This paper has been accepted for publication on IEEE International Conference on Computer Communications (INFOCOM) 2021.

SteaLTE: Private 5G Cellular Connectivity as a Service with Full-stack Wireless Steganography

Leonardo Bonati,* Salvatore D'Oro,* Francesco Restuccia,*† Stefano Basagni,* Tommaso Melodia*
* Institute for the Wireless Internet of Things, Northeastern University, Boston, MA, U.S.A.
† Roux Institute, Northeastern University, Portland, ME, U.S.A.

***Abstract*—Fifth-generation (5G) systems will extensively employ radio access network (RAN) softwarization. This key innovation enables the instantiation of "virtual cellular networks" running on different *slices* of the shared physical infrastructure. In this paper, we propose the concept of *Private Cellular Connectivity as a Service* (PCCaaS), where infrastructure providers deploy *covert network slices* known only to a subset of users. We then present *SteaLTE* as the first realization of a PCCaaS-enabling system for cellular networks. At its core, *SteaLTE* utilizes *wireless steganography* to disguise data as noise to adversarial receivers. Differently from previous work, however, it takes a *full-stack approach* to steganography, contributing an LTE-compliant steganographic protocol stack for PCCaaS-based communications, and packet schedulers and operations to embed *covert data streams* on top of traditional cellular traffic (*primary traffic*). *SteaLTE* balances undetectability and performance by mimicking channel impairments so that covert data waveforms are almost indistinguishable from noise. We evaluate the performance of *SteaLTE* on an indoor LTE-compliant testbed under different traffic profiles, distance and mobility patterns. We further test it on the outdoor PAWR POWDER platform over long-range cellular links. Results show that in most experiments *SteaLTE* imposes little loss of primary traffic throughput in presence of covert data transmissions ($< 6\%$), making it suitable for undetectable PCCaaS networking.**

***Index Terms*—Steganography, 5G, Private Cellular Connectivity as a Service, Undetectability.**

I. Introduction

The *softwarization* of the Radio Access Network (RAN) is being heralded as the core of fifth generation (5G) cellular networks [1–7]. Enabling virtualization technologies, softwarization will allow Infrastructure Providers (IPs) to create *virtual networks* on top of their physical infrastructure, each assigned to a different infrastructure *slice* [8–10]. This fundamental innovation will concretely realize the long-standing vision of *Cellular Connectivity as a Service (CCaaS)*, where the IP assigns physical resources (e.g., spectrum, power, base stations, etc.) to each Mobile Virtual Network Operator (MVNO) according to their requirements [11, 12]. CCaaS is envisioned to provide unparalleled levels of Quality of Experience (QoE) to mobile users, as well as usher in new business opportunities between IPs and MVNOs [13, 14].

In this paper we leverage RAN softwarization and network slicing to concretely realize *Private Cellular Connectivity as a Service (PCCaaS)*, pushing the CCaaS innovation to the realm of *private* networking. Through PCCaaS the IPs can instantiate and deploy *private network slices* sharing the virtualized infrastructure with other (public) slices. In this paper we use the word *private* to identify slices whose existence is known only to selected users that can exchange sensitive data embedding it *covertly and undetectably* into *primary traffic*, used as decoy.

The opportunities and applications of PCCaaS are multifold. For instance, with PCCaaS law enforcement agencies could leverage the ubiquitous connectivity offered by extant cellular infrastructure and use private slices to establish undetectable communications with undercover agents in the field. Similarly, law enforcement could deploy tiny Internet of Things (IoT) devices as "bugs," collecting audio and video content and communicating it covertly. Such devices would pose as regular IoT sensors and conceal sensitive covert information on top of innocuous primary traffic, e.g., temperature readings. An example of PCCaaS is shown in Fig. 1.

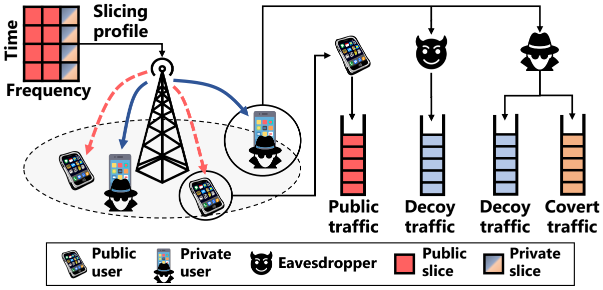


Fig. 1: Private Cellular Connectivity as a Service (PCCaaS).

The IP instantiates three slices whose profile is communicated to the cellular base station. Two slices are public (in red). These slices are for public users of the infrastructures (e.g., cellular subscribers). The third slice is private. In this slice, private users exchange data that is of non-sensitive nature (decoy traffic, in blue). This is their primary traffic. They also exchange *covert* traffic (in orange), which is hidden into the primary data. The challenge of PCCaaS is that of fooling a malicious eavesdropper to believe that the private users are only exchanging decoy traffic, namely, in allowing the eavesdropper to capture only their primary traffic.

Clearly, no form of effective data encryption is the solution to realizing the vision of PCCaaS. First of all, not all devices have the necessary resources to support the execution of power-hungry and computationally complex encryption algorithms. The IoT scenario described above is a typical example. Fur-

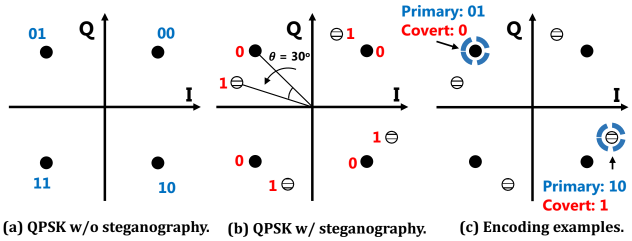
E-mail: {l.bonati, s.doro, frestuc, s.basagni, t.melodia}@northeastern.edu

This work was supported by the U.S. Office of Naval Research under Grants N00014-19-1-2409 and N00014-20-1-2132.

thermore, encrypted traffic is still subject to jamming: An adversarial user that is capable to detect the transmission of sensitive information could prevent its intended recipient to receive it. The key question is therefore how to ensure that communication of sensitive data is not only secure, but also *undetectable*, independently of encryption.

