The Detection Technology of LTE based Stratified Fuzz

Jun Yang[1,2], Haixia Yang[1,2], Qinshu Xiao[1,2],
[1]School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China
[2]National Engineering Laboratory for Mobile Network Security, China
{yangjun, yanghaixia, xiaoqinshu}@ bupt.edu.cn

***Abstract*—Fuzz test usually used in detecting network protocol vulnerabilities, Though that common fuzz test can cover as many as testing cases, its efficiency is relatively low. It may be spend many time to detect an aspect of a protocol. For this problem the paper put forward a more efficient method based on common fuzzing test. This method is applied for LTE protocol because it is raised against the features of LTE protocol. The paper in-depth studied the structure and process of GTP protocol, and designed stratified Fuzz testing process for the detection of GTP protocol to prove that the detection technology of LTE based stratified Fuzz is feasible and more efficient compared to common Fuzzing.**

1 Introduction

In recent years, 4G network has fully penetrated into all aspects of people's life. At the same time, the army also starting their dedicated LTE network. If the security of 4G network can't be guaranteed[1], it will cause a serious threat to the people's life and national security. Excavating exploitable security vulnerabilities of 4G protocol can deter hostile in military, at the same time, make our network more safely.

There are usually two vulnerability mining technology. One is analysis by researchers completely, and the other one is using formal tools. The most common formal tools is fuzzing[3][4]. Li Weiming, Huazhong University of Science and Technology, realized automated detection of fuzz test, and successfully detected many vulnerabilities of EM protocol and ISQLPlus protocol and so on, some of which has been published, but the other haven't been detected before[5]. Zhang yajun and Li zhoujun, Beijing University of Aeronautics and Astronautics, have brought forward a distributed model for automated white-box fuzzing[6]. Feng shengbo, Beijing University of Posts and Telecommunications, posed an improved fuzzing test methods, and successfully detected the unknown RLC protocol loopholes. Though there are many example of fuzzing, many application of fuzz test are focused on Internet Protocol, particularly in authentication protocol, and research of fuzz in LTE network protocol is still relatively small[7].

On the other hand, traditional fuzzing input random case to test, which improves the comprehensiveness of testing. It is beneficial to detect somewhere researcher difficult to detect by themselves[8][9]. But this is why fuzzing has low efficiency. The improved fuzz test method present are not fully applicable for LTE

L. Barolli et al. (eds.), *Advances on Broad-Band Wireless Computing, Communication and Applications*, Lecture Notes on Data Engineering and Communications Technologies 2, DOI 10.1007/978-3-319-49106-6_39

communication protocols, so this paper raise a fuzzing test method for LTE protocol from the hierarchy and priority assignment, based on characteristics of LTE protocol. It can improve the efficiency of fuzzing in mining LTE protocol vulnerability.

2 LTE Protocol Layer Architecture

There are three part will in this paragraph. The first part is the architecture of LTE protocol layer, which will introduce stratified LTE protocol in user plane. The second part will describe GTP protocol which is High-level protocol above TCP/IP. The last part will give an account of the characteristics of LTE protocol.

2.1 Architecture of LTE Protocol Layer

LTE network protocol stack is same to Internet protocol stack that they all adopt the thinking of hierarchical. LTE network protocol stack in user plane is structured as figure 1:

