Formal specification of the S1AP protocol

PRASANTH PRAHLADAN, TAEHO KIM, and BOR-YUH EVAN CHANG, University of Colorado Boulder

The architecture and design of the cellular-communication-network is fast evolving to handle the needs of IoT and 5G communication. With the increase in scale, comes the need to update the different protocols used to control-coordinate decision making in the control-plane. While the protocols get updated, the implementations of the protocols need to be evaluated for compliance with older versions, for the sake of backward compatibility. Current methodology of testing doesn't help to address this challenge. There has been an interest in the community to adopt formal methods to specify and to ensure correct implementations of protocols within the software and internet-network domain. There hasn't been any prior work that does this within the cellular domain. We explore one method of compositional testing of S1AP protocol, which is used in the control-plane of cellular communication infrastructure. Our contribution is to transfer the tools and techniques that's being adopted for internet-protocols to the domain of control-plane protocols within cellular communications with a particular focus on the interface between the Radio part and the Core part of the network.

Additional Key Words and Phrases: S1AP, EPC, IVy

ACM Reference Format:
Prasanth Prahladan, Taeho Kim, and Bor-Yuh Evan Chang. 2020. Formal specification of the S1AP protocol. CSCI 5535, Spring 2020 (April 2020), 8 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

The 5G cellular network deployment, with the promise of enabling an internet-of-things and high-bandwidth, low-latency-applications, is an endeavour that has been the target of industrial players in the last few years. The promises of the 5G newtork architecture arises primarily from the follpwing enhancements over the 4G LTE architecture: (1) improvements in the physical and application layer technologies enabling large number of devices with higher bandwidth provisions, (2) enhanced security features introduced into the protocol stack. The 5G standard proposes a complex architecture of subsystems that interact with each other at different layers of the communication stack, while using multiple sub-procotols between entitites.

The cellular network infrastructure may be viewed as a large-scale wireless with a wired backend that is designed to support mobile data and voice services. Communication and messaging between the entities may be surmised under two layers of its design-abstraction called control-plane and data-plane. The control plane protocols form a significant part of its design, as it provides complex signalling functions, which makes it quite different from the network protocols that enable the internet. They follow the layered protocol architecture and run at both the network infrastructure and the end device. The cellular network control-plane, consists of a number of critical procedures which are leveraged by the primary cellular services like paging, voice-call, SMS, data and billing. Incorrect implementations of the protcols can have adverse consequences to these services.

Authors' address: Prasanth Prahladan, prasanth.prahladan@colorado.edu; Taeho Kim, taeho.kim@colorado.edu; Bor-Yuh Evan Chang, bec@colorado.edu, University of Colorado Boulder, 1111 Engineering Drive, Boulder, Colorado, 80309.

Accordingly, related protocols are frequently updated, and new protocols frequently appear. *User equipment (UE)*, such as a smartphone or a tablet, connects to an *evolved universal terrestrial radio access network (E-UTRAN) node B (eNB)*, which performs radio resource management to UE, to use cellular communication networks. In addition, when the UE is connected to networks, a *mobility management entity (MME)* must be connected to the eNB to manage this connection, as shown in Figure 1. MME provides evolved packet system (EPS) mobility management (EMM) and EPS session management (ESM) functions to the UE through *non-access stratum (NAS)* signaling. A NAS is a functional layer in the wireless telecom protocol stacks between the core network and UE. This layer is used to manage the establishment of communication sessions and for maintaining continuous communications with the UE as it moves.

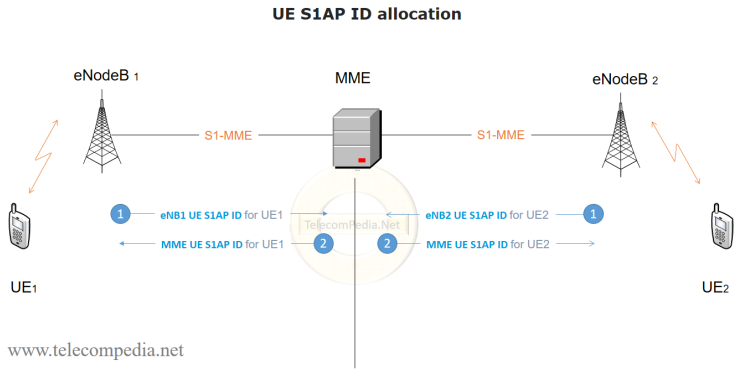

Fig. 1. UE S1AP ID allocation (will be changed)

In this work, we examine the protocol-specific interactions between critical components over the control plane, with an intention of identifying problems/mis-compliance of entities to the protocol specifications. Among them, we focus on the interface between the eNB and the Core-Network. To be precise, we intend to address the following concrete research question:

Is it possible to formally verify the correctness of a subset of the critical procedures handled by the control plane interactions between the evolved packet core (EPC) and the MME (cellular core network)?