One of the key challenges in realizing PCCaaS is that data transmitted over wireless channels cannot be easily hidden. To address this problem, *wireless steganography* directly operates on Radio Frequency (RF) waveforms by applying "hand-crafted" tiny displacements to the I/Q symbols being transmitted, also known as *primary* symbols [15–22]. While a steganographic receiver (the private user of Fig. 1) can decode the covert information by translating the "dirty" I/Q symbols to a corresponding covert bit sequence, public users would be able to decode primary symbols only.

Fig. 2 illustrates a practical example of wireless steganography where covert data are embedded into a QPSK-modulated signal.



(a) QPSK w/o steganography. (b) QPSK w/ steganography. (c) Encoding examples.

Fig. 2: Wireless steganography over a QPSK modulation.

Specifically, Fig. 2a shows the set of QPSK symbols used to form the primary message. Let us assume that the transmitter sends the primary symbol "01" to deceive adversaries, but at the same time it wants to embed a covert message in it through wireless steganography. To achieve this, the transmitter can rotate the phase of the I/Q symbols of Fig. 2a by an angle θ to send the covert bit "1" (Fig. 2b), while symbols with no rotations correspond to the bit "0" (Fig. 2c) [23]. (More sophisticated schemes are described in Section II-B.)

Despite recent advances, wireless steganography has not yet found widespread application in networking. We believe this is because existing approaches operate only at the physical layer, which is insufficient to make PCCaaS systems possible.

In this paper we leverage steganography as the core of a ready-to-use *full-stack* approach to PCCaaS-based networking. Our system, that for testing purposes has been realized on open-source LTE implementations, is called *SteaLTE* to indicate the *stealthy*, private nature of the networking it enables to satisfy PCCaaS requirements. *SteaLTE* achieves:

- *End-to-end Reliability and Security*. We design a *full-stack steganographic system* leveraging proven reliable data transfer techniques. These are integrated to a *steganographic mutual authentication* mechanism where legitimate parties authenticate each other before exchanging confidential information.
- *Adaptive Traffic Embedding*. Covert data need to be embedded over *primary* traffic, which is inherently varied and unpredictable. Clearly, a large covert data packet cannot be embedded on a small primary packet, or cannot be transmitted at all in the absence of primary traffic. This requires the process of embedding covert traffic to be *flexible enough* to deal with such unpredictability. To this purpose, *SteaLTE* features a covert packet generator component that creates and embeds covert packets that seamlessly adapt to primary data traffic, generating "dummy" primary traffic on-demand, if necessary.
- *Standard compliance*. To successfully operate over existing cellular networks, PCCaaS must adhere to standard protocol implementations of 4G/5G systems. *SteaLTE* has been designed to seamlessly integrate with cellular systems without disrupting primary communications or affecting their performance.
- *Undetectability*. A goal of PCCaaS is to make covert data communications undetectable, concealing them from eavesdroppers and jammers. To this aim, we design a stochastic steganography scheme that embeds covert transmissions by mimicking wireless channel noise. We show that *SteaLTE* reduces the Kolmogorov–Smirnov (K-S) distance from the "clean" (i.e., without covert data) distribution by 4.8x, improving undetectability with respect to previous solutions [24].

Our LTE-compliant prototype of *SteaLTE*—the first for PCCaaS-based cellular networking—has been evaluated through experiments over indoor and outdoor testbeds (including the POWDER platform from the PAWR program [25, 26]) on scenarios with varying parameters, including topology, traffic patterns, mobility and link ranges. Our results show that, overall, the *SteaLTE* covert throughput is comparable to the primary throughput, that it minimally affects primary transmissions imposing $< 6\%$ loss of primary throughput, and that, even in challenging outdoor settings, effectively delivers covert data on links up to 852 ft long.

The rest of paper is organized as follows. The *SteaLTE* system design is presented in Section II. Its prototype over LTE-compliant implementations is described in Section III. Section IV reports results from our testbed-based experimental evaluation of *SteaLTE*. A review of previous work on the topic is surveyed in Section V. Conclusions are drawn in Section VI.

II. SteaLTE Design

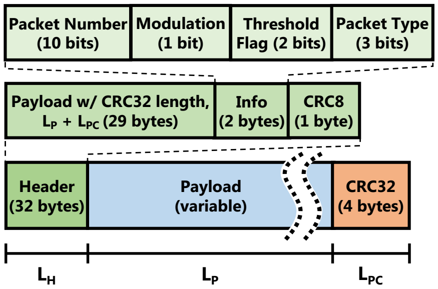
In this section we describe *SteaLTE*, providing details on its *covert communications* (Section II-A), *transmitter* and *receiver design* (sections II-B and II-C), and the mechanisms to enable *undetectable covert communications* (Section II-D).

A. Covert Communications: Formats and Operations

This section describes the packet format and the operations that allow *SteaLTE* to enable PCCaaS-based reliable and secure covert communications.

1) Packet format: The structure of *SteaLTE* covert packets is illustrated in Fig. 3a. Each packet consists of three elements: A *header*, a *payload* and the *Cyclic Redundancy Check (CRC)*.

The **header** consists of 32 bytes carrying information on how to decode a received covert packet. For packet detection and demodulation, the header is modulated through a fixed covert modulation known by the receiver. Its structure is as follows:



(a) Packet structure.

Type	Flag	Description
0	000	ACK
1	001	NACK
2	010	Data
3	011	Source / Destination, Packet Info
4	100	Challenge
5	101	Challenge Response
6	110	Authentication ACK
7	111	Reserved

(b) Packet types.

Fig. 3: *SteaLTE* covert packet structure and types.

(i) A 29-byte field with the length of the covert payload (L_P) and the CRC (L_{PC}); (ii) a 2-byte *info* field with information on how to demodulate the covert packet, and (iii) a 1-byte CRC8 field to detect errors on the header. The *info* field contains:

- *Packet Number:* 10 bits uniquely identifying a packet, also used to request packet retransmission (Section II-A2).
- *Modulation:* A bit flag indicating the modulation used to encode payload and CRC. *SteaLTE* chooses between two covert modulation schemes depending on the quality of the wireless channel. This field can be extended to account for additional covert modulation schemes (see also Section II-B1).
- *Threshold Flag:* 2 bits to instruct the receiver on how to demodulate and decode covert data. This field is paramount for our undetectability scheme (Section II-D).
- *Packet Type:* A 3-bit field to discern among data and control packets. The different packet types and their flags are shown in Fig. 3b. Packet types 0 and 1 are `ACK` and `NACK` control packets sent by the receiver to give feedback on the covert transmission (Section II-A2). Packets carrying covert data are of type 2. Packets of type 3 carry information on source and destination of covert packets and on the total number of packets in the current transmission. Each source and destination address is encoded by 5 bytes containing the corresponding Mobile Subscription Identification Number (MSIN) (i.e., the telephone number commonly used to identify mobile subscribers). Upon receiving an uplink covert transmission, the Base Station (BS) maps the destination MSIN to the corresponding International Mobile Subscriber Identity (IMSI), which uniquely identifies the User Equipment (UE). It then relays the covert message to the receiver via a downlink covert transmission or forwards it to the *SteaLTE* BS serving the receiver, as for regular voice traffic). Packets of type 4, 5 and 6 are used for the mutual authentication of covert transmitters and receivers (Section II-A3). Packet type 7 is reserved for future use.

