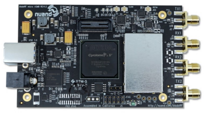
BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2022
http://excel.fit.vutbr.cz

Building Intrusive Device Against LTE Mobile Networks

Timotej Kamenský*



Abstract
This work strives to create an intrusive device targeting LTE networks. It should implement several already described attacks: IMSI catcher, downgrade attack, denial of service. These attacks are taking use of inherent weaknesses of the LTE protocol, which makes defending against them is hard. This goal should be achieved in a compact hardware package. The key piece of hardware is Software Defined Radio (SDR), namely Blade-RF 2.0. It is a general-purpose radio which allows us to work with radio technology through an abstract programming layer instead of relying on tinkering with hardware. The result of this work can serve the security community in penetration testing and researching security of mobile networks.

Keywords: LTE — SDR — IMSI catcher — DoS — Downgrade Attack

Supplementary Material: Code Repository (WIP) — s Conference media

*xkamen24@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

High speed wireless networks have already become a backbone of civilisation as we know it. They provide an essential service of communication wherever you go. Their security and availability are therefore of a high importance.

The mobile networks (namely those used by the general public) are based on GSM (2G), UMTS (3G) and LTE (4G) protocols. The fact that they are wireless opens new vectors for an attack. GSM networks are widely considered insecure [1]. LTE however has higher security thanks to its encryption standards. Not only that - while GSM is considered a backup or a legacy system, LTE is the standard regularly used by people in their day to day lives. And since it is a new system, its lifespan is also longer. Considering this, it makes sense to focus research on LTE protocol.

When it comes to Wi-Fi penetration testing, one of the more polished solutions is Wi-Fi pineapple. It is a hardware-software package, which offers an easy way to launch attacks in a matter of seconds. There are no widely available tools like that for LTE. There are attacks already developed, targeting L1-L3 of LTE. This work does not want to develop new attacks.

We propose to create a prototype of an intrusive device, implementing the following attacks:

- IMSI Catcher, which saves the international mobile subscriber identity (IMSI) of phones in the area
- Denial of Service, which can block mobile services for the attaching phones in the area (and can be targeted by using IMSI)
- Downgrade attack, which forces the mobile devices to use 2G services, which are less secure.

While the attacks were described and implemented in the past, each of them has its problems, disadvan-

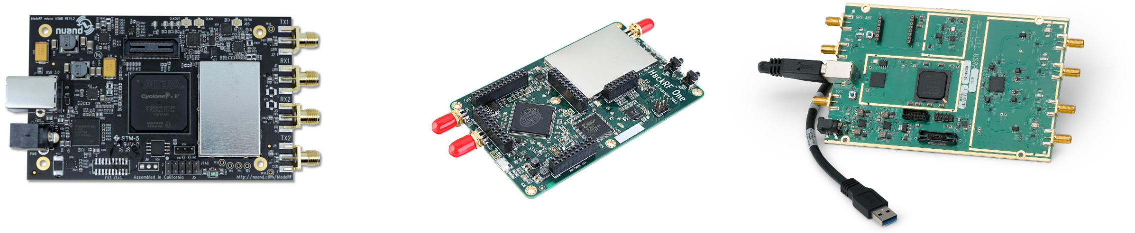


Figure 1. The best choices for an SDR. From left to right: bladeRF, HackRF One, Ettus Research B210.

tages or caveats. Some of the previous researchers have not given away publicly their source code [2]. Other work was implemented on old versions of libraries and no longer supported [3], or it was developed using different, more expensive SDR hardware (the most popular by far being Ettus B2x0). Additionally, we have not found a single example that would have implemented multiple attacks.

Because of this, our work aims to offer an all-in-one, easy to use package; implementing the proposed attacks using a single application. Such a package should be modestly sized, powered by battery, and practically usable in the field for penetration testing.

In Section 2, we will take a look at state of the art in hardware and software solutions. In Section 3, we will describe the proposed attacks in depth. In Section 4 we describe the taken implementation.

And finally in Section 5, we will summarise and conclude this work.

2. State of the Art

2.1 Available Radios

Radio Overview

While mobile networks are ever-present, the hardware accessible to a programmer to access these networks is very narrow. Using a mobile phone is not an option, because it requires routing of the mobile phone and even then the results are nondeterministic. Additionally, there is no support or community for such solution. Other than mobile phones, capable radio equipment is generally manufactured for the needs of telecommunications companies.

Therefore, it is large, expensive and inflexible (the direct opposite of what we need).

