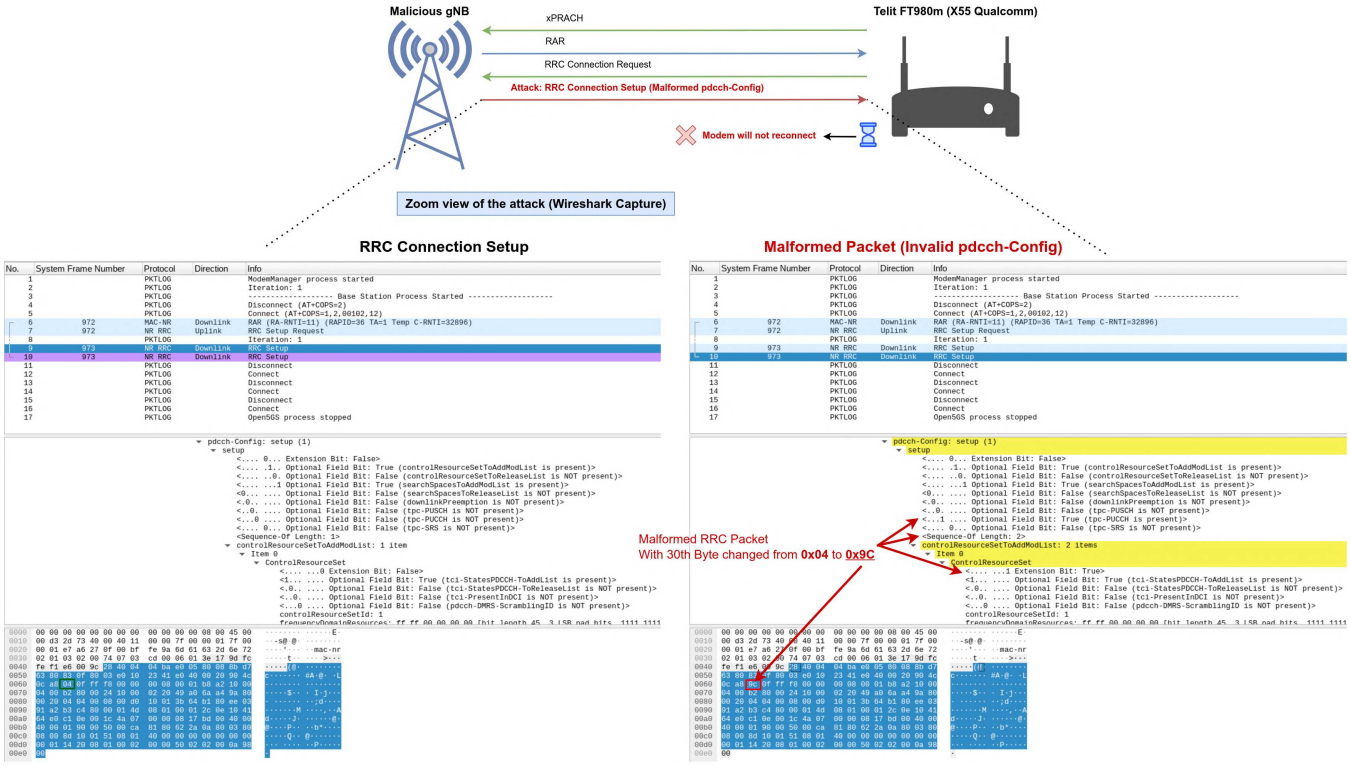


Figure 13: Disabling 5G / Downgrade via Invalid RRC pdcch-Config.

7.4 V8: Invalid RRC Setup spCellConfig (CVE-2023-32842)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the *35th* byte of the downlink transport block from *0x4C* to *0xD6* and *66th* byte from *0x9A* to *0x71* (See this bytes highlighted in red in the Figure 14). This corresponds to changing some fields within the RRC payload such *spCellConfigDedicated* optional bits: ***defaultDownlinkBWP-Id=1***, ***pdcch-ServingCellConfig=1***, ***pdsch-ServingCellConfig=0***, ***sCellDeactivationTimer=1***. After such an invalid packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[ASSERT] file:mcu/l1/mml1/mml1_endc/src/mml1_endc_db_hdlr.c line:524 p1:0x91920c70
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