Payload and CRC32. The variable-size packet payload carries sensitive user data to be transmitted covertly. This field adapts to the size of primary packets to improve the efficiency of covert communications (Section II-B1). To ease reception, the length of this field is included in the packet header (Fig. 3a). The packet ends with a 4-byte CRC32 field utilized for error detection and to ensure the integrity of covert transmissions.

2) Reliable covert communications: SteaLTE provides built-in reliability through standard reliable data transfer mechanisms. These include error detection, receiver-to-transmitter feedback (positive and negative acknowledgments), packet sequence numbers, timeouts, and retransmissions [27]. Error detection is performed through two Cyclic Redundancy Check (CRC) codes: A CRC8 code is used to protect the header of the packet and a CRC32 code for the packet payload (see Fig. 3a).

3) Mutual Authentication: SteaLTE implements a scheme for the mutual authentication of BSs and UEs through covert challenge/response operations. After standard cellular attachment procedures are completed, the BS sends a randomly-generated *challenge* to the UE using a type 4 packet (Fig. 3b). Upon receiving this packet the UE computes the Keyed-Hash Message Authentication Code (HMAC) from the BS challenge and key (which has been pre-shared), and sends the HMAC result as the *challenge response* (packet of type 5). After receiving the response from the UE, the BS compares it with the expected HMAC result. If the two match, the BS considers the UE *authenticated*. To notify the successful end of the UE authentication procedures, the BS sends an *authentication* `ACK` message to the UE (packet of type 6). If the challenge response is not received, the BS retransmits the challenge to the UE. After a certain number of unresponded attempts, or in case of erroneous response, the BS considers the UE *not authenticated* and will avoid any covert communication with it. When the UE receives the authentication `ACK` from the BS, it follows a similar procedure to *authenticate* the BS.

B. Transmitter Design

This section presents the main components of *SteaLTE* transmitter: The *Covert Packet Generator*, the *Covert Modulator*, and the *Covert Embedder*. In the reminder of the paper, *orange-colored* blocks with dashed lines denote system components of *SteaLTE*. All other colors identify standard cellular components that do not require hardware or software modifications.

Fig. 4 provides a high-level overview of the building blocks of a *SteaLTE* transmitter.

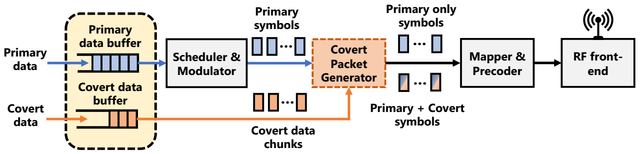


Fig. 4: High-level *SteaLTE* transmitter design.

Primary and covert data streams are separate and independent from one another. Primary data are processed through standard scheduling and signal processing procedures (e.g., modulation) of the primary system. These data result in a sequence of primary symbols that are fed to the *SteaLTE covert packet generator* (Section II-B1). After the covert symbols have been embedded in the primary symbols, they are mapped and precoded according to standard cellular procedures (Section III), and transmitted through the RF front-end.

1) Covert Packet Generator: This block reads covert data from the covert data buffer, and embeds it in the modulated primary symbols. This is achieved by executing the following three steps (see Fig. 5): (i) Verifying that there are enough primary symbols to embed a complete covert packet; (ii) generating covert symbols to be transmitted, and (iii) embedding them into primary ones.

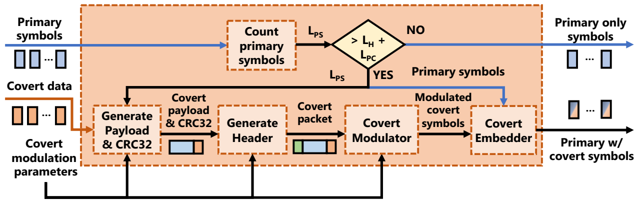


Fig. 5: Covert packet generator block overview.

The covert packet generator starts by verifying if the number of primary symbols L_{PS} is large enough to accommodate at least $L_{min} = 36$ bytes, which are required for the covert packet header ($L_H = 32$ bytes) and the CRC32 field ($L_{PC} = 4$ bytes). In the positive, it generates the covert packet payload and CRC32 field. The length of the payload and of the CRC32 are included in the packet header, as described in Section II-A1. The packet is then modulated through the *covert modulator* according to set *covert modulation parameters* (also in the header). Finally, the resulting covert modulated symbols are embedded in the primary symbols through the *covert embedder*. If $L_{PS} \leq L_{min}$ no covert data are embedded in the primary traffic. Note that the adaptive structure of the covert packets allows to embed variable size covert data on top of *time-varying* and *unpredictable* primary traffic. This feature makes *StealTE* transparent to primary traffic dynamics, thus enabling the integration of *SteaLTE* with any softwarized cellular system.

Covert Modulator. This block is in charge of encoding covert packets into covert symbols that can be embedded into primary transmissions (Fig. 5). Several approaches are possible for covert embedding of data through wireless steganography. Fig. 6 illustrates three examples of 2 covert bits to be added on top of a primary QPSK constellation.

