CSAI: Open-Source Cellular Radio Access Network Security Analysis Instrument

Thomas Byrd and Vuk Marojevic
Dept. of Electrical and Computer Engineering
Mississippi State University
Mississippi State, MS
{tkb140, vuk.marojevic}@msstate.edu

Roger Piqueras Jover*
Bloomberg LP
New York, NY
rpiquerasjov@bloomberg.net

***Abstract*—This paper presents our methodology and software toolbox that allows analyzing the radio access network security of laboratory and commercial 4G and future 5G cellular networks. We leverage a free open-source software suite that implements the LTE UE and eNB enabling real-time signaling using software radio peripherals. We modify the UE software processing stack to act as an LTE packet collection and examination tool. This is possible because of the openness of the 3GPP specifications. Hence, we are able to receive and decode LTE downlink messages for the purpose of analyzing potential security problems of the standard. This paper shows how to rapidly prototype LTE tools and build a software-defined radio access network (RAN) analysis instrument for research and education. Using the Cellular Security Analysis Instrument (CSAI), a researcher can analyze broadcast and paging messages of cellular networks. CSAI is also able to test networks to aid in the identification of vulnerabilities and verify functionality post-remediation. Additionally, we found that it can crash a software eNB which motivates equivalent analyses of commercial network equipment and its robustness against denial of service attacks.**

***Index Terms*—open-source LTE, SDR, paging, radio access network signaling, analysis, wireless security**

I. Introduction

The long-term evolution (LTE) is a cellular communications standard developed by the 3rd Generation Partnership Project (3GPP). LTE was finalized in 3GPP Release 8 in December 2008, and LTE-Advanced followed in 3GPP Release 10. Only recently has there been significant enough open source software development efforts for producing stable implementations of the LTE and LTE-A specifications to allow for rapid prototyping and testing of 4G networks by the broader research community.

Next generation 5G networks promise a huge leap from 4G. The reality however is that the initial 5G releases leverage LTE networks in many regards: New Radio (NR) initially implements a similar radio access network (RAN) and hooks to LTE's evolved packet core (EPC). 5G generally allows more flexible waveform and protocol configurations, transmission in sub 6 GHz and millimeter wave bands, and higher bandwidths than LTE. The waveform will initially be orthogonal frequency division multiplexing (OFDM) and the 5G signaling frame will carry user data and control information.

There is a huge need for research and development tools that enable cellular signaling analysis for a multitude of purposes. It can help understand the limitations of current implementations and guide the evolution of the standard. They can also be effectively used for education and training. Security is another important aspect where RAN signaling analysis is needed. It has been shown that the LTE control signaling suffers from targeted interference that an adversary can exploit, easily and cheaply [1]. We therefore propose a flexible signal analysis tool for analyzing commercial and experimental cellular communication systems, assisting in the detection of potential vulnerabilities, and evaluating corrective measures which will pave the path to secure wireless networks.

We leverage open-source software implementations of LTE and introduce the free and open-source *Cellular Security Analysis Instrument (CSAI)* in this paper. CSAI is lightweight and can process data in real time. It interfaces with common software radio hardware, such as Ettus Research USRPs, and can capture LTE control messages and be extended to capture 5G NR signals. It can emulate an eNodeB (eNB) or user equipment (UE) and implement specific processes to test the behavior of the UE or eNB. It also allows testing larger RANs which involve multiple UEs or multiple eNBs. For example, in commercial networks that have dozens of UEs, or more, that rotate between serving cells, this tool will be able to monitor paging traffic in a particular cell and identify new UEs as they are paged for analysis.

It is very important to be able to analyze protocol edge cases and understand their implications in terms of RAN security. Not only can it be used for examining the standard specifications of a modern cellular standard, but this tool can also test vendor specific implementations. Additionally, it is a benchmarking tool for stress testing 4G and 5G networks and can be adapted to fit different use cases. For instance, if a vendor needs an automated tool to determine the limits of their Radio Resource Control (RRC) buffers, this instrument will be able to facilitate that.