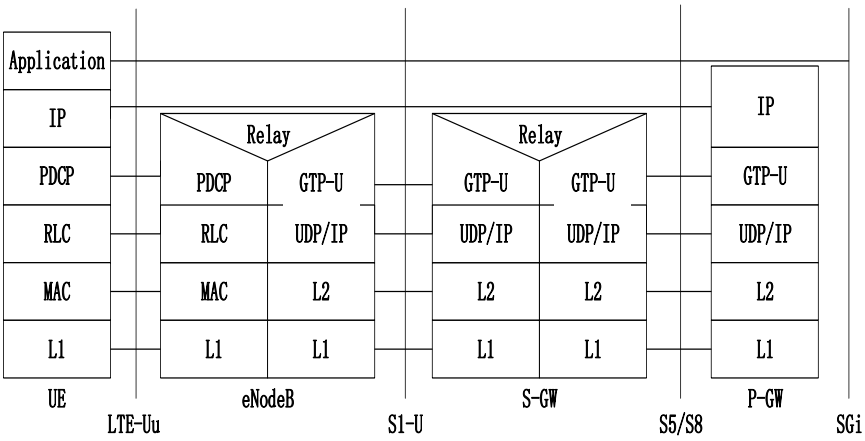

Figure 1. The architecture of LTE network protocol stack in user plane

LTE network protocols can be divided into several layers above. High-level protocols including RLC, PDCP, GTP protocols in LTE network achieve routing through IP address[2]. When transmitting data, the upper layer packets, encapsulated by lower layer protocol and marked the underlying protocol's header, pass down to the physical layer. When receiving data, the upper layer protocol receive lower layer packets to deal and remove the lower layer protocol's header, then transfer them to the upper until they arrive the destination node[10].

The above description outline the overall architecture of the LTE network protocol. For each protocol, they all has a fixed format and field. This paper will bring a brief analysis LTE protocol by an example of high-level protocol GTP.

2.2 GTP Protocol

GTP protocol is the abbreviation of GPRS tunneling protocol. As same as most tunnel technologies, GTP is a high-level protocol above TCP/IP or UDP/IP. And it is transparent for the router when providing the end to end communication between hosts. In TCP / IP protocol stack, GTP can even be understood as an application layer protocol. GTP protocol use tunnel identified TEID multiplex on the network path [11]. GTP protocol can be divided to GTP-U、 GTP-C and GTP` protocols according to different functions. GTP-U is a user-level protocol used to transmit user data, GTP-C is a control plane protocol for the management of GTP tunnel, and GTP' is for charging.The fellowing picture is GTP protocol format.

version number	protocol type	(*)	E	S	PN
Message Type					
length (1st Octet)					
length (2nd Octet)					
TEID(1st Octet)					
TEID(2nd Octet)					
TEID(3rd Octet)					
TEID(4th Octet)					
sequence number(1st Octet)[1) 4)]					
sequence number(2nd Octet)[1) 4)]					
N-PDU number[2) 4)]					
Next Extension Header Type[3) 4)]					

Figure 2. GTP head thumbnail

• The version number field is used to determine the GTP protocol version. In GTP-C protocol, P-GW will throw away the message when the version number isn't supposed by GTP-C protocol.

• Protocol type field is used to distinguish the GTP protocol and the GTP` protocol.

• Message type defines a number of message in GTP-C and GTP-U protocol to manage the path.

• TEID field clearly identifies the endpoint of the GTP-U or GTP-C protocol in the other side of the tunnel. It is used to transfer GTP packet multiplex on the tunnel between S-GW and P-GW.

2.3 The Characteristics of LTE Protocol

Based on the above analysis we can see that each protocol has its fixed protocol format, specifically in 3GPP protocol specification[2]. The different functions of each field are defined in the agreement. For example, GTP protocol version number field defines the GTP protocol version, and message category (GTP protocol or GTP 'protocol) are defined by the protocol type field and so on, but there are some fields have correlation, such as field E which defining whether the packet has extension header. If the value of field E is 0 then the following field Next Extension Header Type will not make sense. Field S defines that whether the packet use sequence and if this bit is set as zero, as the same, the following field sequence number will lose significance.

In summary, the characteristics of 4G protocol is:

- It's format is fixed, divided into different fields;
- It's fields can be relevant. Through there are also independent fields.

3 The Algorithm of Stratified Fuzzing

According to analysis for the characteristics of LTE protocol, the paper put forward a new fuzzing algorithm which use thinking of stratification and setting prioritization in the light of LTE protocol's characteristics.

3.1 Policy of Stratification

Depending on the function of each protocol field value, we can put those fields to different levels to test anew.

Specific hierarchical rules:

1) Unrelated fields, which have separate functions, not associated with other fields, can be divided into single layer;

2) Relevant fields, which values may affect whether the value of the second field is valid, are divided into a layer.

The following picture describe in detail the layering strategy of GTP protocol as example.