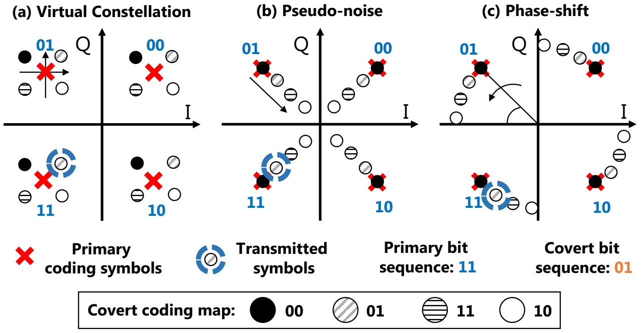
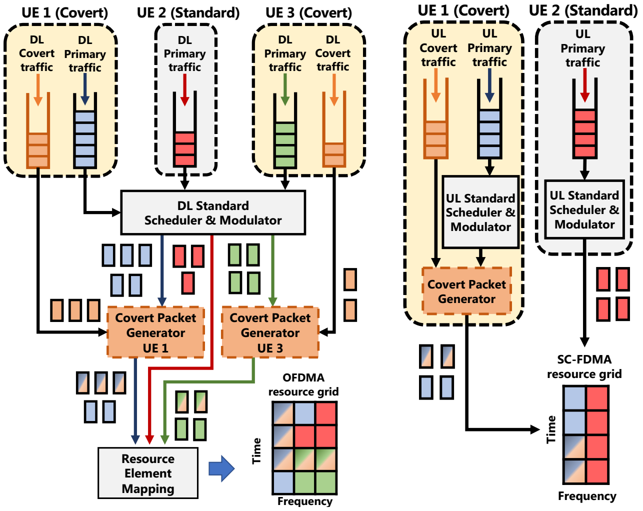


Fig. 6: Approaches to wireless steganography.

The first approach generates a "dirty" QPSK constellation around each primary symbol (Fig. 6a) mimicking a hierarchical constellation on top of the primary QPSK constellation [16]. The second one introduces a hierarchical Amplitude Shift Keying (ASK) modulation manipulating the amplitude of the primary symbols (Fig. 6b) such that different amplitude values encode different covert bit sequences [24]. The third approach modifies the phase offset of the primary symbols (Fig. 6c) in a way that each phase rotation encodes a specific bit sequence [23]. As *SteaLTE* is not tied to any specific steganographic procedure its covert modulator supports any of these approaches. In the following we assume that the covert modulator block implements the approach depicted in Fig. 6b [24], which we call M_C-ASK, where M_C is the number of symbols in the covert constellation. The advantages of this approach include that it is robust against phase rotations introduced by fading, that it supports high-order modulation schemes (for high covert data rates), and that it can be seamlessly integrated with OFDM systems such as those used in the latest generations of cellular networks. The covert modulator receives the covert packets together with the set of covert modulation parameters, which specify the modulation order, the corresponding coding map, and the packet type (Fig. 5). The coding map uniquely associates covert packets (i.e., bit sequences) to modulated covert symbols. Covert symbols are, then, embedded in primary symbols through the *covert embedder* block.

Covert Embedder. Once covert symbols have been generated, they are embedded by the covert embedder (Fig. 5). This procedure modulates the amplitude (and phase) of the primary symbols based on the covert symbols to embed [24]. The output is a sequence of primary symbols with embedded covert data. The symbols are then processed by mapping and precoding blocks and transmitted through the RF front-end (Fig. 4).

2) Downlink and Uplink Procedures: SteaLTE runs seamlessly on both downlink and uplink transmissions (Fig. 7), and does not depend on specific Medium Access Control (MAC) strategies (e.g., TDD/FDD, OFDMA/SC-FDMA).



(a) Downlink (DL) transmitter.

(b) Uplink (UL) transmitter.

Fig. 7: High-level *SteaLTE* downlink and uplink transmitter design.

Fig. 7a shows the downlink transmitter design at the BS. In this example, the BS is serving three subscribers: Two covert users (UE 1 and UE 3), and a standard user (UE 2). After scheduling the primary transmissions through typical cellular procedures, the BS generates the covert packet to embed on the primary traffic of UE 1 and UE 3 (Section II-B1). Then, the data for all users is mapped on the cellular resource grid (e.g., the OFDMA grid in case of LTE downlink), and transmitted.

The high-level uplink transmitter design is shown in Fig. 7b, where two users are connected to a *SteaLTE* BS: UE 1 (covert user), and UE 2 (standard user). After completing the mutual authentication procedures, UE 1 generates and embeds the covert packets in the primary uplink traffic to send to the BS. Then, it maps the primary uplink transmission with embedded covert data on the cellular resource grid (e.g., SC-FDMA grid in case of LTE networks). On the other hand, UE 2, which is not aware of the ongoing covert communications, schedules its uplink transmission according to standard cellular procedures.

C. Receiver Design

Fig. 8 shows the receiver design. Primary data are processed as per standard cellular procedures. Covert data follow a separate receive chain with two main components: The *Covert Packet Detector* and the *Covert Payload Demodulator*.

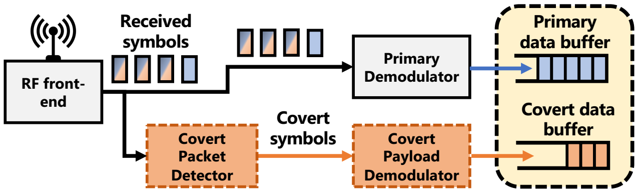


Fig. 8: High-level *SteaLTE* receiver design.

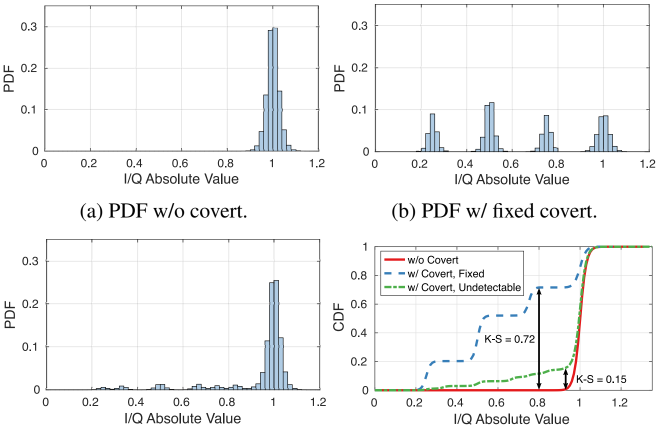
1) Covert Packet Detector: This block detects the presence of covert packets demodulating the covert packet header (first L_H bytes of the covert packet). As mentioned in Section II-A1, the covert header is modulated through a 2-ASK modulation. Thus, if the CRC8 check passes (Section II-A2), the receiver assumes a covert packet has been received.