The remainder of this paper is organized as follows. Section II briefly outlines other work in the area of capturing LTE messages and performing LTE security analyses. Section III describes the important LTE signaling procedures over the

*Author did not contribute to source code. Code published to accompany this paper was written by Mr. Thomas Byrd and Dr. Vuk Marojevic.

978-1-7281-5207-3/20/$31.00 ©2020 IEEE

RAN. This allows for better comprehension of Section IV, which introduces CSAI, our software instrument for analyzing broadcast and paging messages, among others. Section V discusses experiments and data collected from commercial networks. Section VI focuses on the security implications of our initial results, and Section VII concludes the paper.

II. Related Work

Security research of cellular communications standards has a long history and helped evolve systems to the current 4G and emerging 5G networks [2]. The insecure 2G systems are still used today and whenever 4G or 3G coverage is not available, handsets look for 2G networks. 4G systems introduce network and user authentication, where a user can authenticate the network it connects to. However, certain 4G security vulnerabilities were identified that 5G networks intend to fix.

With the emergence of software radios, increasing processing power of general-purpose computers, and software implementations of cellular standards, experimental LTE security research took off [3]. Researchers dissected the entire LTE signaling frame looking for vulnerabilities of the system when specific subsystems are interfered with. Two types of attacks were examined, control channel jamming and spoofing, and mitigation mechanisms were proposed in [4] and [5]. Other research groups tested LTE's higher layer signaling protocols and published their findings in open literature [6].

While there exist many commercial tools that perform LTE traffic capture and decoding, to our knowledge, there is no open source software that will accomplish this. Papers that have been published regarding LTE security require the use of commercial LTE capture tools or the development of custom tools as observed in [7] and [8]. The relevance of this subject is apparent from the availability of professional test instruments, offered by various hardware and software companies. But their high cost limits their widespread use in research and education. Our goal is to provide a framework for making cellular RAN signaling analysis accessible to all, enable wireless security research, increase the transparency and visibility of RAN operations, and allow easy adoption by industry and standardization bodies.

III. Background

This section provides the necessary background on how LTE UEs register to the network and get notified by the network of incoming messages or calls. When a UE powers on, it first needs to receive and decode the Primary and Secondary Synchronization Signals (PSS/SSS) [9]. Together, these two signals allow the UE to synchronize on a slot and frame level basis, respectively, as well as correct for frequency offsets between the eNB and UE oscillators. Now that the UE is synchronized with the eNB, it needs to know more information about the serving network before it can initiate an attach request. Therefore, it decodes the Master and System Information Blocks (MIB/SIBs). These blocks are transmitted in the clear by the eNB on a regular basis to ensure that UEs have the necessary information needed to attach. This is the initial cell search that each UE performs when turned on or when returning out of coverage and is part of the information that our tool can capture and analyze.

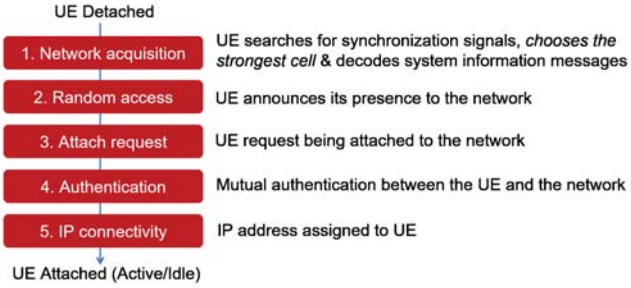

UE Detached
1. Network acquisition
UE searches for synchronization signals, *chooses the strongest cell* & decodes system information messages
2. Random access
UE announces its presence to the network
3. Attach request
UE request being attached to the network
4. Authentication
Mutual authentication between the UE and the network
5. IP connectivity
IP address assigned to UE
UE Attached (Active/Idle)

Fig. 1. The LTE UE attachment process.

Once a UE knows the network configuration details that are provided in the MIB/SIBs, it can then use its Random Access Radio Network Temporary Identity (RA-RNTI) to initiate a RRC connection with the eNB. After a UE has established an RRC connection, the UE will communicate to the EPC through the eNB over the Non-Access Stratum (NAS) protocol layer. Fig. 1 illustrates this attachment process.