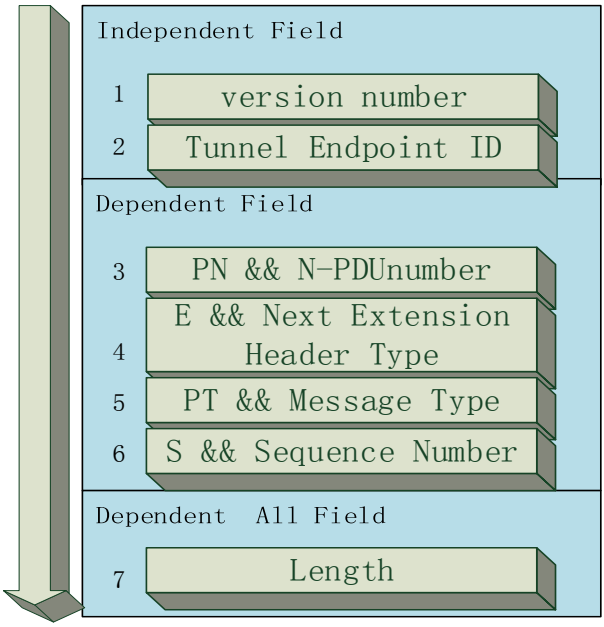

Figure 3. The layering strategy of GTP protocol

As shown as figure 2, in GTP header, version number, TEID can be divided into different independent level. Related Fields include protocol type and message type, field E and the next extension header type, field S and the sequence number, field PN and N-PDU number. The field length is related with all the field of GTP header, so we put it in a single level.

3.2 Policy of Priority

Protocol field values generally have specified range, and the protocol decode data in this range. There are may be many methods to dispose the message when it's value beyond the range. The most method used is discarding. The method of handling error message is usually to be attacked by hackers. According to the specified range of values, we can construct the abnormal data to avoid more data made by fuzz to be abandoned. So we use a more efficient way of setting priorities to detect every layer divided in previous section.

Specific strategies are as follows:

1) For reasonable data, we mainly test the value of the data within the range, especially values in the boundary of range. For example, if the rang is 1-100, we set the value 1 or 100 to test. Many protocol have a relatively safe method in dealing with the issue of the value of the edge value, and this is easily to lead to issue of pointers cross-border and make system collapse.

2) For unreasonable data, firstly, we detect whether those data can be accepted by protocol. If not, it is unnecessary to test other values beyond the range. If the abnormal data can be accepted, we can continue our test by exhausting random.

We design the processes of fuzz test aimed to GTP protocol :