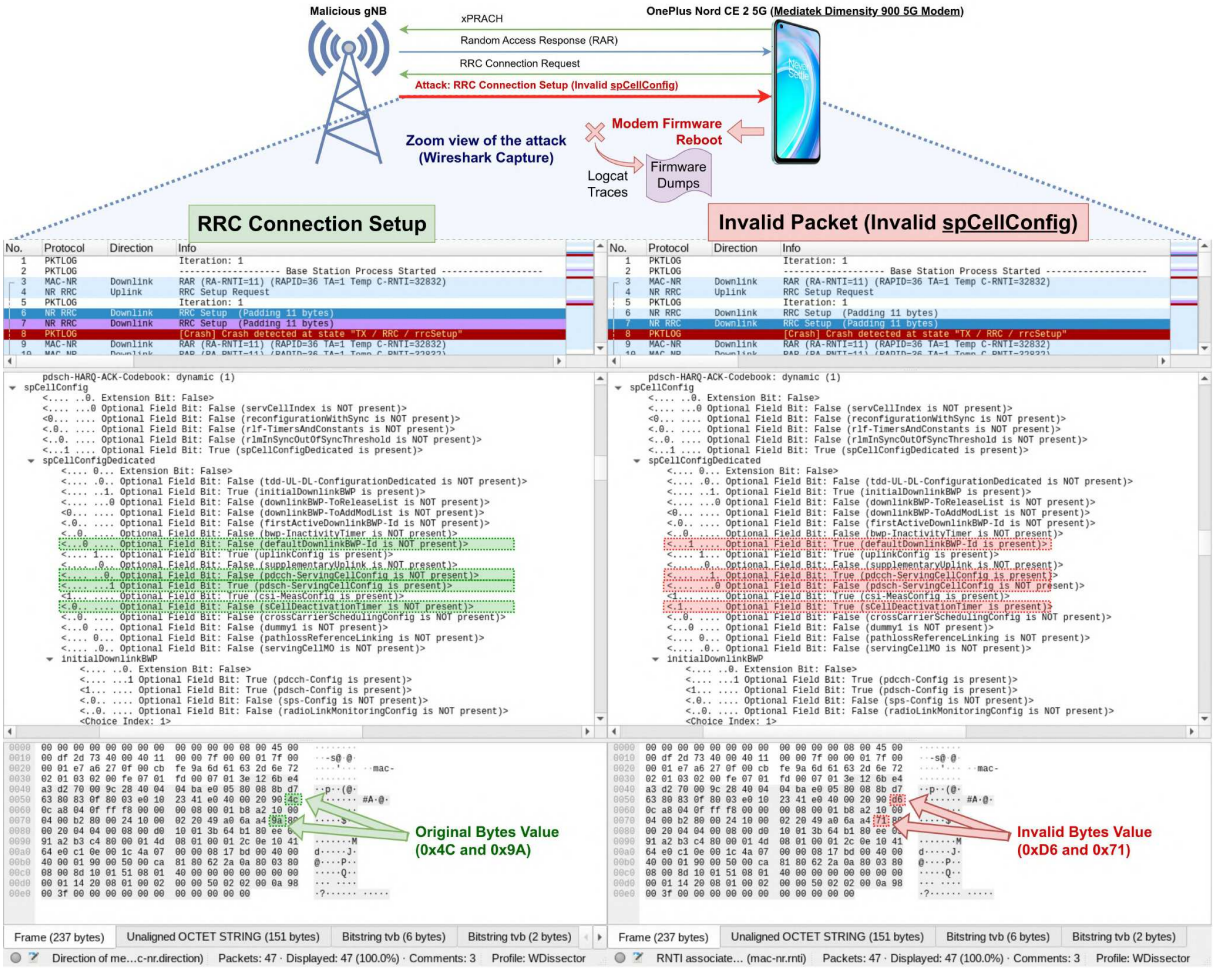


Figure 14: Invalid RRC Setup spCellConfig.

7.5 V9: Invalid RRC pucch CSIReportConfig (CVE-2023-32844)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the ***192nd byte*** of the downlink transport block from *0x0A* to ***0xD0*** (See this byte highlighted in red in Figure 15). This corresponds to changing some fields within the RRC payload (*searchSpacesToAddModList* element) such as ***pucch-Resource*** from *0x02* to ***0x52*** and ***reportQuantity*** from *0x05 (cri-RSRP)* to ***0x01 (cri-RI-PMI-CQI)***. After such invalid packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[ASSERT] file:mcu/l1/nl1/internal/md97/src/rx/nr_rx_dspcmd_ext_csif_csi.c line:2657 p1:0x00000000
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