In order to identify themselves, UEs utilize the International Mobile Subscriber Identity (IMSI). This secret identifier can be leveraged in a number of privacy-invading attacks [10] and, as such, should always be kept private. However, the UE will authenticate with the EPC's Home Subscriber Server (HSS) transmitting its IMSI in the clear if the UE has no history with the network.

Once all of the NAS and RRC connections are established, the UE will enter an Idle state and deactivate the radio link between itself and the eNB. If the EPC needs to deliver a message to an idle UE, it is the job of the eNB to wake-up the idle device and re-establish a physical connection [11]. This is done by sending out a paging message to UEs in the operational area of the eNB. Paging messages use a specific Paging RNTI (P-RNTI) [12] to indicate the broadcast nature of paging and UEs are required to respond if their IMSI or System Architecture Evolution (SAE) Temporary Mobile Subscriber Identity (S-TMSI) is being paged. The S-TMSI is a combination of the Mobility Management Entity (MME) Code and the Mobile TMSI (m-TMSI); herein both are simply referred to as the TMSI. Our tool is able to capture and decode the SIBs and paging messages for specified eNBs, which enables performing RAN security analyses.

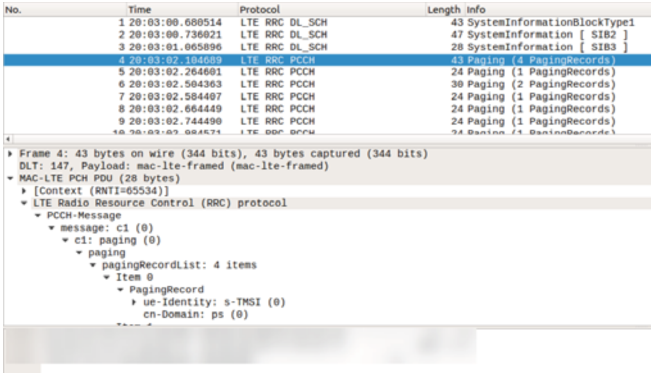
IV. CSAI

There are numerous ways to analyze the LTE RAN. These range from expensive commercial cellular test equipment to open source GNU Radio implementations that provide insight into cellular networks. We chose srsLTE [13] for its simplicity and applicability toward analyzing LTE signals and messages.

srsLTE implements 3GPP Release 8 with certain components of Release 9 integrated into its software. It is licensed under the GNU Affero General Public License for free use for non-commercial purposes, such as research and education. The srsLTE software suite is compatible with software defined radio (SDR) hardware to build LTE networks. As the names suggest, srsUE implements the LTE UE and srsENB the LTE eNB. To accompany these, Software Radio Systems (SRS) has also published srsEPC which provides an Evolved Packet Core (EPC) that is needed for a fully working LTE network with one or several eNBs serving one or several UEs. At the time of experimentation, we are using the most current version of srsLTE, version 18.12.0 based on commit 3cc4ca85 from the master branch [14].

CSAI implements the ability to capture cellular paging messages and information blocks along with capabilities to stress test RRC connections. Both of these features are achieved by modifying the srsUE source code. Specifically, we modify the code that implements the RRC protocol. The RRC protocol is primarily responsible for connection establishment and release as well as handling paging messages. In the `connection_request` function, `send_con_request` is called which is responsible for sending the RRC connection request message to the lower layers that is transmitted to the eNB. If we comment this function and replace it with a call to `rrc_conection_release`, we instruct the UE to remain disconnected and not communicate with the eNB. This alone allows capturing the SIBs transmitted by the eNB, but is not enough to capture paging messages.

In order to capture paging messages, we add an additional line after the connection release call to update the RRC state to reflect a successful connection. The other layers of srsUE will now look for paging messages and they are automatically captured and logged. Figures 2 and 3 give examples of captured information blocks and paging messages.