Upon detecting a covert packet, the covert packet detector reads the length $L_P + L_{PC}$ of the covert payload and CRC32 fields, the packet number and the modulation parameters in the info field of the header (Fig. 3a). Finally, it extracts the symbols corresponding to the encoded $L_P + L_{PC}$ bytes of the covert packet, that will be demodulated by the *covert demodulator* block of Section II-C2.

2) Covert Demodulator: This block extracts the encoded covert information from each packet. As shown in Fig. 3a, covert modulation parameters necessary to demodulate covert packets, such as employed covert modulation, packet length and the packet type are specified in the header. This way, the demodulator block can reconstruct the decoding map and use it to demodulate the received symbols into covert data (e.g., bit sequence). Since all the covert modulation parameters are specified in the packet header, no further interaction is required between transmitter and receiver. Section II-D (below) shows that this approach also enables time-varying coding/decoding mappings that make covert transmissions undetectable and secure against eavesdroppers. Finally, the received CRC32 value is checked (Section II-A2). If the check passes, the data are saved, otherwise a retransmission will be requested.

D. Undetectable Covert Communications

Steganography is not immune from attacks. For instance, through steganalysis [28] an eavesdropper may analyze the statistical properties of captured I/Q samples and infer the presence of a covert slice. For example, let us consider the case of primary QPSK transmissions where *SteaLTE* embeds covert data through a 4-ASK covert modulation [24]. Fig. 9 shows the Probability Density Function (PDF) of the I/Q samples captured through the testbed described in Section IV-A in three different cases: Primary-only transmissions are shown in Fig. 9a; primary with *fixed*, i.e., detectable, 4-ASK covert transmissions in Fig. 9b, and primary with *SteaLTE undetectable* covert transmissions in Fig. 9c. We also show the Cumulative Distribution Function (CDF) of all cases in Fig. 9d. The Kolmogorov–Smirnov (K-S) distance is also shown to measure the similarity of the CDFs: The smaller the distance, the better.



(a) PDF w/o covert.

(b) PDF w/ fixed covert.

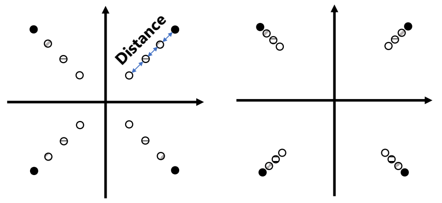
(c) PDF w/ undetectable covert.

(d) CDF and K-S distance.

Fig. 9: PDF and CDF of received I/Q samples.

Fig. 9a shows the PDF of the absolute value of the captured primary I/Q samples without any covert transmission. As expected, the PDF assumes a Gaussian distribution with mean 1 due to noise and fading. Fig. 9b shows the PDF when covert data are embedded through a 4-ASK covert modulation [24]. Comparing Fig. 9b with the primary-only case of Fig. 9a, we notice that the absolute value of the captured samples no longer exhibits a Gaussian PDF centered around 1, but multiple bell-shaped Gaussian curves centered at 0.25, 0.5, 0.75, and 1. This is also illustrated in Fig. 9d where the CDF of the I/Q samples resembles a step function with a K-S distance with the primary-only case equal to 0.72. Such statistical behavior is not surprising: This result is inherited from the 4-ASK covert scheme in Fig. 6b, whose operation results in 4 possible covert points per primary symbol with amplitude equal to 0.25, 0.5, 0.75, and 1. Steganalysis can easily identify such an abnormal statistical pattern, thus revealing the ongoing covert transmissions to the eavesdropper. For steganographic communications to be undetectable, they must statistically behave like primary ones, which is possible by reducing their K-S distance.

For this reason, *SteaLTE* implements a mechanism that mimics I/Q displacements introduced by channel noise by randomizing the covert embedding procedures. Rather than utilizing a *fixed distance* between covert symbols (as done in [24]), *SteaLTE randomly changes* this distance, providing the *first-of-its-kind undetectability mechanism*. This process is illustrated in Fig. 10, where we show 4 possible configurations of a 4-ASK covert constellation.



Flag	Distance
00	0.333
01	0.25
10	0.165
11	0.0825

(a) Flag = 01. (b) Flag = 11. (c) Thresholds.

Fig. 10: Examples of covert 4-ASK constellations with different distances and threshold flag values.

To decode covert ASK messages, the receiver must be aware of the distance between covert symbols. As a consequence, randomizing the transmitted covert constellation could potentially undermine the receiver's covert demodulation procedures. To overcome this problem, *SteaLTE* packets carry a *threshold flag* field (see Section II-A1) instructing the receiver on the covert constellation used by the transmitter. The value of this flag is changed on a per-packet basis, thus reducing the probability of successful steganalysis attacks. In our current implementation, this field consists of 2 bits encoding the 4 different distance configurations shown in Fig. 10c. However, it is straightforward to increase the size of this field to further enhance undetectability.

The effectiveness of our approach is depicted in Fig. 9c, where we show the PDF of *SteaLTE* undetectable covert messages. The absolute value of the captured I/Q samples is centered around 1, while the remaining small peaks are hardly distinguishable from the noise of the wireless channel. The same behavior can also be observed in the CDF of the primary I/Q samples shown in Fig. 9d. When compared with the CDF of the primary-only LTE transmissions (solid line), the CDF of the covert signal not only does not show the steps observed with fixed displacements (dashed line), but it also results in a 4.8x shorter K-S distance.

III. SteaLTE Prototype

We prototyped *SteaLTE* on Commercial Off-the-Shelf (COTS) NI USRPs B210 and X310 Software-defined Radios (SDRs). Our implementation is based on the LTE-compliant srsLTE open-source software, which offers protocol stack implementations for LTE base stations (eNBs), UEs, and core network [29]. We remark that as *SteaLTE* follows a software-defined approach, it is not bound to LTE technology, and it can be easily extended to future 5G-and-beyond cellular networks.

We extended srsLTE to allow *SteaLTE* to *embed*, *encode*, and *decode* covert data on the downlink and uplink LTE primary traffic. Specifically, we enhanced the Physical Downlink Shared Channel (PDSCH) and Physical Uplink Shared Channel (PUSCH) procedures at the PHY-layer. The PDSCH carries the downlink data sent by the eNB to the UEs, and the random access response messages if the PDSCH is mapped to the Random Access Channel (RACH). The PUSCH carries the uplink data that the UEs transmit to the eNB, and ACKs and NACKs for PDSCH data. Fig. 11 depicts the modified structure of the *SteaLTE* covert transmitter, i.e., the PDSCH at the eNB downlink side or the PUSCH at the UE uplink side.