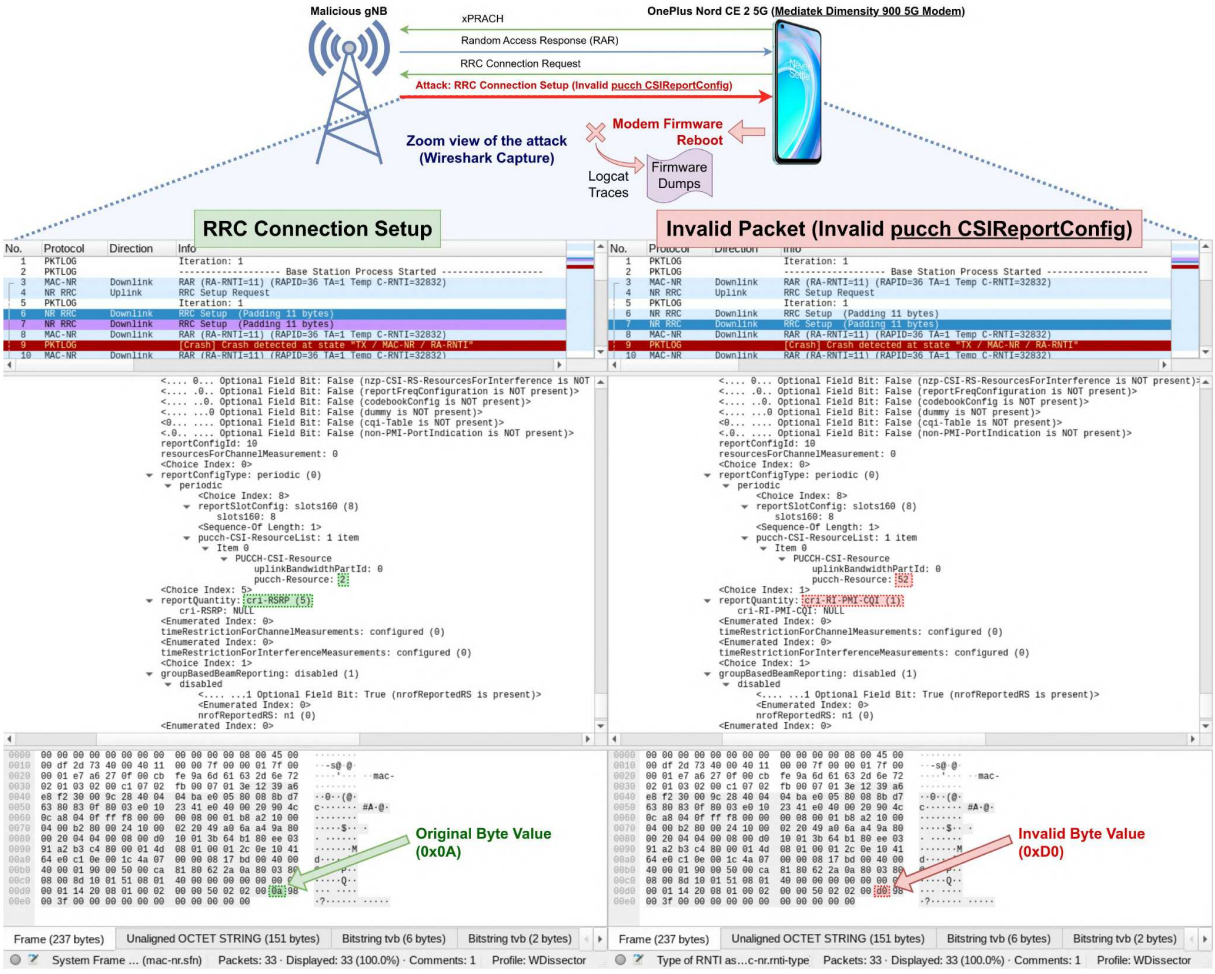


Figure 15: Invalid RRC pucch CSIReportConfig.

7.6 V10: Invalid RLC Data Sequence (CVE-2023-20702)

An attacker within radio range can trigger a denial of service (***null pointer dereference***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RLC payload to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. After the *RRC attach procedure*, the attacker (malicious gNB) sends a malformed *RLC Status PDU*, changing the ***4th*** byte of the downlink transport block from 0x00 to ***0x84*** (See this byte highlighted in red in Figure 16). This corresponds to changing some fields within the RLC header such as ***Data/Control field*** to ***0x01 (Data PDU)*** and ***RLC Sequence Number*** to ***1025***. The original downlink packet is an *RLC Control PDU (Status ACK)*. However, after writing the invalid byte ***0x84*** to the RLC Header, such PDU is changed to a *Data PDU* with a wrong (out-of-order) RLC Data sequence number. After the invalid RLC payload is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[Fatal error(MPU_NOT_ALLOW)] err_code1:0x0000001D err_code2:0x910D66F6 err_code3:0x910D66E2
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *after* the RRC attach procedure, which happens before any NAS authentication procedure.