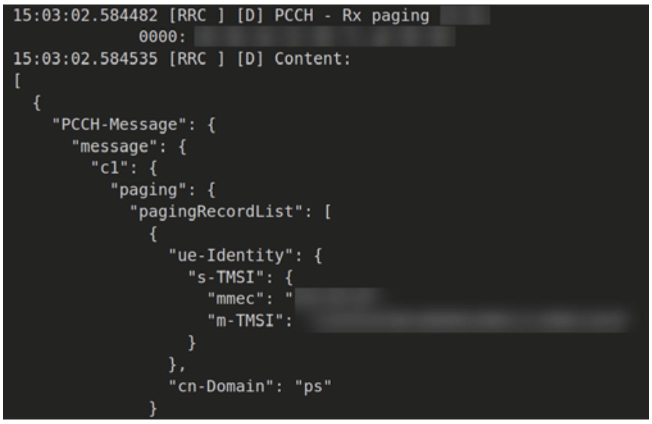
```
No.     Time               Protocol          Length Info
      1 20:03:00.680514    LTE RRC DL_SCH        43 SystemInformationBlockType1
      2 20:03:00.736021    LTE RRC DL_SCH        47 SystemInformation [ SIB2 ]
      3 20:03:01.065896    LTE RRC DL_SCH        28 SystemInformation [ SIB3 ]
      4 20:03:02.104689    LTE RRC PCCH          43 Paging (4 PagingRecords)
      5 20:03:02.264601    LTE RRC PCCH          24 Paging (1 PagingRecords)
      6 20:03:02.504363    LTE RRC PCCH          30 Paging (2 PagingRecords)
      7 20:03:02.584407    LTE RRC PCCH          24 Paging (1 PagingRecords)
      8 20:03:02.664449    LTE RRC PCCH          24 Paging (1 PagingRecords)
      9 20:03:02.744490    LTE RRC PCCH          24 Paging (1 PagingRecords)
     10 20:03:02.984571    LTE RRC PCCH          24 Paging (1 PagingRecords)

▸ Frame 4: 43 bytes on wire (344 bits), 43 bytes captured (344 bits)
  DLT: 147, Payload: mac-lte-framed (mac-lte-framed)
▾ MAC-LTE PCH PDU (28 bytes)
  ▸ [Context (RNTI=65534)]
  ▾ LTE Radio Resource Control (RRC) protocol
    ▾ PCCH-Message
      ▾ message: c1 (0)
        ▾ c1: paging (0)
          ▾ paging
            ▾ pagingRecordList: 4 items
              ▾ Item 0
                ▾ PagingRecord
                  ▸ ue-Identity: s-TMSI (0)
                    cn-Domain: ps (0)
```

[redacted]

Fig. 2. Wireshark analysis of paging messages.

V. Analysis

In order to benchmark CSAI, we capture commercial network traffic and provide masked statistics to show the effectiveness of the tool. Two USRP B210s with the modifications detailed in Section IV were used to capture SIBs and paging traffic on an Ubuntu 18.04.02 computer. Similarly to [15] and [16], only SIB and paging messages were acquired; careful consideration was taken to ensure that no user data was captured or retained despite being encrypted.



```
15:03:02.584482 [RRC ] [D] PCCH - Rx paging [redacted]
             0000: [redacted]
15:03:02.584535 [RRC ] [D] Content:
[
  {
    "PCCH-Message": {
      "message": {
        "c1": {
          "paging": {
            "pagingRecordList": [
              {
                "ue-Identity": {
                  "s-TMSI": {
                    "mmec": "[redacted]
                    "m-TMSI": [redacted]
                  }
                },
                "cn-Domain": "ps"
              }
```

Fig. 3. Log output of paging message.