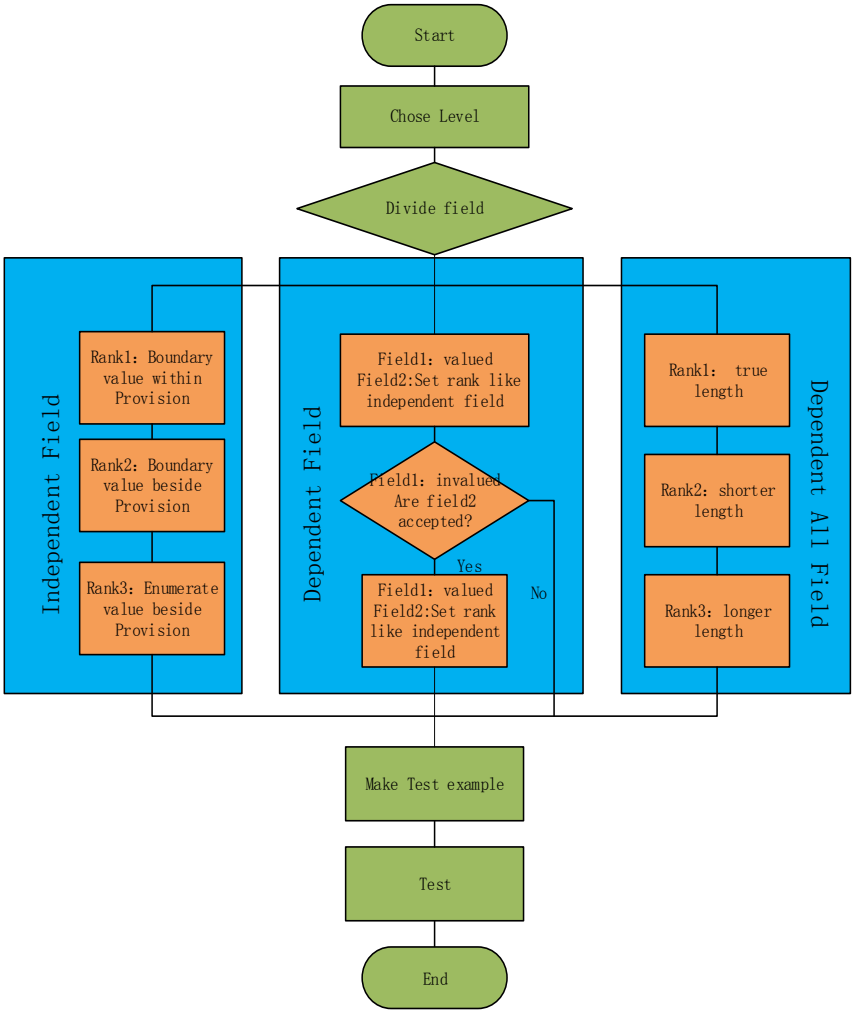

Figure 4. Flow diagram of fuzz test to GTP

4 Experiment

4.1 Experimental Platform

In this paper, we use NS-3 simulation environment to experiment for fuzzing method putted in the third paragraph. We will give a brief introduction of NS-3 simulation environment at first.

NS-3 is network simulator driven by discrete event, mainly used in research and education, which is designed to meet the needs of the academic and teaching. NS-3 project is a fully open source development project.

The development of the LTE module for ns-3 was carried out during the Google Summer of Code 2010. The module is built completely in C++. It comprises 89 classes and approximately 9000 lines of code. The module has been merged into ns-3.10.NS-3 including some modules, for example, network mudule， WiFi module，wimax module, LTE module and so on.

We build a custom LTE environment to test the GTP protocol based the LTE template in NS-3. Specific custom protocol stacks as follows:

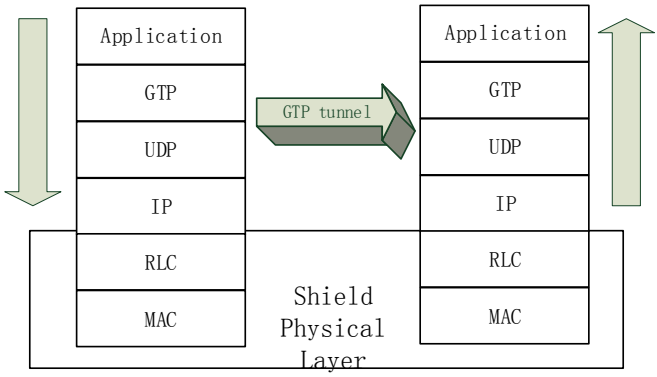

Figure 5. Custom stack of GTP

Construction of custom small-scale LTE network as follows:

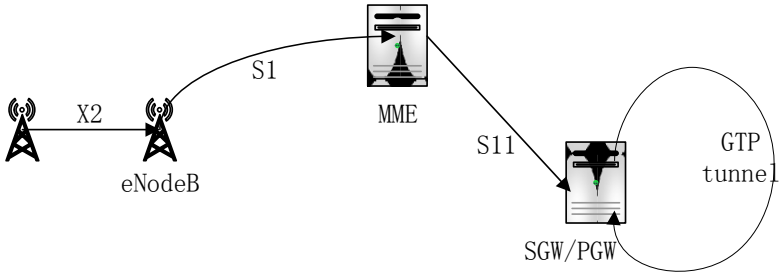

Figure 6. Custom small LTE network

As shown above, in order to simplify the network structure and avoid unnecessary costs, we combine SGW and PGW together using one node. GTP tunnels is implemented in application class nodes.

4.2 Result of Experiment

When data in custom scene is normal, the result is as follows:

```
kouzi@kouzi-Lenovo:~/ns3/ns-allinone-3.25/ns-3.25$ ./waf --run lena-simple-epc
Waf: Entering directory `/home/kouzi/ns3/ns-allinone-3.25/ns-3.25/build'
Waf: Leaving directory `/home/kouzi/ns3/ns-allinone-3.25/ns-3.25/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (2.030s)
```

When we alter the data randomly of GTP message made by the new method of fuzzing, and then re-compile and run the virtual environment, the result is as following:

```
kouzi@kouzi-Lenovo:~/ns3/ns-allinone-3.25/ns-3.25$ ./waf --run lena-simple-epc
Waf: Entering directory `/home/kouzi/ns3/ns-allinone-3.25/ns-3.25/build'
Waf: Leaving directory `/home/kouzi/ns3/ns-allinone-3.25/ns-3.25/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (2.039s)
assert failed. cond="it != m_teidRbidMap.end ()", file=../src/lte/model/epc-enb-application.cc, line=281
terminate called without an active exception
```

After successful compiling environment, the custom environment failed to link. As unreasonable value set in the fields, the virtual environment can't continue to go smoothly after starting up normal.