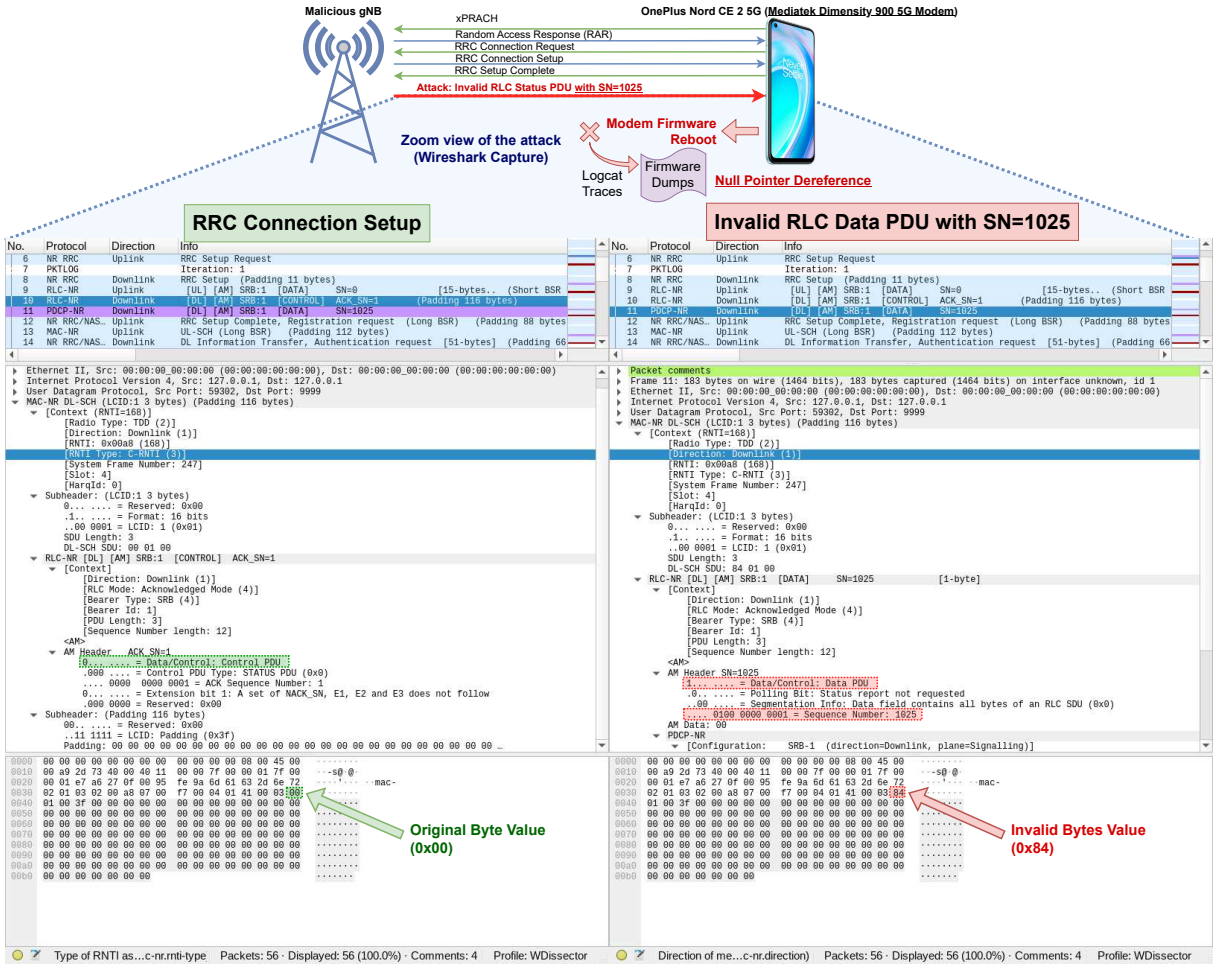


Figure 16: Invalid RLC Data Sequence.

7.7 V11: Truncated RRC physicalCellGroupConfig (CVE-2023-32846)

An attacker within radio range can trigger a denial of service (***null pointer dereference***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the ***14th*** and ***15th*** bytes of the downlink transport block from *0xBA 0xE0* to ***0x13 0x46*** (See these bytes highlighted in red in Figure 17). This corresponds to changing some fields within the ***physicalCellGroupConfig*** such as ***Extension bit*** to ***0x01 (True)*** and ***pdsch-HARQ-ACK-Codebook*** to ***0x00 (semiStatic)***. Consequently, this causes the RRC message to be truncated, thus causing encoding errors as highlighted in Wireshark (yellow messages in Figure 17). After such malformed packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware memory access error in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[Fatal error(MPU_NOT_ALLOW)] err_code1:0x0000001D err_code2:0x90F5D83A err_code3:0x90F5D836
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such malformed packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