There are two key challenges to this endeavour - (1) compared to the internet, cellular networks are still closed systems i.e. signalling exchanges between devices and base-stations, or between the base-stations and the core-network are not readily accessible during normal operations, (2) patterns of inter-protocol communication over the control-plane are more diverse and complex, in comparison to their counterparts over the internet. Protocol interactions are visible not just at the inter-layer interfaces, as seen over the internet, but also in cross-domain (e.g data and carrier grade voice services require signalling for circuit-switching and packet-switching in these networks), and cross-system scenarios (due to heterogenous deployment strategies inter-system switching between 3G, 4G and 5G systems need to be facilitated).

Instead of translating the 3GPP protocol specification for the *S1-Application Protocol (S1AP)* and the NAS protocol, we intend to demonstrate the methodology for specifying on-the-wire messages that are exchanged between the entities while interacting over a stateful-protocol. We shall focus on implementing one of the primary procedures facilitated by the EPC, the UE-Attach Procedure.

In this procedure, the EPC facilities the connection of a UE to the cellular network, to permit data-transmission over IP.

We intend to extend the application of the formal verification of Internet-protocols [5] to the domain of cellular communication networks. We are using the IVy tool [4] for providing a formal-specification of the S1AP/NAS protocol. This approach has been successfully used to verify a complex stateful internet-protocol QUIC, which runs as an application-layer protocol, while using UDP to send/transmit messages over the network. This use case is very similar to the domain of cellular-communication networks, where the control-plane protocols are executed as application-layer protocols, executing over an underlying stream control transmission protocol (SCTP) as a transport-layer protocol, which is used to communicate messages on-the-wire. In addition, we test the MME component (server) of the core network while generating test-messages that shall be sent by the eNB (client, radio-link).

We plan to demonstrate it by encoding the specification of the protocol in the IVy tool, and to adopt its automated-test-message generation process to evaluate some open-source implementations of cellular-core-network (e.g., OpenEPC [1]). The measure of success would be the following:

- Implementation of SCTP transport protocol within IVy, to facilitate message transmission between the IVy system and the EPC-implementation (server). The functionality shall be evaluated by the ability to transmit "S1-Setup Request Message" and also to receive "S1-Setup Response" message.
- Specification of subset of S1AP/NAS protocol that enables the "UE-Attach Procedure". This shall be evaluated by a successful simulation of a UE-Attach by an arbitrary UE with the EPC.

2 OVERVIEW

2.1 LTE Preliminaries

In the following subsection we shall cover some aspects of the cellular network system, with an emphasis on the sub-systems that we shall be targeting our testing procedure on.

2.1.1 LTE Network Architecture. As mentioned earlier, the LTE network architecture is broadly constituted of three components: (1) the cellular device (UE), (2) the radio access network (base station (BS) and radio channels), (3) the EPC or the core-network (CN and wired communication channels between the BS and the core).

(1) **User Equipment (UE)**: The end-user communication device which is equipped with a Universal Subscriber Identity Module (SIM) card, serves as a terminal device. The SIM contains unique identification information for each subscriber, of which the following two parameters are vital: *(1) the international mobile subscriber identity (IMSI) number* and *(2) the international mobile equipment identity (IMEI) number*, apart from the associated cryptographic keys that are required to ensure security and privacy protection of various interaction-procedures with the network.

(2) **Base Station(BS)**: The cellular radio access network partitions the geographical space into hexagonal cells to cater to the needs of subscribers within the region. Each geographical cell is serviced by a single BS, located at its 'center', that acts as an intermediary connecting the geographically dispersed subscribers with the CN. The E-UTRAN refers to the network between an eNBs(Base-stations) and the UEs.