Software Defined Radio

There is, however, a third path. Software defined radio (SDR) is a type of general purpose radio, which replaces more traditional hardware electronic circuit parts with their software-driven equivalents. This has wide flexibility of use (generally, radios can operate on frequencies from just a few MHz to low GHz with wider bandwidths). Utilising this flexibility offers faster developement than with classical radio, and mostly allows for a compact package. All SDR radios we approached had a GNU Radio library support [4], which is the basis for their use on Linux-based PCs. Some of them have support from mobile network developers, such as srsRAN [5], as well. All of these factors make SDR an ideal platform for such our application.

As with classic radios, SDRs come in many shapes and sizes. They are mostly differentiable by their ability to receive and/or transmit signal, possible wavelengths, use or capabilities of FPGA chip, bandwidth and sampling frequency, just to name a few.

The most popular SDR is RTL-SDR [6]. It is a derivative of many different DVB-T receivers, all based on RTL2832U chipset. It is very cheap (20-30 €), and popular among the radio amateurs. It offers very low barrier entry to the ham radio or SDR community. While this does not meet our requirement for transmission capability, such a cheap and readily available system can find its applications.

Compatible Radio Hardware

Our SDR needs to be capable of full-duplex (transmitting and receiving signal at the same time), and has to be able to operate in frequencies used by LTE (ideally up to 3,800 MHz, however 2,200 MHz should be sufficient for most of them in Czechia). Additionally, strong support and thriving community around them is wholeheartedly welcomed. This reduces our choices to just a few possibilities outlined below.

The most popular SDR for research and development is Ettus Research B210, the rightmost on image 1. It is capable of full duplex and operation up to 6 GHz. Being targeted on serious research instead of hobbyist market, it is not a low-cost option, with the cheapest units going for 1200 €. [7]

Very comparable in the SDR low-end are HackRF [8] and BladeRF [9], also displayed on image 1. Both offer full duplex with frequency range going up to and above 3,5 GHz. The cheapest types sell for about 400 €. In our opinion, these two are the best choice for