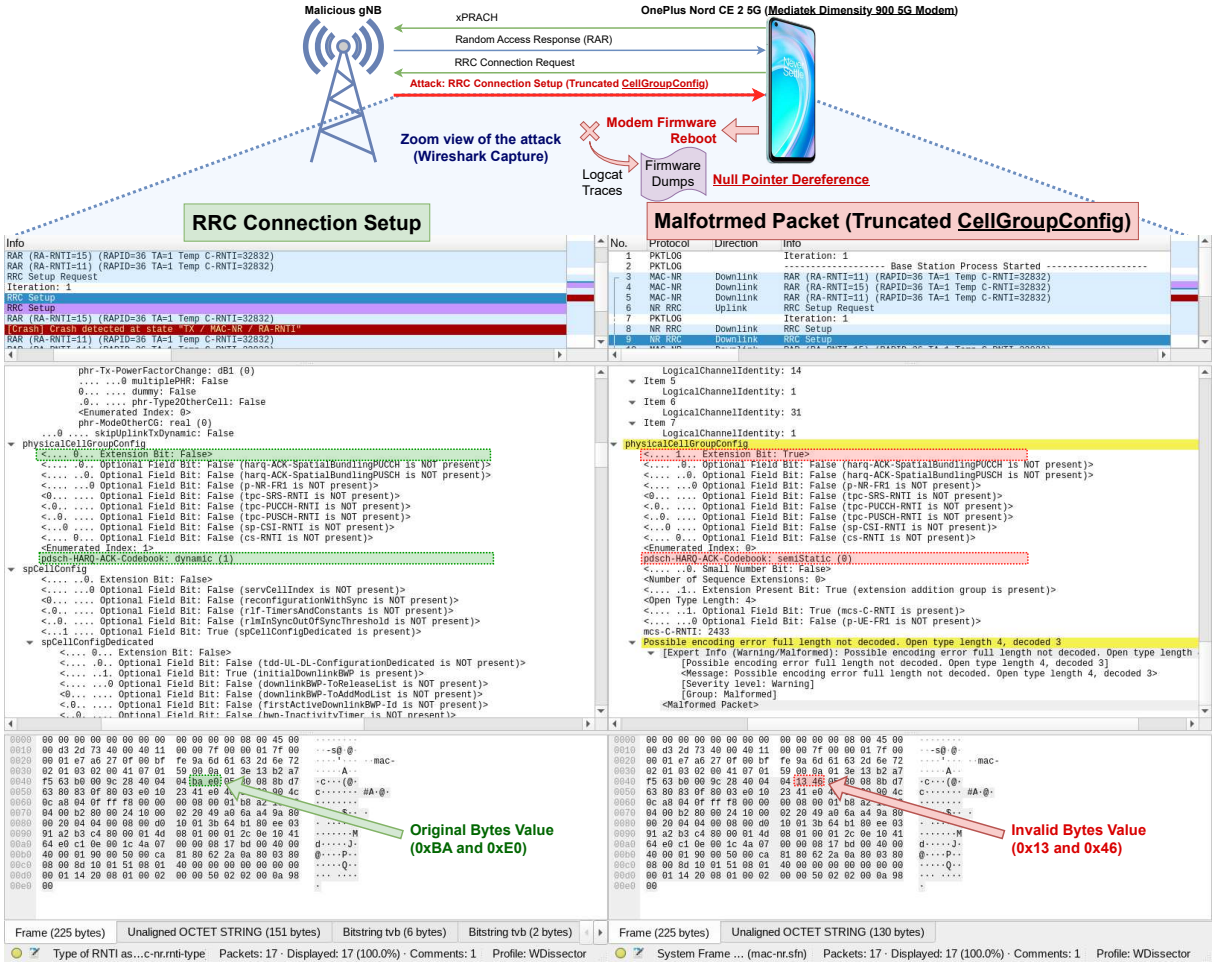


Figure 17: Truncated RRC physicalCellGroupConfig.

7.8 V12: Invalid RRC searchSpacesToAddModList (CVE-2023-32841)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the ***52nd byte*** of the downlink transport block from *0x00* to ***0x5B*** (See this byte highlighted in red in Figure 18). This corresponds to changing some fields within the RRC payload (*searchSpacesToAddModList element*) such as ***monitoringSymbolsWithinSlots*** from *0x80 0x00* to ***0x82 0xd8***. After such invalid packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[ASSERT] file:dsp3/coresonic/msonic/modem/slm/nr/nr_post_proc/src/nr_slm_rpt_hndlr.c line:9302
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