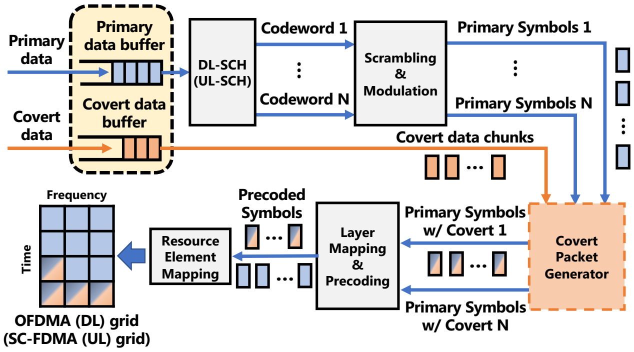


Fig. 11: The *SteaLTE* covert transmitter.

When there is primary data to transmit (either in downlink to the UE or uplink to the eNB), this is converted into *codewords* through the Downlink Shared Channel (DL-SCH) (or the Uplink Shared Channel (UL-SCH)), which performs transport channel encoding operations. Then, the resulting codewords are *scrambled* and *modulated* into *primary symbols* (for illustration purposes, in our experiments we adopt a QPSK modulation for the primary traffic that carries covert data). After these operations the *SteaLTE covert packet generator* (Section II-B1) modifies the amplitude of the resulting primary symbols, thus applying a second (covert) modulation to them. This way, the *covert data chunks* are embedded on the primary symbols. At this point, the complex-modulated *primary symbols with embedded covert* are mapped into layers for spatial multiplexing, and then precoded, as per LTE specifications. Finally, the resulting *precoded symbols* are mapped into resource elements to be transmitted via OFDMA (downlink), or SC-FDMA (uplink).

IV. Experimental Evaluation

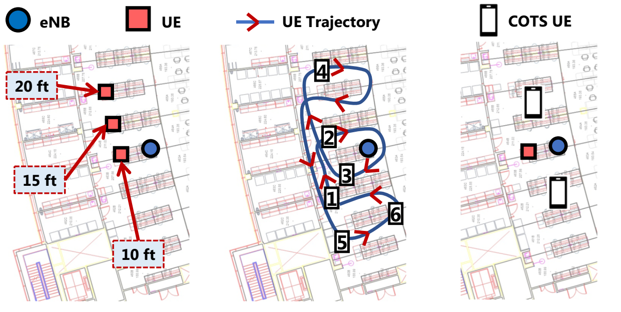
We report results from experimental campaigns for evaluating the performance of *SteaLTE*. Setups are described in Section IV-A. Results are shown and discussed in Section IV-B.

A. Experimental Setup

We evaluated *SteaLTE* on indoor and outdoor testbeds.

Indoor scenarios. For our indoor experiments we leveraged Arena, an indoor ceiling testbed covering an area of 2240 ft^2 [30]. We instantiate LTE-compliant eNBs and UEs on NI USRP X310 SDRs, USRP B210 and COTS Xiaomi Redmi Go smartphones. In all configurations the eNB uses a 10 MHz channel bandwidth corresponding to 50 Physical Resource Blocks (PRBs). These devices are used in the indoor testbed configurations depicted in Fig. 12.

The *static scenario* comprises one eNB and three UEs that are statically placed 10 ft, 15 ft and 20 ft away from the eNB (Fig. 12a). In this configuration all devices are USRPs X310. In



(a) Static scenario. (b) Dynamic scenario. (c) PCCaaS scenario.

Fig. 12: Indoor testbed setup and experiment configuration on the Arena testbed [30].

this scenario eNB and UE exchange primary traffic generated through *iperf3* [31], a software tool for network performance evaluation with TCP and UDP traffic. For experiments with UDP traffic the bitrate varies in the set $\{0.75, 2.5, 5\}$ Mbps. We also use traffic generated by a user-initiated speed test.
The *dynamic scenario* is made up of the eNB and one UE traveling a distance of 190ft around the eNB as illustrated in Fig. 12b. Measurements are collected at six different location in the UE journey. In this configuration the eNB is a USRP X310 and the mobile device is a USRP B210. Primary traffic is generated through the *ping* software utility, which uses Internet Control Message Protocol (ICMP) echo request and reply messages.
The *PCCaaS scenario* comprises two slices, one public and one private, with one eNB (USRP X310), one UE using the private slice for covert traffic (USRP X310), and two COTS Xiaomi Redmi Go smartphones transmitting data over the public slice. All users are statically positioned as in Fig. 12c within 10 ft from the eNB. In this scenario, primary traffic concerns YouTube videos streamed by the users.

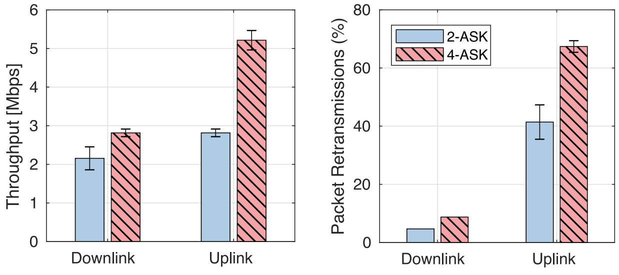
Outdoor scenario. For outdoor, long-range testing we ported *SteaLTE* to the Platforms for Advanced Wireless Research (PAWR) Platform for Open Wireless Data-driven Experimental Research (POWDER) platform [25, 26]. We use the NR version of srsLTE to instantiate one outdoor 5G base station (gNB) and one UE. The gNB is located on the rooftop of a 95 ft-tall building and is realized by a USRP X310. The UE is statically positioned at ground-level. It is implemented through a USRP B210. The distance between gNB and UE is 852 ft. The gNB uses a 3 MHz-bandwidth (15 PRBs). Covert data are embedded through a 2-ASK modulation. Primary traffic between gNB and UE is generated through the *ping* utility.

In all scenarios covert data are images and text files.