A. Short-Term Persistence

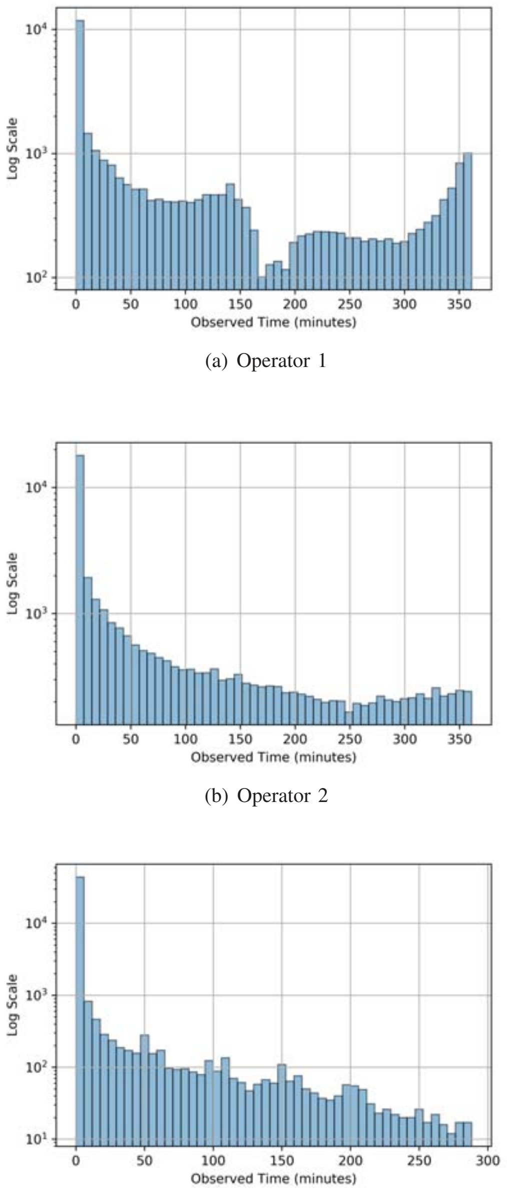
Table I shows the data that we obtained from three networks. We measured the amount of total paging traffic over a five to six hour time frame of two network operators the first day, and repeated the same capture for the third operator on the following day. By inspecting the paging information to see whether IMSIs were used to page users, we found that all pages used S-TMSIs to identify a UE as opposed to revealing the IMSI, which is an encouraging result.

The last row in Table I indicates the longest observed TMSIs in minutes which match the lengths of the experiments. In all three cases, several TMSIs were observed throughout the capture, but the majority of TMSIs were either a single occurrence, or were used for a short time.

TABLE I
Network Statistics.

Metrics	**Network Operators**		
	Operator 1	*Operator 2*	*Operator 3*
Total Pages	586701	280795	156311
Unique TMSIs	31654	36544	49076
Longest active TMSI in minutes	361.25	361.04	288.15

Figure 5 shows the histograms of the lifespans of the observed TMSIs. Most TMSIs are very short lived, whereas some are observed for the entire duration of 6 hours. Our measurements were taken at a single location and we had no control of the UEs in the area. Due to the mobility of users, it is likely that the average TMSI lifespan is longer than shown here. Operator 1 has a significant number of long-lasting TMSIs. This implies that many UEs attached to this cell did not hand-off connectivity during our experiments, or that cellular Operator 1 does not rotate TMSIs as frequently.



(a) Operator 1

(b) Operator 2

(c) Operator 3

Fig. 4. Time distribution of observed Paging messages.

B. Long-Term Persistence

We examine the persistence of TMSIs and for this use two commercial UEs and CSAI. We initiate communication from one UE to the other and capture the paging messages. This is accomplished by sending numerous messages from one UE to the other with ample time between messages so that the RRC connection is released due to UE inactivity. The next day we repeated a similar communication pattern to generate more paging messages to our UE.

We review the log files to see if our TMSIs from day one persisted on the following day. While the test was limited in scope, we did not observe any of the TMSIs from day one in the day two logs. This is a particularly encouraging result, as it implies that this network operator rotates TMSIs at least every day.

C. Potential Attacks

CSAI takes advantage of the inherent nature of pre-authentication and broadcast signaling in LTE. While paging messages do not inherently contain sensitive information, it is possible to map a TMSI to a RNTI if you monitor subsequent RRC connection setup requests. Once a mapping is obtained, an attack as described in [17] could allow for statistical traffic analysis even though the content of the NAS messages are encrypted.

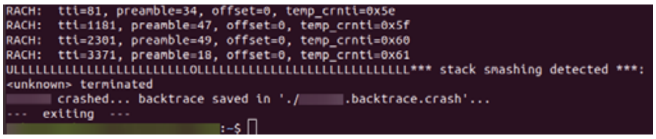
Lichtman et al. outline several attacks in [1]. Once the MIB/SIBs are decoded, it is possible to target jamming efforts towards a specific eNB. Combined with the aforementioned TMSI to RNTI mapping, it would be possible to extend the attack and jam one or several UE's data and control plane traffic.