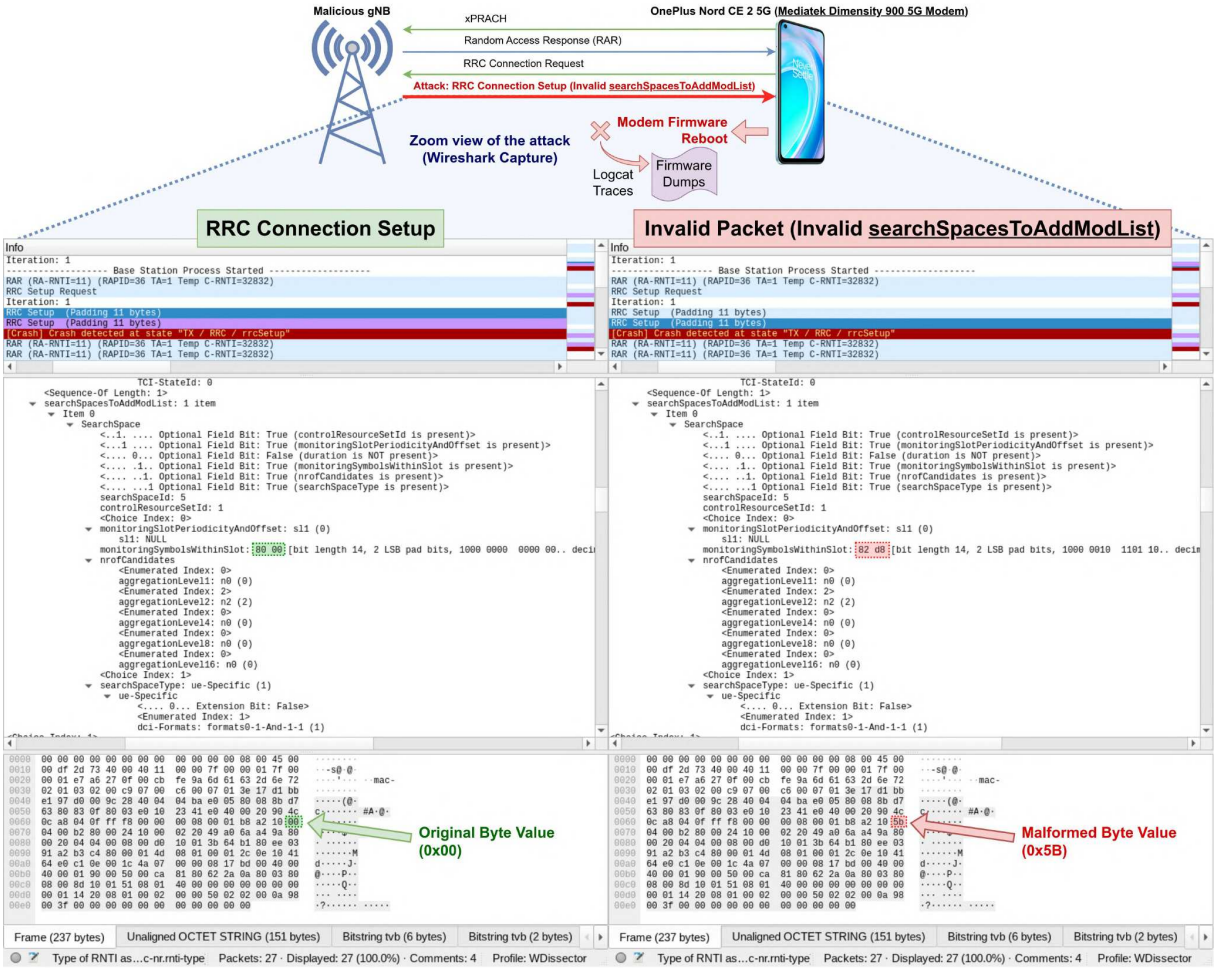


Figure 18: Invalid RRC searchSpacesToAddModList.

7.9 V13: Invalid RRC Uplink Config Element (CVE-2023-32843)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the ***92nd byte*** of the downlink transport block from *0x49* to ***0x67*** (See this byte highlighted in red in the Figure 19). This corresponds to changing many fields within the RRC payload such as ***qcl-Type*** from 0x02 to ***0x03***, ***resourceAllocation*** from 0x01 to ***0x00***, ***rbg-size*** from 0x00 to ***0x01***, ***prb-BundlingType*** from 0x00 to ***0x01*** and other ***uplinkConfig optional bits***. After such invalid packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[ASSERT] file:mcu/l1/nl1/internal/md97/src/rfd/nr_rfd_configdatabase.c line:4380 p1:0x00000001
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

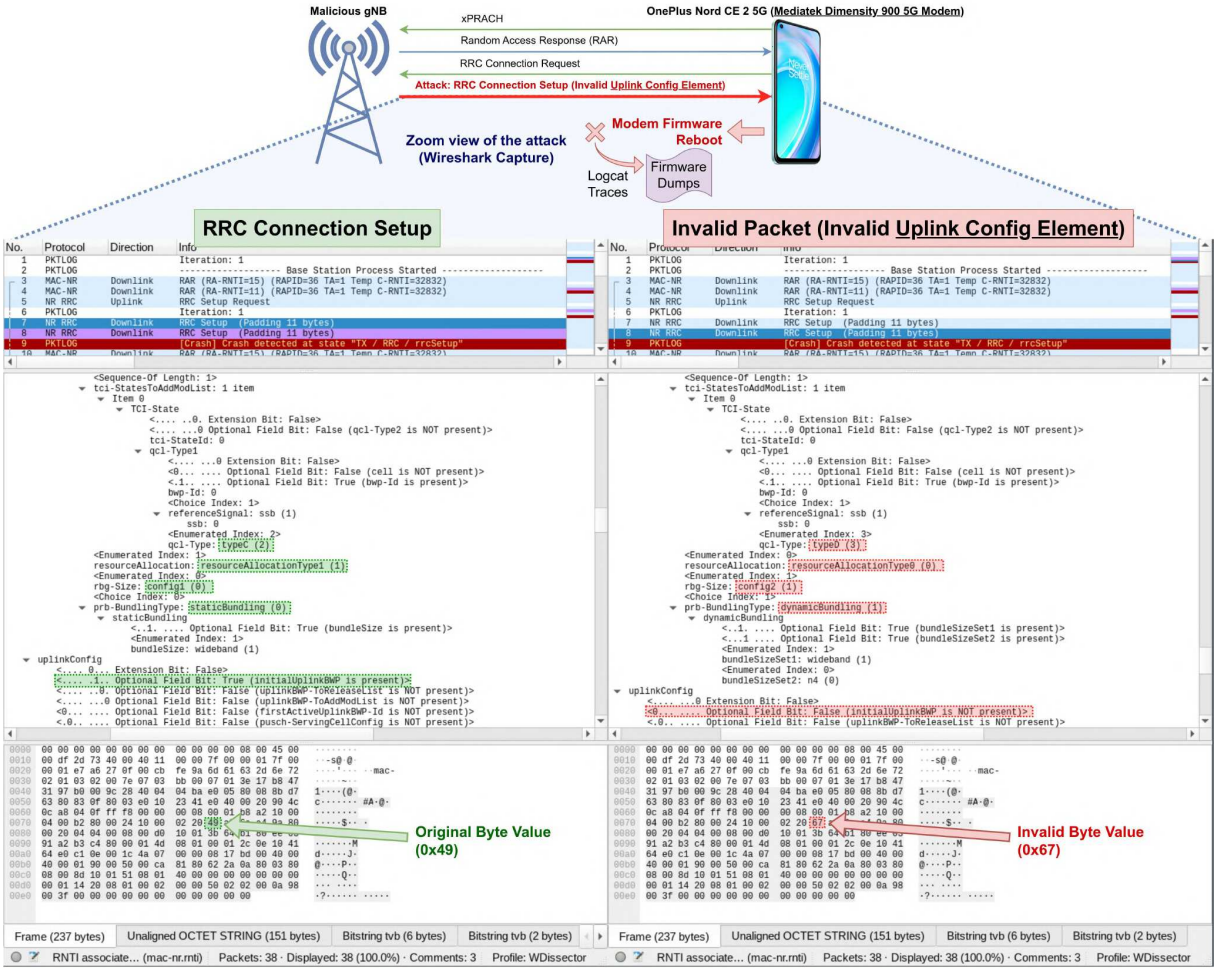


Figure 19: Invalid RRC Uplink Config Element.