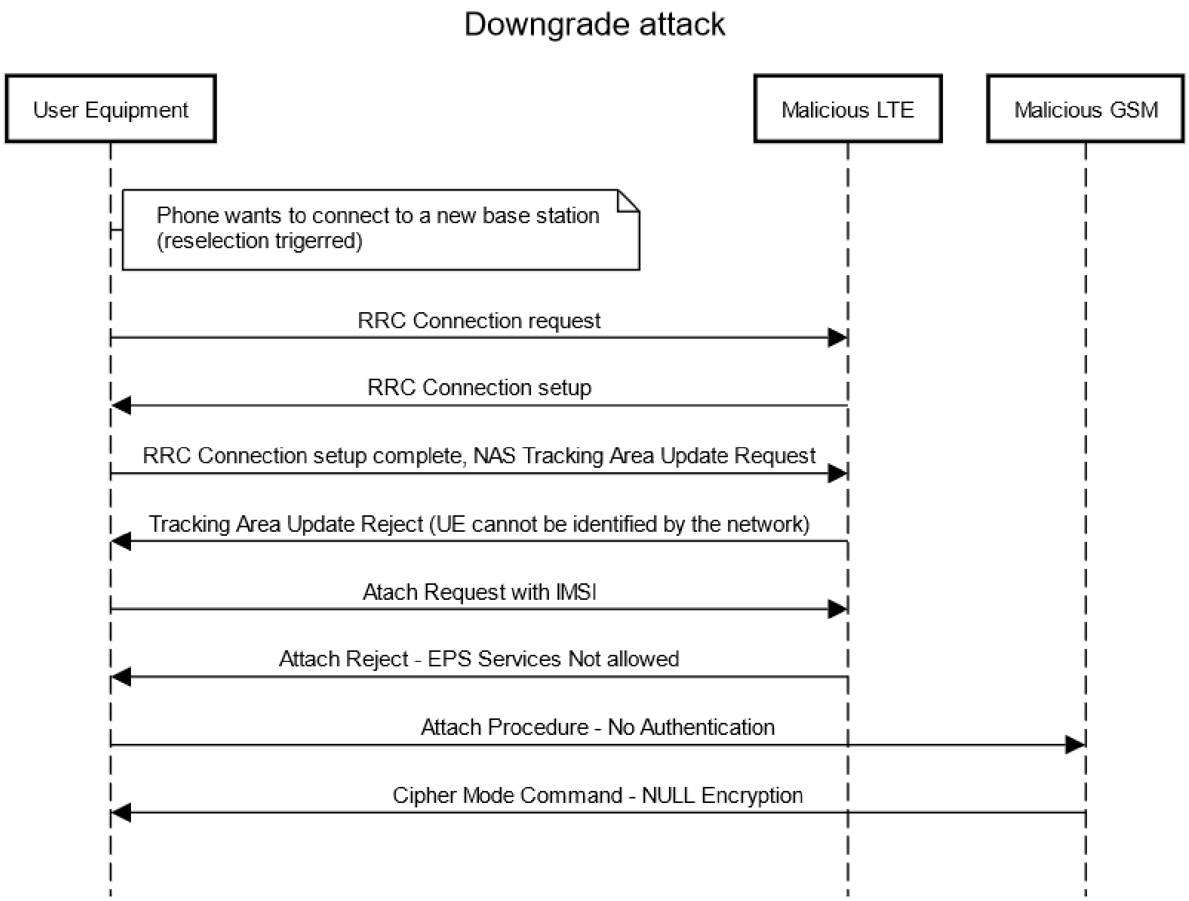

Downgrade attack
User Equipment
Malicious LTE
Malicious GSM
Phone wants to connect to a new base station (reselection trigerred)
RRC Connection request
RRC Connection setup
RRC Connection setup complete, NAS Tracking Area Update Request
Tracking Area Update Reject (UE cannot be identified by the network)
Atach Request with IMSI
Attach Reject - EPS Services Not allowed
Attach Procedure - No Authentication
Cipher Mode Command - NULL Encryption

Figure 2. Sequence diagram of a Downgrade attack. When the device tries to connect to our fake base station, it is refused to Attach to the network properly. Attach reject response is ”EPS Services Not Allowed”. Per protocol, this makes the phone use only GSM networks.

our experiment. While capabilities are similar, they are not equal. BladeRF is a lot matured product than HackRF. It also has wider software support and a more thriving community.

2.2 Compatible Controllers

After selecting the radio, we must pick the computer to connect to SDR. The controller should be able to run some version of Linux, since all of the underlying applications are primarily written for it. We also want the final prototype to have compact size (fits into the pocket range) and possibility of powering it by battery. Additionally, the HackRF and BladeRF require USB 3.0 connectivity to use full capabilities. The first idea that comes to mind in compact Linux computers with USB 3.0 connectivity is a Raspberry Pi 4 [10]. It is cheap with large and very dedicated community. Another alternative is RockPro64 [11]. Compared to the Raspberry, it offers more computational power in exchange for a lot smaller support of the device. [12]

2.3 Existing Solutions

The mobile networks used to be closed-source, big corporations endeavour only. In the last few years, this technology is now accessible to basically anyone. We identified two milestones, which allowed this process.

The first was advent of SDR for the general public, not just niche applications such as in the army. Without the SDR, the hardware was almost unattainable.

The second was creating open-source implementations of the GSM and LTE standards. The first was is OpenBTS [13], which implements GSM and was created in 2008. It is rather famous for enabling the creation of mobile networks without operators for the first time. For example, it allowed mobile network on remote islands or a desert during Burning Man festival. Since then, many more network tools were created, such as OpenAirInterface [14], OpenLTE [15] and srsLTE [5].Similarly to the OpenBTS, these projects allowed for creation of private mobile networks.

These open source protocol implementations allowed people to start tinkering with them. It lowered the cost of entry even for researchers as well. Instead of needing a cooperation with a mobile network corporation, and needing thousands of dollars worth of equipment, now the researchers need nothing more than a few SDRs and open source code.