4.3 Vulnerability Analysis

In the above case, the altered value is field protocol type whose value is 1. In 3GPP protocol specification, value 1 of protocol type mean the message is echo message. GTP protocol use echo message to maintain connectivity of the GTP tunnel, at the same time, the echo message has the effect of resetting the tunnel connection. IE (information element) of echo message can carry normal node information, as well as, can take along information of reset connection between two nodes. P-GW in the other hand of tunnel will reset the count value of GTP tunnel connection according the message and delete local content relevant context after receiving echo request with the recovery information element.

Therefore, in the echo message of GTP protocol, setting the IE as recovery information can make the GTP session flow cut off.

5 Conclusion

In this paper, we put forward a new method of fuzz test anti LTE protocol based on the LTE protocol layering strategy. This method improve fuzzing in blindness and randomness. And we test this method in NS-3 simulation environment by detecting the GTP protocol of LTE. It proved the method is feasible and efficient relatively in detecting LTE protocol's vulnerabilities.

Acknowledgment

This work was supported by National Natural Science Foundation of China (No. 61272493 , 61502536 , U1536122).

References

1.3GPP. TS.33.401. V.12.9.0-2013. 3GPP System Architecture Evolution (SAE): Security Architecture (Release12)

2. 3GPP TS 129.060. General packet radio service (GPRS): GPRS tunneling Protocol (GTP) across the Gn and GP interface[s].2005

3. MurphyG. Whltehouse0. Attacks and COUntS measures in 2.5 and 3G cellular IP networks[R]. Cambride MA USA:@stake.Ine.,2004
4. Bavosa. A GPRS security threats and solution recommendations[R]. Sunnyvale CA USA: Juniper Network Inc.,2004.
5. 3GPP TSG-SA2, Security analysis for tunnel establishment[s], Nortel Networks July,2003.
6. Piro G, Baldo N, Miozzo M. An LTE module for the ns-3 network simulator[C]// Proceedings of the 4th International ICST Conference on Simulation Tools and TechniquesICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011:415- 22.
7. Wang T, Wei T, Gu G, et al. Checksum-Aware Fuzzing Combined with Dynamic Taint Analysis and Symbolic Execution[J]. Acm Transactions on Information & System Security, 2011, 14(2):613-613.
8. Cheng H F, Zhang Y Q. Bluetooth OBEX Vulnerability Discovery Technique Based on Fuzzing[J]. Computer Engineering, 2008, 34(19):151-153.
9. Gtinter Schafer, Research Challenges in Security for Next Generation Mobile Networks, Workshop on Pioneering Advanced Mobile Privacy and Security(PAMPAS), Royal Holloway University of London, Egham, Surrey, United Kingdom. September 2002.
10. Andrei Broder, Michael Mitzenmacher. Network Applications of Bloom Filters: A Survey. Internet Math. Volume 1, Number 4 (2003), pp.485-509.
11. H.N.Hung, Y.B.Lin, “Connection failure detection mechanism of UMTS charging protocol,” IEEE Transactions on Wireless Communication, vol.5, NO.5, pp.1180-1186, 2006
12. Liu Li-Fang, Huo Hong-Wei, Wang Bao-Shu. PHGA-COFFEE:Aligning multiple sequences by parallel hybrid genetic algorithm. Chinese Journal of Computers, 2006, 29(5): 727-733(in Chinese)
13.Liu Qi-Xu, Zhang Yu-Qing. TFTP vulnerability exploiting technique based on Fuzzing. Computer Enfineering, 2007, 33(20):142-144(in Chinese)
14. Makam P. Security vulnerabilities in GPRS networks[R]. Hyder-abad: India Wipro Technologies, 2006.