B. Experimental results

In this section we report the results of the performance evaluation of *SteaLTE* in each of the considered scenarios. For each scenario, we describe the investigated metrics and their relevance, and illustrate the corresponding experimental results. Plots include 95% confidence intervals (not shown if $< 1\%$).

1) Indoor static scenario: In the static scenario of Fig. 12a we start by measuring the performance of *SteaLTE* to deliver covert data by investigating throughout (data delivery over time) and the percentage of packets that needs to be retransmitted. Fig. 13 shows the downlink and uplink covert throughput and retransmissions for both 2-ASK and 4-ASK covert modulations under TCP primary traffic.

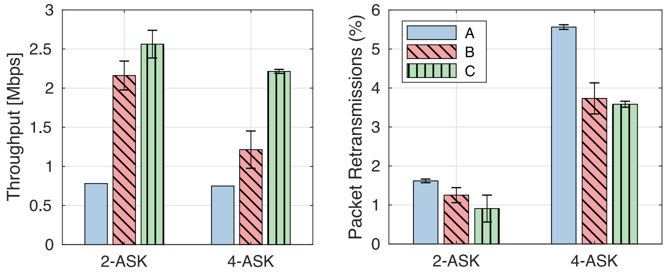


(a) Throughput. (b) Retransmissions.

Fig. 13: Downlink and uplink covert performance with TCP primary traffic for different covert modulations.

Being of higher order, 4-ASK obtains a higher throughput than 2-ASK (Fig. 13a). However, this comes at the cost of transmission errors, leading to a higher percentage of covert packet retransmissions (Fig. 13b). This is because of the higher resilience to errors of the 2-ASK modulation, which requires 38% less retransmissions than the 4-ASK case (uplink).

Fig. 14 depicts the covert throughput (Fig. 14a) and percentage of covert packet retransmissions (Fig. 14b) under UDP primary traffic. The *iperf3* UDP bitrate was set as follows: (A) 0.75 Mbps; (B) 2.5 Mbps, and (C) 5 Mbps.

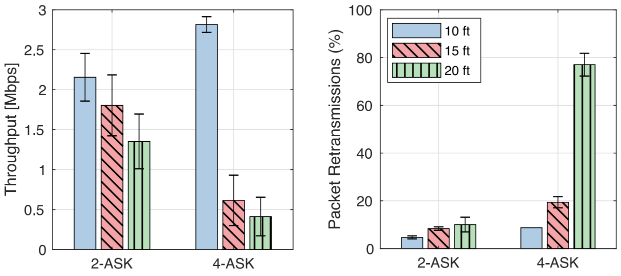


(a) Throughput. (b) Retransmissions.

Fig. 14: Downlink covert performance with UDP primary traffic for different traffic profiles and covert modulations.

Differently from TCP traffic (see downlink performance in Fig. 13a), the covert throughput is higher when data are embedded through a 2-ASK modulation. This is because, unlike TCP, UDP streams data at the specified bitrate, as it does not implement reliable data transfer. This behavior can be further observed in Fig. 14b, which shows a much higher (close to 3x) number of covert retransmissions in case of 4-ASK modulation.

Figures 15a and 15b show the covert throughput and the percentage of packet retransmissions when TCP primary traffic is exchanged between eNB and UE, as a function of their distance (Fig. 12a). We notice that for short distances (i.e., 10 ft) 4-ASK provides the highest performance. However, as the distance increases ($\geq$ 15 ft), modulating the covert message through 2-ASK yields a better performance, both for throughput and retransmissions. This is because of the higher robustness to errors of this modulation compared to the 4-ASK modulation.

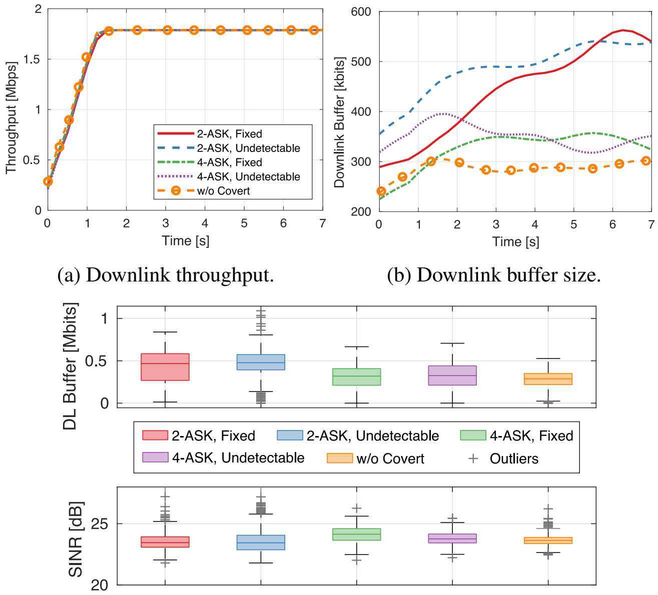


(a) Throughput.

(b) Retransmissions.

Fig. 15: Downlink covert performance with TCP primary traffic for different covert modulations and distances between eNB and UE.

We now investigate the impact of the undetectability schemes presented in Section II-D on the performance of primary transmissions. Their effectiveness in concealing covert data has been shown in Fig. 9. Here we show results for metrics that indicate that these schemes do not have a significant impact over the quality of transmission and on channel quality: Throughput, the number of bytes that are to be transmitted in the downlink and the Signal-to-Interference-plus-Noise Ratio (SINR). The results shown in Fig. 16 refer to UE-generated traffic according to a speed test application at 1.8 Mbps.



(a) Downlink throughput.

(b) Downlink buffer size.

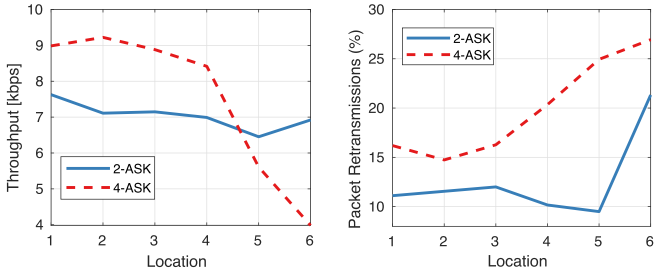
(c) Downlink buffer and SINR.

Fig. 16: Impact of *SteaLTE* on speed test primary traffic.