(3) **Core-Network(CN)**: The CN or EPC consists of the following primary entities: *(1) Mobility Management Entity (MME)*: The MME serves as the primary interface between the BS and the CN. It manages the primary procedures of UE attach, UE detach, paging, etc, apart from the vital role of handling the mobility requirements of the UE. *(2) Home Subscriber Server*

(HSS): The HSS component of a service-provider maintains UE identities and subscription details, apart from the cryptographic keys and information required for authentication of the entities. *(3) Gateways*: Once the control-signalling has established the authenticity of the UE, verified their subscription and established a connection with the core-network, the Gateways handle the communication of data between the UE and the internet.

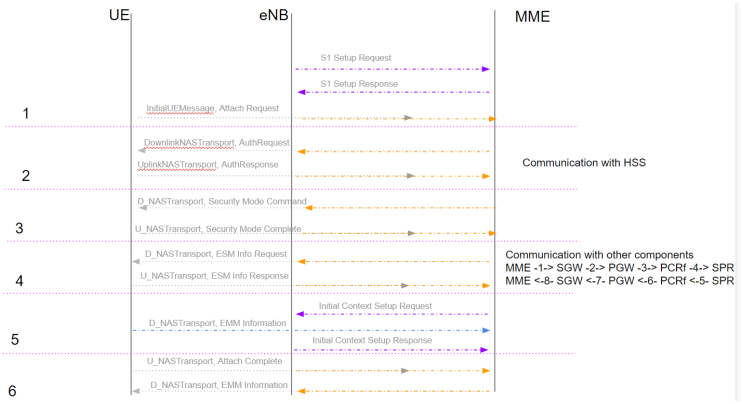

Fig. 2. Initial UE Attach procedure (will be changed)

2.1.2 Initial UE Attach Procedure. Of the set of vital control-procedures handled by the MME, we shall be focusing on the functionality and implementation of the Initial UE Attach Procedure. When a UE wants to connect to the EPC, which is automatically initiated by the device often when the device has been rebooted, it scans the radio waves to determine the eNB(base station) with the strongest signal power and attempts to establish a connection with it. The Initial UE Attach procedure, as shown in Figure 2, establishes a connection with the BS and triggers a subsequent connection to the CN by following the subsequent four sequence of operations:

(1) **Identification**: The UE sends an Attach Request Message to the MME via the BS, by providing self-identifying information of its IMSI/IMEI and security capabilities.
(2) **Authentication**: On reception of the Attach Request by the BS, the MME forwards the request to the HSS to obtain an authentication of the UE and generates an authentication challenge for the UE via an Authentication Request Message, so that it may authenticate the legitimacy of the MME it has established connection with. The UE uses its master cryptographic key to authenticate the MME and responds with an Authentication Response Message to the MME on success. If the authentication step is successful, the entities progress into the next stage of negotiating the security algorithm to establish a secure channel of connection between them.
(3) **Security Algorithm Negotiation**: From the security capabilities available in the UE, as communicated via the Attach Request message, the MME selects a security algorithm pairs (encryption and integrity) and communicates its choice to the UE along with the Message Authentication Code (MAC). Once the UE verifies the MAC, the UE and the MME establish a shared security context for protecting the secrecy and integrity of the messages transmitted over the channel in future.
(4) **Secure Temporary Identifier Exchange**: The MME then sends an encrypted and integrity-protected Attach Accept Message, which includes a temporary identity called Globally Unique Temporary Identity (GUTI) that shall be used as an identifier-pseudonym for the

UE. This is introduced to reduce the exposure of sensitive information like the IMSI/IMEI to potential eavesdropping and security-vulnerabilities in the system. The UE concludes the Attach Procedure, by transmitting an Attach Complete Message to the MME followed, by the establishment of a security context between the BS and the UE.

2.2 Specification Methodology

In this section, we present the specification methodology using examples. To specify the S1AP control-plane messaging protocol between the BS and the MME, we shall use an abstract machine that monitors the protocol events. Conceptually, this machine observes the sequences of message interactions between two communicating entities(tester and target) by observing the protocol-events triggered by the messages on the wire at the interface between the tester and the network. The network is not explicitly modelled in this approach, though there's an assumption that the network may delay or drop or re-order messages, but it shall not create new messages. When an event occurs, the abstract machine (which isn't finite state by design) consults its state-representation to determine whether the event conforms to the protocol specification. If so, it updates its state to account for the event, else, it may either marks the occurrence of an illegal event, or ignore it.