This research has already described a number of

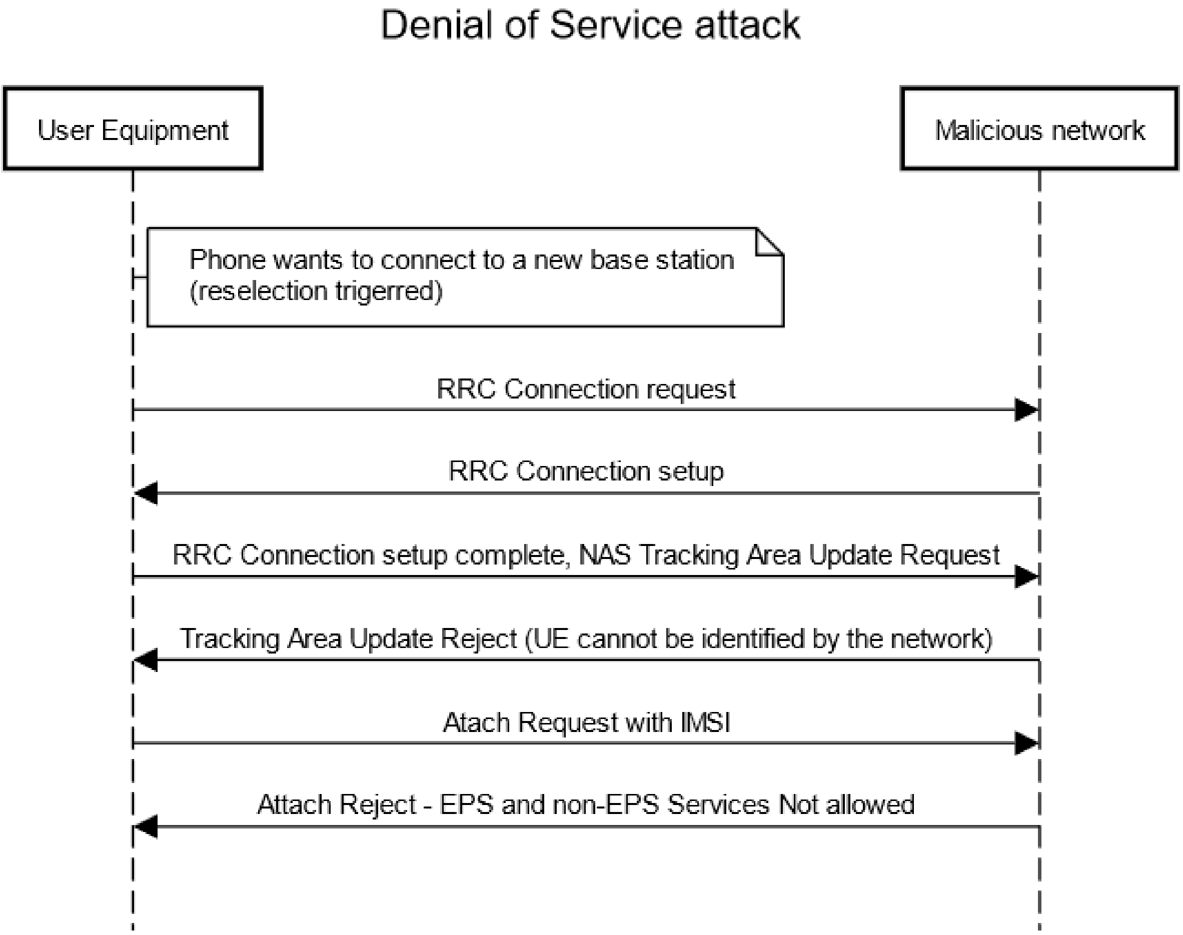

Denial of Service attack
User Equipment
Malicious network
Phone wants to connect to a new base station (reselection trigerred)
RRC Connection request
RRC Connection setup
RRC Connection setup complete, NAS Tracking Area Update Request
Tracking Area Update Reject (UE cannot be identified by the network)
Atach Request with IMSI
Attach Reject - EPS and non-EPS Services Not allowed

Figure 3. Sequence diagram of a device-targeted Denial of Service Attack. After trying to connect to our fake base station, attach process is rejected with message ”EPS and non-EPS not allowed. Per protocol, mobile phone will not attempt to connect to any network.

attacks on L1-L3 layers of the mobile network.

The first and one of the more straight forward attacks is creating a fake base station. Base station (a.k.a. eNodeB for LTE) is a middle-man between an users phone (usually called User Equipment, UE) and the mobile network. Base station software implementation needs to be compatible with corresponding data-link layer standard. In GSM networks, a fake base station can be used to establish a man-in-the-middle (MitM) attack. In 3G and LTE was introduced an authentication between an users‘ phone and the network. Without knowing cryptographic keys, it is not possible to conduct a MitM attack in these protocols. However, a fake base station can be used as a vector for other attacks and is a good way to start the research.

In all attacks using fake base station, we want the phone to connect or switch to our base station. What repeats in all of the attack scenarios is making the fake base station as attractive to connect as possible. This can be achieved with strategically setting region codes, operator code, cell identifier, using priority frequency, and others.

2.4 Attack Demonstrations

All attacks are explained and described in more detail in dedicated Section 3.

IMSI Catcher IMSI catcher is the most famous and basic attack from the whole bunch. It is possible in GSM and was demonstrated many times over. [2, 16]

Downgrade Attack This attack is a lot less popular. But its capabilities are possibly even greater. Simple Downgrade attack was demonstrated by J. Piqueras [17]. Its true potential was demonstrated by Lin Huang [18]. Using a downgrade, the attacker can force the target device to use his own GSM fake base station and forward his data onto a real one, thus establishing a full man-in-the-middle position. Similar feat was demonstrated by Chuan Yu et al. in 2019 [2].

Denial of Service Attack This attack was described here [17]. While it is similar to the other attacks in its functionality, its usefulness is not as high and it is a bit neglected as a result.