7.10 V14: Null RRC Uplink Config Element (CVE-2023-32845)

An attacker within radio range can trigger a denial of service (***reachable assert***) within the Mediatek ***Dimensity 900 / 1200*** 5G modem firmware by means of sending invalid downlink RRC frame to the target 5G UE (e.g., smartphone) from a nearby malicious gNB. During the *RRC Attach Procedure*, the attacker (malicious gNB) sends a malformed *RRC Connection Setup*, changing the ***50th byte*** of the downlink transport block from *0xA2* to ***0xA1*** (See this byte highlighted in red in Figure 20). This corresponds to changing some fields within the RRC payload such as ***controlResourceSetId*** to ***0x00***, ***pdsch-Config*** to ***0x00 (release)***, amongst other fields as shown in Figure 20. Particularly, when *pdsch-Config* is set to release, the *Uplink Config Element* is decoded as *Null*. Consequently, when such an invalid packet is sent to the 5G UE, the *Dimensity 900* 5G modem fails and reboots, indicating a firmware crash/assert in the Android logs (logcat). More specifically, the modem temporarily hangs and performs a reboot in ***2 to 7 seconds***, being noticeable by logcat outputting the following firmware assert message:

```
[ASSERT] file:dsp3/coresonic/msonic/modem/brp/nr/nr_brp/src/nr_brp_top_irq.c line:927
```

Impact: *The attacker can block 3GPP Modem connectivity by continuously sending such invalid packet*. Furthermore, this vulnerability is fairly easy to trigger since no information about the target's SIM card information is required to launch the attack. This is because the invalid downlink frame is sent *during* the RRC attach procedure, which happens before any NAS authentication procedure.

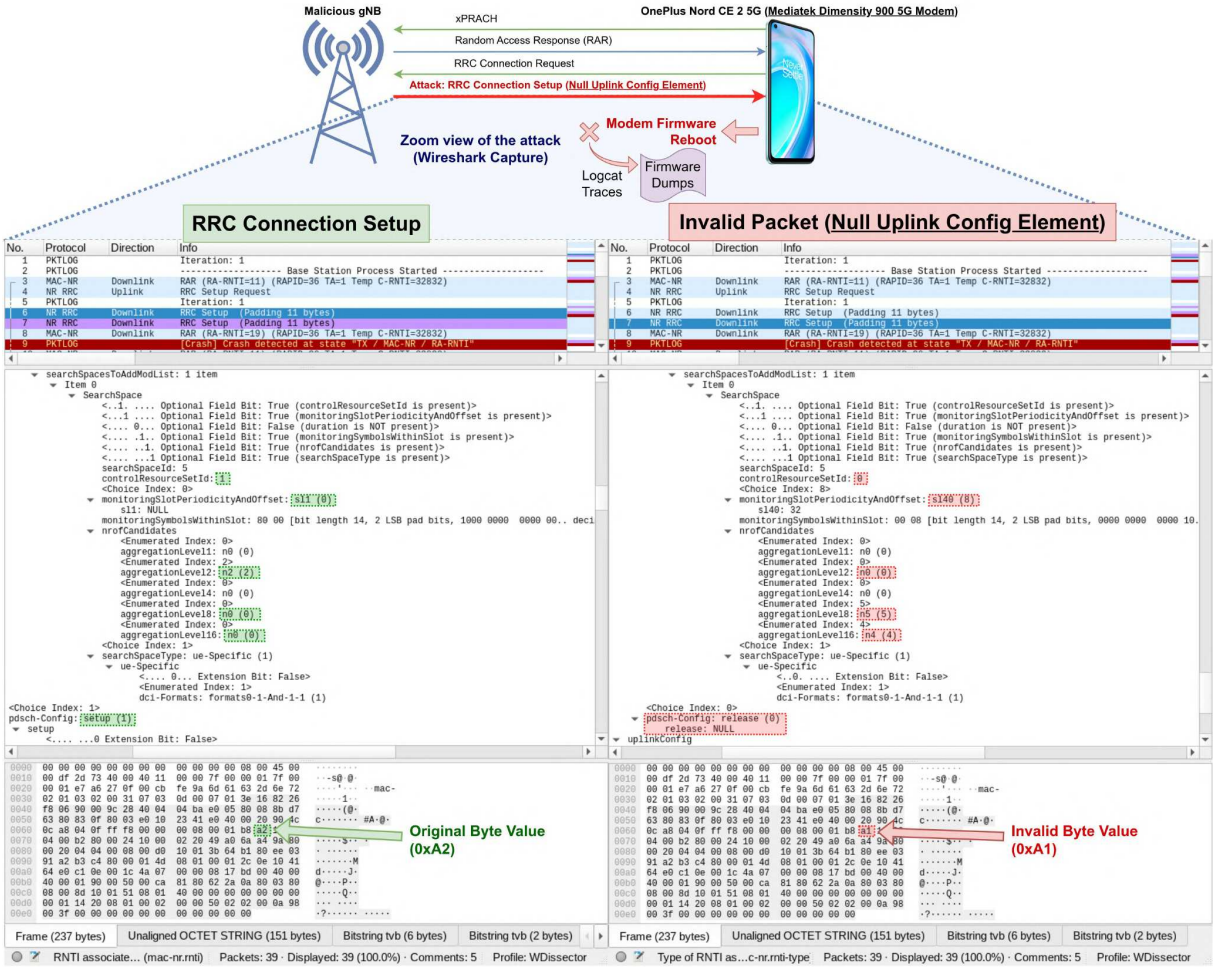


Figure 20: Null RRC Uplink Config Element.