Most network operators will page a UE using a TMSI; however, the 3GPP standard allows eNBs to page a UE using its IMSI in cases where a UE does not respond to three subsequent paging attempts using a TMSI. This presents a significant security issue as many follow-on attacks are capable once a UEs IMSI is known and include down bidding attacks or man in the middle style interceptions as demonstrated in [7] and [8]. Our instrument enables research on security analysis and system hardening. Researchers will benefit from CSAI as they test modifications to 4G and 5G protocols to prevent the exploitation of pre-authentication messages [18].

D. Crashing a Software eNB

Another interesting behavior that we observed in the course of developing CSAI was the potential for a straightforward denial of service attack against an SDR eNB through active RF attacks that mimic older Transmission Control Protocol (TCP) synchronize (SYN) flood attacks. When modifying the RRC handshake source such that a UE sends an RRC Connection Request to an eNB and immediately begins the RRC Connection Release process, the UE will not respond to the eNBs request for RRC Connection Setup. The eNB will allocate resources for the UE expecting the UE to reply with an RRC Connection Setup Complete; however, the UE has already begun the process of releasing the connection. This leaves the base station in a half open state waiting for the UE to finish the RRC handshake. Since the UE was instructed to release the RRC connection, after a short delay it will attempt to reconnect to the eNB, further exhausting its resources. A fake UE could constantly request and release RRC connections thus identifying the eNBs maximum capacity for RRC connections, and potentially crashing the eNB

We performed this attack against an SDR eNB and observed that it crashed due to automated buffer overflow protections enabled by default when using the GNU C Compiler. An example of this crash is shown in Fig. 6.



```
RACH:  tti=81, preamble=34, offset=0, temp_crnti=0x5e
RACH:  tti=1181, preamble=47, offset=0, temp_crnti=0x5f
RACH:  tti=2301, preamble=49, offset=0, temp_crnti=0x60
RACH:  tti=3371, preamble=18, offset=0, temp_crnti=0x61
ULLLLLLLLLLLLLLLLLLLLLLLLLOLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL*** stack smashing detected ***:
<unknown> terminated
[redacted] crashed... backtrace saved in './[redacted].backtrace.crash'...
---  exiting  ---
[redacted]:~$ 
```

Fig. 5. Crash of a software eNB.

Whereas more testing is required to determine the scope of this active attack, one potential mitigation may be similar to how SYN cookies mitigate TCP SYN Flood attacks as detailed in [19], where the eNB would only allocate resources for RRC connections after the UE responds with the full Setup Complete message. It is possible to modify CSAI to include a delay that ensures the eNB does not crash, but rather denies service to legitimate UEs that are connected or are trying to connect. This finding exemplifies the potential of fuzzing attack analysis against cellular networks. Ideally, commercial equipment would have timeout thresholds to prevent a malicious RRC flood attack. We are currently investigating this using commercial femtocells with the objective to evaluate their protection mechanisms against the attacks demonstrated above and those discussed in [20].

VI. Conclusions

This paper describes how an SDR LTE implementation of a cellular RAN can be repurposed for analyzing the security of wireless networks. Our example is for 4G LTE, but similar principles can be applied to other cellular communications protocols. Using the modifications described in Section IV, SIBs and paging messages are able to be passively recorded from experimental or commercial LTE networks. We also discussed another modification that allows for testing RRC layer limits which can result in a denial of service attack.

CSAI [21] is a community resource that can be used and modified for supporting research using advanced wireless testbeds, such as AERPAW [22]. In continuing research, we are using CSAI as we investigate practical attacks against UE and eNB implementations and their remedies. These include base station/small-cell fuzzing, location leakage, and UE denial of service attacks and their countermeasures.