Signalling-based Denial of Service Signalling-based Denial of Service was researched and demonstrated again by Shuhui Chen et al. in 2019 [19]. LTE

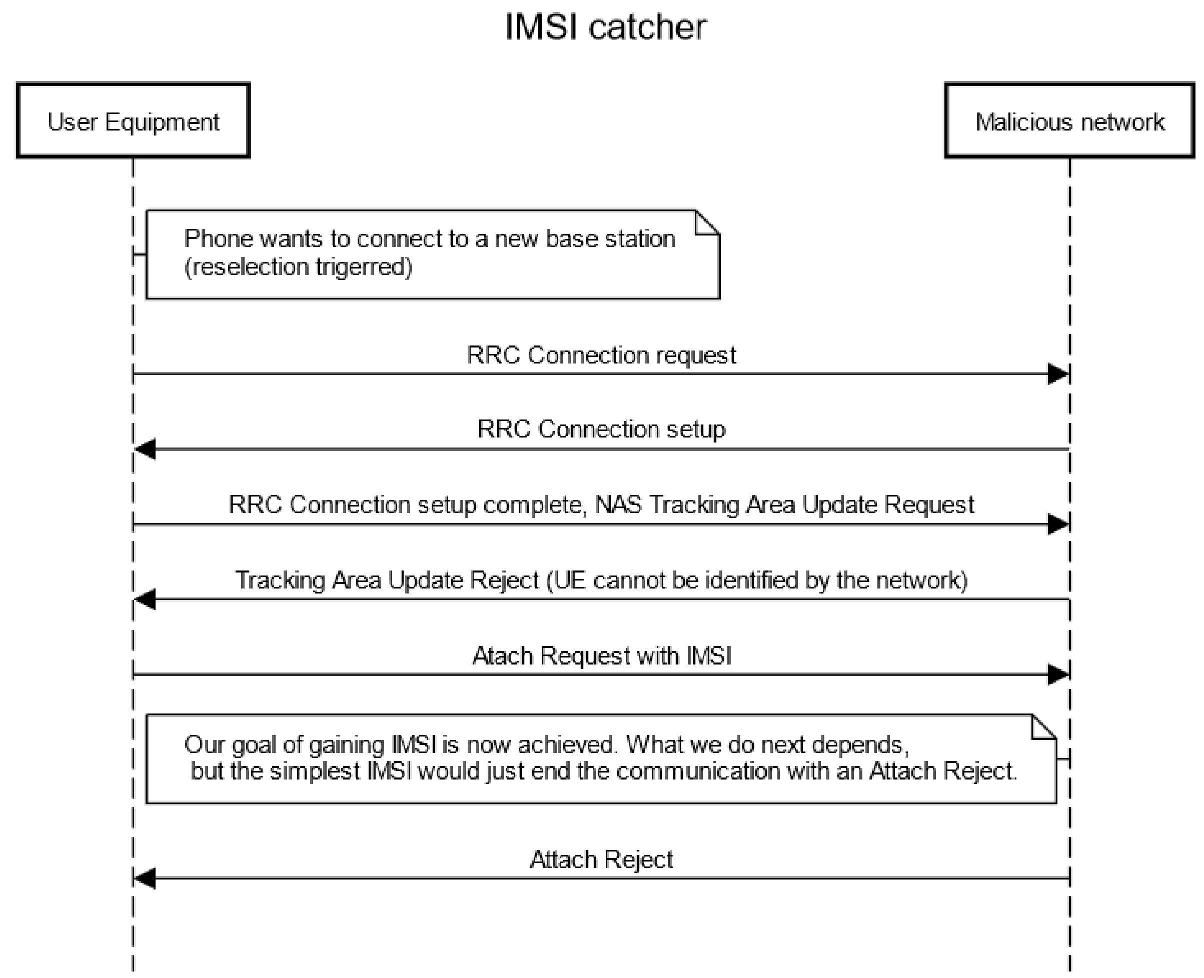

IMSI catcher
User Equipment
Malicious network
Phone wants to connect to a new base station
(reselection trigerred)
RRC Connection request
RRC Connection setup
RRC Connection setup complete, NAS Tracking Area Update Request
Tracking Area Update Reject (UE cannot be identified by the network)
Atach Request with IMSI
Our goal of gaining IMSI is now achieved. What we do next depends,
but the simplest IMSI would just end the communication with an Attach Reject.
Attach Reject

Figure 4. Sequence diagram of an IMSI catcher. When phone tries to connect to our fake base station, it sends us an IMSI. After getting an IMSI, we have reached our goal and do not need to finalize the attach procedure.

protocol relies on synchronisation signals for its communication. These signals serve the purpose of synchronisation during the establishment of the connection, and during an entire connection. This role is done by *Primary* and *Secondary Synchronisation Signals* (PSS, SSS), designed to be detectable even at low signal strength ratio. Hence, brute-force jamming these essential signals is hard. Instead, we can spoof this signal and desync the system. Spoofing the PSS signal has one of the highest complexity/efficiency ratio of any of the target-able channels in LTE. [20]

3. Proposed Attacks

3.1 Enablers of the proposed attacks

There are two problems with the protocols with huge implications, that we will use to our advantage.