The protocol specification and the monitor are encoded in a language called IVy [4]. In IVy, the events associated with message-packets are modelled by an *action*. Constraints may be imposed upon the state and the parameters upon an event, by defining some conditions to be satisfied as a pre-condition or a post-condition. If these constraints are satisfied, the event is legal according to the protocol. Else, the events may be considered to be in violation of the protocol.

The protocol state is represented by a collection of functions or relations. The choice between the functional-representation or the relational-representation is based on how the state-information shall be queried in downstream actions. The two forms of representation are conceptually equivalent. Any element of the relational-set represents a predicate-abstraction of the actual event.

The properties of the protocol can thus be verified by the communication of abstracted messages between the entities, or it may be used to generate actual messages on the wire, using a *shim* that generates actual network-protocol encoded messages to be sent on the wire, while also capturing the packet events received by the entity by tracing packets on the wire. While doing so, the shim triggers the specific protocol-events that get recorded as state-updates within the monitor. It is important to note that the specification is an executable monitor that observes the behaviour of all protocol nodes, in contrast to being an abstract implementation of the protocol entities.

An important capability provided by the IVy tool, is the capacity to obtain a *generator* that produces random sequences of events that conform to the specification. The generator uses an SMT solver to randomly select a valuation of parameters that satisfy the 'require' constraints encoded in the pre-condition of the selected action. Consequently, the generator helps to generate a random sequence of packet-events in compliance with the protocol-specification that is sent to the test-target.

The S1AP application protocol messages are communicated between the eNB and the MME, based on a lower-layer transport protocol based in SCTP(in comparison to UDP or TCP for general data packets over the internet). We create a shim that connects the abstract packet events to actual SCTP packets on the wire. The *send* and *receive* events are encoded within the shim. The shim is thus an ad-hoc mechanism that connects abstract protocol events of the specification to real-events in the system. Thus, if the socket or other implementation details of the system gets modified, the shim alone may be modified, without altering the protocol specification. This shim, thus helps us to connect the abstract protocol specification, to the physical resources such as the operating system, the networks and compilers to test real-world implementations. It may also be modified

to conduct experiments in a simulated environment based on NS3, or be used alone to conduct a formal analysis of the protocol in an abstract theoretically-sound manner.

The specification may be encoded at a global level, accounting for all the roles that different protocol-entities may play, or at a role-based level, where only the specifications for a single-role are encoded. In the latter case, one needs to effectively assume that the role-being-modelled generates events that are correct according to the protocol, since it's the task of the generator to create the corresponding messages. The task of the verification process is to "guarantee" the correctness of events generated by counterparts of the tester i.e roles played by the be multiple(or single) test targets as required by the protocol design. This framework of Assume/Guarantee specification based testing has an important formal property: if the composition of roles ever violates the specification, then there must exist a failing assume/guarantee test for one of the many roles within the protocol.

2.3 Specification Examples

This section shall be updated after we've had some progress in using the tool to model the messsages.

3 CONTRIBUTION 1

...

4 CONTRIBUTION 2

...

5 EMPIRICAL EVALUATION

...

6 RELATED WORK

The primary research exercise that we undertake is to determine an good methodology for providing a formal-specification for the control-plane protocols used in the cellular communication infrastrucuture. When we think about task for formalizing a specification for a protocol, there are two main use-cases: (1) to use the specification as an input for verifying that a particular model of its implementation satisfies some desriable properties, and (2) to enable the development of correct by construction implementations of the protocol. In our work, we explore the domain of specification as a means to conduct testing of an implementation, by facilitating a mechanism of automated generation of test-messages that shall be used to test the communication interface between two entities.