Acknowledgments

This research was partially supported by Future Communications Program hosted at Singapore University of Technology and Design and iTrust National Satellite of Excellence (NSOE) on Design Science and Technology for Secure Critical Infrastructure (IoT Sector). We thank Sakshi Udeshi for coining the term 5GHOUL . We thank Rushati Chakraborty for designing the 5GHOUL logo (Ghost icon by Flaticon). We greatly appreciate the help from Mao Ngo in the Future Communications Lab, he kindly lent us the 5G smartphones from his lab where we found the first vulnerability. Finally, we are highly thankful for the support of Qualcomm and MediaTek during the coordinated disclosure process and generously awarding us for our findings.

References

[1] Open5GS implementation for 5G Core and EPC. https://github.com/open5gs/open5gs, 2023. [Online; accessed 04-Dec-2023].

[2] 3GPP. 3GPP Specifications and Technologies. https://www.3gpp.org/specifications.html/, 2023. [Online; accessed 06-Dec-2023].

[3] Evangelos Bitsikas, Syed Khandker, Ahmad Salous, Aanjhan Ranganathan, Roger Piqueras Jover, and Christina Popper. UE security reloaded: Developing a 5g standalone user-side security testing framework. In *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '23, page 121–132, New York, NY, USA, 2023. Association for Computing Machinery.

[4] freedesktop.org. ModemManager. https://www.freedesktop.org/wiki/Software/ModemManager/, 2023. [Online; accessed 01-Dec-2023].

[5] Matheus E. Garbelini, Zewen Shang, Sudipta Chattopadhyay, Sumei Sun, and Ernest Kurniawan. Towards Automated Fuzzing of 4G/5G Protocol Implementations Over the Air. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, pages 86–92, Rio de Janeiro, Brazil, December 2022. IEEE.

[6] Syed Rafiul Hussain, Mitziu Echeverria, Imtiaz Karim, Omar Chowdhury, and Elisa Bertino. 5GReasoner: A Property-Directed Security and Privacy Analysis Framework for 5G Cellular Network Protocol. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, pages 669–684, New York, NY, USA, November 2019. Association for Computing Machinery.

[7] Imtiaz Karim, Fabrizio Cicala, Syed Rafiul Hussain, Omar Chowdhury, and Elisa Bertino. Opening Pandora's Box through ATFuzzer: Dynamic Analysis of AT Interface for Android Smartphones. In *Proceedings of the 35th Annual Computer Security Applications Conference*, ACSAC '19, pages 529–543, New York, NY, USA, December 2019. Association for Computing Machinery.

[8] Eunsoo Kim, Dongkwan Kim, CheolJun Park, Insu Yun, and Yongdae Kim. BaseSpec: Comparative Analysis of Baseband Software and Cellular Specifications for L3 Protocols. In *Proceedings 2021 Network and Distributed System Security Symposium*, Virtual, 2021. Internet Society.

[9] Martin Kotuliak, Simon Erni, Patrick Leu, Marc Roschlin, and Srdjan Čapkun. {LTrack}: Stealthy Tracking of Mobile Phones in {LTE}. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1291–1306, 2022.

[10] MediaTek. December 2023 Product Security Bulletin. https://corp.mediatek.com/product-security-bulletin/December-2023/, 2023. [Online; accessed 04-Dec-2023].

[11] Navid Nikaein, Mahesh K. Marina, Saravana Manickam, Alex Dawson, Raymond Knopp, and Christian Bonnet. OpenAirInterface: A Flexible Platform for 5G Research. *SIGCOMM Comput. Commun. Rev.*, 44(5):33–38, October 2014.

[12] OpenWrt. OpenWrt 22.03.4 - Service Release - 10 April 2023. httpshttps://openwrt.org/releases/22.03/notes-22.03.4/, 2023. [Online; accessed 04-Dec-2023].

[13] Qualcomm. December 2023 Security Bulletin. https://docs.qualcomm.com/product/publicresources/securitybulletin/december-2023-bulletin.html/, 2023. [Online; accessed 04-Dec-2023].

[14] Telit Cinterion. FT980-WW Advanced LTE/5G Wireless Router Demo Kit. https://www.telit.com/support-tools/development-evaluation-kits/ft980/, 2023. [Online; accessed 04-Dec-2023].

[15] Vodafone. Vodafone Unveils Prototype 5G Network Built on a Raspberry Pi Computer. https://www.vodafone.com/news/technology/vodafone-unveils-prototype-5g-network-built-raspberry-pi-computer/, 2023. [Online; accessed 07-Dec-2023].