1. Question of priorities in the design of mobile network protocols. The highest priority was always high availability and reliability. While security was always taken into account, it was never on the first place. This applies both to GSM and to LTE. LTE standard says that the mobile phones trust the base stations, at least before establishing an attach procedure, by default. This approach is discontinued in future 5G networks, as mobile phones will not trust the base station before a security handshake (key exchange).
2. Security, mainly encryption, of GSM and LTE are almost incomparable. There have been demonstrations of a real-time decryption of GSM voice services protected A5/1 or A5/2 encryption. A higher standard of encryption used in GSM, called A5/3, might not hold for long as well. Compared to these, LTE is well encrypted. [1]

3.2 Downgrade Attack

The goal of this attack to force the target device to use GSM instead of LTE. This allows us to passively eavesdrop on the communication. The other option is to actively create a fake GSM base station, make the target connect on it, and redirect the communication onto a legit network. The attacker gaining such position between the user and application is called Man-in-the-Middle Attack.

The attack diagram is shown on Figure 2 and works as follows: The attacker creates a fake base station. When the target device tries to connect, it is allowed to establish radio connection and send an attach request. This request is the one actually trying to connect to a mobile network through this base station. When this happens, the attacker refuses the connection, using *Attach Reject - No EPS Services Allowed* (this means no 3G and LTE services allowed). By protocol, the target device will obey and will not try to connect to LTE and 3G networks. It is also possible to order the target device not to use any encryption.

3.3 Device-targeted Denial of Service Attack

In this attack, we are able to deny service to a specific target device, based on its IMSI (International Mobile Subscriber Identity - unique identifier baked into a SIM card). This DoS has persistence based on the implementation of mobile device manufacturer. Most devices will not try to connect to any network until turned off and on again. Other devices have a timer set at random by flat distribution between 12 and 24 hours. However, there have been reports of some devices not connecting for a rather short 10 minutes. Therefore, the efficiency varies greatly.

This attack is surprisingly similar to previously mentioned downgrading attack.

The attack diagram is on Figure 3 and works as follows: The attacker creates a fake base station. When the target device tries to connect, it is allowed to establish radio connection and send an attach request. This request is the one actually trying to connect to a mobile network through this base station. When this happens, the attacker refuses the connection , using *Attach Reject - EPS and non-EPS Services not Allowed* (this means no GSM 3G and LTE services allowed). By default, the target device will obey and will not try to connect to any network.

3.4 Brute-force Denial of Service Attack

It is very hard to jam any base station without high-power antennas. Those, however, are is too big for comfort of use, too hungry for electricity to be carried around and too visible. Not only that, but LTE protocol works surprisingly well in high signal-to-noise (SNR) environments, which lowers the efficiency of the jamming even more. Therefore, brute force attack is not suitable for the low-power, flexible radio such is an SDR.

3.5 Smart Jamming Denial of Service Attack

The brute force jamming is very hard. However, knowing the inner workings of the LTE channel, one can strike where it hurts the most. Instead of trying to noise out the signal, we can add our own. There are different signalling channels in LTE, all with different functionalities. Injecting spoofed signals into these signalling channels can ruin broadcast with a lot higher efficiency. It has been found by a different research [21], that spoof jamming the *Control Format Indicator Channel* had the highest effect.

3.6 IMSI catcher

The attack is similar to Downgrade and Denial of Service. Its diagram is depicted on Figure 4.

Mobile devices in the network are identified using their IMSI. When a mobile device attempts to connect to a base station, this unique identifier is attached to the request. This can, in turn, be used for tracking the position of the mobile device. In this attack, the attacker creates a fake base station. When mobile devices try to connect, they attach an IMSI to this request. Based on this, we now know that this phone is in the vicinity.

4. Design and Implementation

4.1 Choosing Hardware Setup

Going back to the hardware solutions, the favourites are abundantly clear: Nuand BladeRF as an SDR, and Raspberry Pi 4 as the controller. Both are on the cheap side of things and can be powered by battery. Moreover, their combination had been tested by srsRAN for their compatibility.

In advanced uses, we must take the antenna used into consideration. During a research, this is not important, as any antenna will mostly fare well enough.