Most related researches have focused on the issue of formal-verification of correctness and security properties that are provided by the authentication protocols used in this domain. They have attempted a formal methodology of testing a communication protocol. One research team developed a formal specification of the wire protocol [5]. They used the specification to generate automated randomized testers for implementation of QUIC (Quick UDP Internet Connections) that is an Internet secure transport protocol. The testers take one role of QUIC protocol: either interacting with the other role to generate full protocol executions or verifying that the implementations conform to the formal specification. Considering the process of evaluating strict compliance with standards in various communication environments, it is necessary to test implementations in adversarial environments. In addition, they developed and released Ivy Tool [4] to evaluate QUIC protocol. Although the protocols are different, the considerations and suggested approaches covered in this study can also be used to solve our problems. Another interesting research project, that

aligns with our endeavour, is the Project Everest [1], which attempts to create a formally verified stack to guarantee verified low-level implementations of the HTTPS stack. Especially, they include cryptographic algorithms and develop new implementations of existing and new protocol standards while formally proving that their implementations provide a secure-channel abstraction between the communicating endpoints.

There are also several tasks related to the verification of cellular network protocols. CNetVerifier [6] is a tool to analyze the inter-layer, inter-domain and inter-system protocol interactions within the control-plane of cellular communication networks. The tool adopts a model-checking methodology within its two-phase protocol-diagnosis strategy to detect issues arising from (1) design problems within the protocol standards specification, and (2) operational mistakes of the service-provider. The verification strategy, however, is user-centric, i.e. the properties that are verified is related to the user-entities and cannot be used to examine interactions between the BS and CN, which would be of interest to improve the operational needs of the carrier/service-provider. In this work, the protocol is modelled as two interacting FSMs, with one representing the UE, and the other representing the network entity(BS, MME, etc), within the model-checking framework, SPIN. The measurement based verification is handled at the UE level.

LTEInspector [2] which employs a property-driven adversarial model-based testing philosophy. LTEInspectortakes the relevant 4G LTE abstract model and a desired property (ϕ), and tries to find aviolation of ϕ in the model. The tool checks for the following properties - authenticity, availability, integrity, and secrecy , all from the perspective of the end-user/customer. The model they develop comprises of as synchronous communicating finite state machines which abstract away the functionality while ignoring low-level implementation details. They adopt an instance of the parameterized system verification problem (i.e., parameterized by the number of protocol participants).Their instantiation of the LTEInspector framework, however,adopts the following constraints: (1) They consider only a single layer of the protocol stack in isolation; (2) For the sake of scalability, they only model packet type and do not model critical data or packet payload, missing out on interesting data-/payload-dependent protocol behavior; (3) Their adversary instantiation cannot handle protocols spanning across different layers of the stack.

In the most recent work, 5GReasoner [3], the authors adopt a modelling procedure which models about five different control-plane procedures, with a FSM modelling each layer of the stack. The state-machines corresponding to different entities communicte via a public, adversary-controlled channel, where the adversary is also modelled as a FSM. The NAS layer protocol packets which constitute data-packets of the RRC layer protocols, is modelled using a behaviour-specific predicate-abstraction methodology, in which they do not directly model the data, but only predicates-over the data which are essential to verify the specific properties they are examining. These models are implemented in two infinite-state model-checkers and a cryptographic protocol verifier. They have verified about 187 properties which were extracted either from the standards, or were specified based on domain-knowledge.

ACKNOWLEDGMENTS

TBD

REFERENCES

[1] M. Corici, F. Gouveia, T. Magedanz, and D. Vingarzan. 2010. Openepc: A technical infrastructure for early prototyping of ngmn testbeds. In *International Conference on Testbeds and Research Infrastructures*. Springer, 166–175.

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

[1] https://www.microsoft.com/en-us/research/project/project-everest-verified-secure-implementations-https-ecosystem/

[3] S. R. Hussain, M. Echeverria, I. Karim, O. Chowdhury, and E. Bertino. 2019. 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*. 669–684.

[4] K. L. McMillan and L. D. Zuck. 2019. Compositional Testing of Internet Protocols. In *2019 IEEE Cybersecurity Development (SecDev)*. IEEE, 161–174. https://ieeexplore.ieee.org/abstract/document/8901577.

[5] K. L. McMillan and L. D. Zuck. 2019. Formal specification and testing of QUIC. In *Proceedings of the ACM Special Interest Group on Data Communication*. 227–240. https://dl.acm.org/doi/abs/10.1145/3341302.3342087.

[6] G.-H. Tu, Y. Li, C. Peng, C.-Y. Li, H. Wang, and S. Lu. 2014. Control-plane protocol interactions in cellular networks. *ACM SIGCOMM Computer Communication Review* 44, 4 (2014), 223–234.