Acknowledgement

The work of V. Marojevic was supported in part by the NSF Platforms for Advanced Wireless Research (PAWR) program under grant number CNS-1939334. The work of T. Byrd was supported in part by Mississippi State University's James Worth Bagley College of Engineering, Bobby and Judy Shackouls Honors College, Department of Electrical and Computer Engineering, and NSF's CyberCorps(R) Scholarship for Service program under grant number DGE-1565484.

References

[1] M. Lichtman et al., "LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation," *IEEE Commun. Mag.*, vol. 54, no. 4, pp. 54–61, 2016.

[2] R. P. Jover and V. Marojevic, "Security and protocol exploit analysis of the 5g specifications," *IEEE Access*, vol. 7, pp. 24 956–24 963, 2019.

[3] R. M. Rao, S. Ha, V. Marojevic, and J. H. Reed, "LTE PHY layer vulnerability analysis and testing using open-source sdr tools," in *IEEE MILCOM 2017*, pp. 744–749.

[4] V. Marojevic, R. M. Rao, S. Ha, and J. H. Reed, "Performance analysis of a mission-critical portable lte system in targeted rf interference," in *IEEE VTC-Fall 2017*, pp. 1–6.

[5] M. Labib, V. Marojevic, J. H. Reed, and A. I. Zaghloul, "Enhancing the robustness of lte systems: analysis and evolution of the cell selection process," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 208–215, 2017.

[6] S. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino, "LTEInspector: A systematic approach for adversarial testing of 4G LTE," in *Network and Distributed Systems Security (NDSS) Symposium 2018*, 2018.

[7] A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J.-P. Seifert, "Practical attacks against privacy and availability in 4G/LTE mobile communication systems," *arXiv preprint arXiv:1510.07563*, 2015.

[8] R. P. Jover, "Lte security, protocol exploits and location tracking experimentation with low-cost software radio," *arXiv preprint arXiv:1607.05171*, 2016.

[9] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation," Technical Specification (TS) 36.211, 03 2013, version 10.7.0. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2425

[10] T. Engel, "Locating mobile phones using signalling system 7," in *25th Chaos communication congress*, 2008.

[11] A. Shrut, "LTE for layman (part 3) - the complete picture!" 2016, [Online; posted 15-June-2016].

[12] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC); Protocol specification," Technical Specification (TS) 36.321, 04 2019, version 15.5.0. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2437

[13] I. Gomez-Miguelez et al., "srslte: an open-source platform for lte evolution and experimentation," in *ACM WiNTECH*, 2016, pp. 25–32.

[14] Software Radio Systems, "srslte," https://github.com/srsLTE/srsLTE/tree/3cc4ca851a18b15234d849a5a4a8f9bf0768d30f, 2019.

[15] S. R. Hussain, M. Echeverria, O. Chowdhury, N. Li, and E. Bertino, "Privacy attacks to the 4G and 5G cellular paging protocols using side channel information," 2019.

[16] M. Chlosta, D. Rupprecht, T. Holz, and C. Pöpper, "LTE security disabled misconfiguration in commercial networks," in *Proceedings of the 12th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, ser. WiSec '19, 2019.

[17] D. Rupprecht, K. Kohls, T. Holz, and C. Pöpper, "Breaking LTE on layer two," in *IEEE Symposium on Security & Privacy (SP)*, 2019.

[18] M. A. Ferrag, L. Maglaras, A. Argyriou, D. Kosmanos, and H. Janicke, "Security for 4g and 5g cellular networks: A survey of existing authentication and privacy-preserving schemes," *Journal of Network and Computer Applications*, vol. 101, pp. 55–82, 2018.

[19] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach (7th Edition)*, 7th ed. Pearson, 2016.

[20] H. Kim, J. Lee, L. Eunkyu, and Y. Kim, "Touching the untouchables: Dynamic security analysis of the LTE control plane," in *Proceedings of the IEEE Symposium on Security & Privacy (SP)*, May 2019.

[21] T. Byrd, "CSAI," https://github.com/tkb140/CSAI, 2019.

[22] V. Marojevic, I. Guvenc, M. Sichitiu, and R. Dutta, "An experimental research platform architecture for UAS communications and networking," in *IEEE VTC2019-Fall*, Sep. 2019, pp. 1–5.