4.2 Software Design

Now is the time to look at the LTE stack. Its implementation is made of three discreet parts. The first is UE implementation, which has all of the user endpoint functionality. The second is implementation of a base station (eNodeB). It takes care of creating radio connection, routing traffic between UE and the EPC etc. In most cases, it acts as a stripped-down middleman between the EPC and UE.

The last is Evolved Packet Core (EPC) implementation, displayed on Figure 5. This is made of four parts.

- Mobility Management Entity (MME) is a brain of the entire operation. It authenticates the incoming UE connections for the eNodeB, controls connections for even multiple base stations and manages connection sessions.

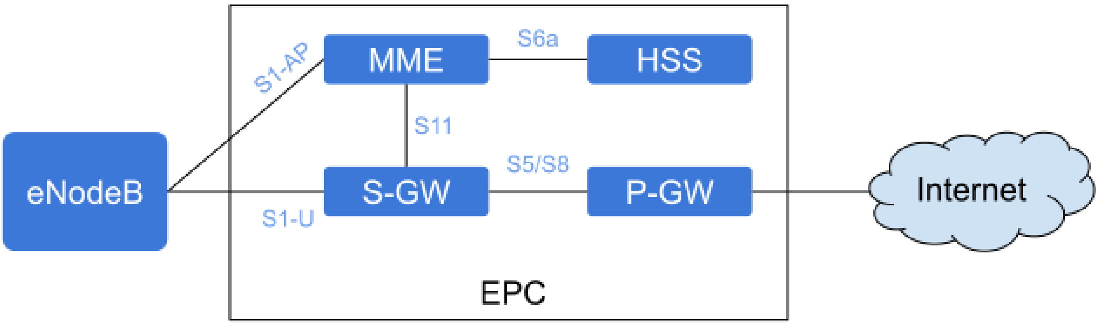


Figure 5. Diagram of components, relations, and used communication protocols between the components of EPC. [5] Of these, we only need to implement MME and its connection to eNodeB through S1-AP protocol.

- Packet Gateway, used as a gateway to the internet.
- Service Gateway, used to route traffic form the Packet Gateway to the device in LTE network.
- Home Subscriber service, that serves a role of server. There, it is possible to find valid clients with the required authentication information.

The Home Subscriber service serves us no purpose, since we cannot know the authentication keys and therefore cannot create a full man-in-the-middle fake base station. Following this logic, there will never be a connected phone to the internet, so both gateways are also without any use for us.

All of the proposed attacks use a weakness in the protocol during attach process. This process is fully in control of the MME - the base station serves only as a middleman. This means, that we only need to implement a relatively simple mock of MME. This mock will then respond to the incoming attach requests as required in our attack.

Even if we implement our own mocked MME, we still need it to connect to an eNodeB. In our experiments, we are using srsENB from the srsLTE stack [5]. This eNodeB needs to be set-up to attract connections, as outlined here: 2.3. Even though we are using srsRAN, eNodeB conforming to the protocol should be usable for this effect. Testing other eNodeB is also a subject of our future testing.

The protocol defines the connection between eNodeB and MME using S1 signalling, defined as: [22]

1. datalink layer shall not be prevented;
2. network layer supports both IPv4 and IPv6, and only point-to-point trasmissions are supported;
3. transport layer shall be supported by Stream Control Transmission Protocol (SCTP);
4. between a pair of eNodeB and MME, only one SCTP association shall be established (note: association being a terminus technicus for a connection on SCTP);
5. application layer is implemented using S1-AP messages over S1-MME interface.

The definition S1-AP is, in turn, defined by an ASN.1 language. [23]

4.3 Implementation

There are a few open-source libraries, that can help us with encoding and decoding these messages. One stands out in particular: Pycrate project by P1Sec offers both an SCTP python library [24], as well as S1AP python library [25].

Given the nature of the attacks, we need a fixed response to the given messages. Most of the messages have no way to be triggered, since there will never be an attached mobile phone. These can be ignored. Therefore, our implementation reduces to just a few messages. The first is S1 setup request and response, which we need to create a connection between eNodeB and MME. Another message is *InitialUEMessage* with attach request. And to store all the gathered information, we do not need more than to write it to a file. We need to implement Identity request as well. In case of the phone connecting with already established radio connection, it is assigned a Temporary Mobile Subscriber Identity, (TMSI), which is used instead of IMSI, so it is not trasmitted over the air ever so often. If, however, the MME has no record of the TMSI, it can ask for an IMSI instead. This is done using Identity request.

The final version of the program should, aside from pure functionality, simplify the complete setup. Right now, one must install GNU-radio, python, SDR drivers, and finally eNodeB implementation. This is somewhat cumbersome and should be simplified. Goal is to have a single script that will install all of the various dependencies. This will make it a truly simple tool to use.

4.4 Testing

Laws prevent radio transmission on licensed frequencies without license. We must therefore test the setup in a Faraday cage. In our experiments, a simple Faraday cages blocked ”only” around 30 dB worth of signal. To fully block LTE signal, we needed more like 80 dB. This is not an isolated experience, as suggested here [26].

This also makes legal testing possible in just laboratory conditions, which greatly limits the testing scenarios.

We were able to create an srsENB mounted on our chosen hardware and software implementation. In the laboratory conditions, we were able to recreate the attacks as described in previous attacks. At the time of writing, we had problem simulating condition where TMSI is already assigned. It will be described and updated later in code repository, linked at page 1.

5. Conclusions

In our work, we have described the LTE and GSM protocol security vulnerabilites. We used LTE open-source implementation, and explored using it with Software Defined Radio in applications that require compact size. At last, we implemented our own mock of MME, which implements three previously described attacks on LTE protocol: IMSI catcher, Downgrade attack, and Denial of Service. This work can be for example used in penetration testing of LTE networks. The vulnerabilities are caused by accepting control messages in LTE before security handshake, which is fixed in the next-gen 5G networks.

Acknowledgements

I would like to thank my supervisor for his kind help in this project. It is quite a ride, and I’ve been given every support I could ask for.

References

[1] Orr Dunkelman, Nathan Keller, and Adi Shamir. A practical-time attack on the a5/3 cryptosystem used in third generation gsm telephony. *IACR Cryptol. ePrint Arch.*, (N/A):13, 2010.

[2] Chuan Yu, Shuhui Chen, and Zhiping Cai. Lte phone number catcher: A practical attack against mobile privacy. *Security and Communication Networks*, 2019(1), 2019.

[3] ”Koh Heng Woon”. Github - wooniety/srslte-sniffer: Stuff for srslte imsi catcher. github.com/Wooniety/srsLTE-Sniffer.

[4] Gnu radio - the free and open source radio ecosystem · gnu radio. www.gnuradio.org/.

[5] srsran - your own mobile network. https://www.srslte.com/.

[6] Rtl-sdr (rtl2832u) and software defined radio news and projects. www.rtl-sdr.com/.

[7] Usrp b210 usb software defined radio (sdr) - ettus research. ettus.com/all-products/ub210-kit/.

[8] Hackrf product line - great scott gadgets. greatscottgadgets.com/hackrf/.

[9] Products archive - nuand. nuand.com/shop/.

[10] Raspberry pi 4 model b — raspberry pi. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/.

[11] Rockpro64 — pine64.org. https://www.pine64.org/rockpro64/.

[12] https://www.electromaker.io/blog/article/rockpro64-vs-raspberry-pi-4. https://www.electromaker.io/blog/article/rockpro64-vs-raspberry-pi-4.

[13] Openbts — open source cellular infrastructure. http://openbts.org/get-the-code/.

[14] Openairinterface — 5g software alliance for democratising wireless innovation. openairinterface.org/.

[15] openlte - sourceforge. sourceforge.net/projects/openlte/.

[16] Christian Sørseth. Location disclosure in lte networks by using imsi catcher. Master’s thesis, NTNU, 2017.

[17] Roger Piqueras Jover. Lte security, protocol exploits and location tracking experimentation with low-cost software radio, 2016.

[18] Hitb2016ams d1t1 forcing a targeted lte cellphone into an eavesdropping network - lin huang. https://www.youtube.com/watch?v=hNDChDM1hEE.

[19] Chuan Yu and Shuhui Chen. On effects of mobility management signalling based dos attacks against lte terminals. In *2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8, 2019.

[20] Marc Lichtman, Roger Piqueras Jover, Mina Labib, Raghunandan Rao, Vuk Marojevic, and

Jeffrey H. Reed. Lte/lte-a jamming, spoofing, and sniffing: threat assessment and mitigation. *IEEE Communications Magazine*, 54(4):54–61, 2016.

[21] Roger Piqueras Jover. Security attacks against the availability of lte mobility networks: Overview and research directions. In IEEE, editor, *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pages 1–9, 2013.

[22] Lte, (e-utran) s1 signalling transport 3gpp ts 36.412, 7 2018.

[23] Lte evolved universal terrestrial radio access network (e-utran) s1 application protocol (s1ap) (3gpp ts 36.413 version 13.6.0 release 13). 2017.

[24] P1Sec. Github - p1sec/pysctp. `https://github.com/P1sec/pysctp`.

[25] P1Sec. Github - p1sec/pycrate. `https://github.com/P1sec/pycrate`.

[26] Do faraday cages block cell signal? — quora.com. `https://www.quora.com/Do-Faraday-cages-block-cell-signal`.