Figures 16a and 16b show the downlink primary throughput and transmission buffer size averaged over 10 independent experiments. Fig. 16a clearly indicates that embedding covert messages—both fixed and undetectable—does not noticeably affect primary traffic. In Fig. 16b we notice a slight increase of the size of the downlink buffer queue when covert communications happen, especially for the 2-ASK undetectability schemes. This suggests that a higher number of retransmissions is needed for the eNB to deliver the primary traffic to the UEs, with an impact on the number of resources that the eNB uses to communicate to the users. Yet again, in this scenario, this does not translate in a noticeable degradation of the throughput.

Fig. 16c shows the distribution of the downlink buffer size (top) and the SINR measured by the user (bottom). These two results show that the statistical distributions of these two metrics do not vary significantly in the case when *SteaLTE* is used (independently of the undetectability scheme and modulation used) and the case when it is not. Particularly, the average difference among the downlink buffer size in the two cases never exceeds 106.5 kbits. Also, the difference among SINR is within 0.15 dB, indicating that embedding covert data does not increase noise perceptively.

2) Indoor dynamic scenario: The throughput and retransmission performance of *SteaLTE* for the different user locations of Fig. 12b and covert modulations is shown in Fig. 17.



(a) Throughput.

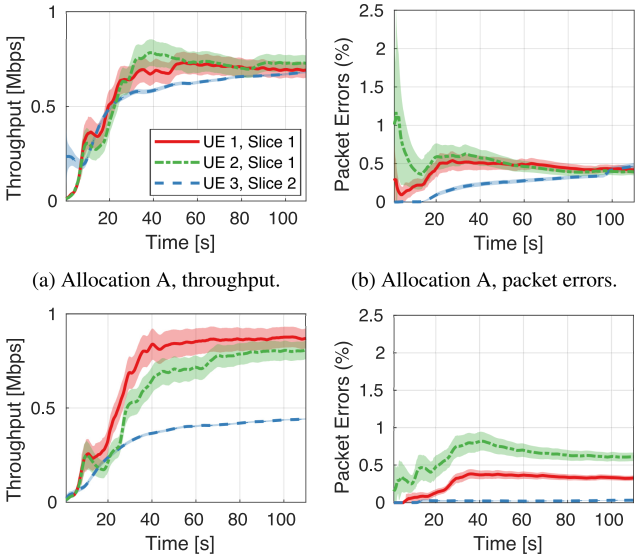
(b) Retransmissions.

Fig. 17: Covert performance with ICMP echo reply primary traffic for different covert modulations in presence of UE mobility.

As the distance between eNB and UE increases (as for locations 4, 5 and 6—Fig. 12b), the covert throughput (Fig. 17a) for both 2-ASK and 4-ASK modulations decreases, while the percentage of packet retransmissions increases (Fig. 17b). As noticed before, in general, 2-ASK outperforms 4-ASK because of its higher robustness to channel impairments, which are more noticeable at larger distances. Despite the performance degradation, *SteaLTE* still manages to enable secret communications between covert transmitter and receiver in presence of mobility.

3) Indoor PCCaaS scenario: We now set to investigate the effectiveness of *SteaLTE* for instantiating private network slices on a shared cellular infrastructure, which serves the need of those critical applications requiring rapid instantiation of private and secure cellular networks. In this set of experiments, we instantiate two slices: A primary-only slice (Slice 1) serving covert-agnostic UEs 1 and 2 (for which we use smartphones), and a *SteaLTE* private slice (Slice 2) for covert communications between UE 3 and the eNB (both USRPs X310). The covert data of UE 3 is embedded through 4-ASK modulation. We consider two different network slicing allocations: (A) Spectrum resources are evenly split among the two slices, and (B) 70% of the resources are allocated to Slice 1 and 30% to Slice 2. For these scenarios we investigate the primary throughput and the percentage of packets erroneously received for both slicing configurations when all users stream videos from YouTube. Results are shown in Fig. 18.

We notice that in both allocations, the primary throughput of all users is not impacted by the presence of covert communi-



(a) Allocation A, throughput.

(b) Allocation A, packet errors.

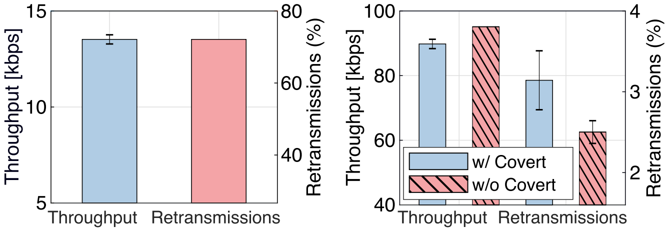
(c) Allocation B, throughput.

(d) Allocation B, packet errors.

Fig. 18: Throughput and percentage of packet errors on primary traffic for different resource allocations of the private and standard slices.

cations. The percentage of packet errors (Figures 18b and 18d) is negligible in both allocations (i.e., below 0.5% on average, with a peak of 2.5%). As a result, Figures 18a and 18c show that the throughput level achieved by all users is enough for rate-demanding applications, thus confirming the low impact of *SteaLTE* on primary communications.

4) Outdoor scenario: SteaLTE has been tested also on a long-rage link ($>$ 850 ft) in the outdoor scenario provided by the POWDER platform [25]. Results on throughput and retransmission percentage are shown in Fig. 19. Specifically, Fig. 19a shows downlink covert throughput and retransmissions and Fig. 19b depicts the same metrics for the primary traffic (with and without covert).



(a) Covert.

(b) Primary.

Fig. 19: Long-range experiments on the POWDER platform.

Fig. 19a shows that *SteaLTE* is capable of delivering covert data despite the severe path-loss, multi-path and fading conditions experienced over the long-range link, thus demonstrating its suitability for traditional (mostly outdoors) cellular applications. As for the indoor results above, Fig. 19b confirms that embedding covert data into primary traffic has negligible effect on its throughput performance. The primary throughput degradation over the long-range link is merely 5.59%, whereas the packet retransmission percentage increases from 2.6% to 3.2% (0.6% increase) when covert traffic is embedded.

V. Related Work

Wireless steganography has been frequently used for covert communications among parties. Differently from approaches where covert data are embedded in the packet control fields (e.g., checksum [32], flags [33, 34], and padding fields [35], among others [36, 37]), wireless steganography introduces tiny displacements in the I/Q constellation plane that can be controlled to encode covert information. Typical methods include frequency/phase shifts [17, 23], I/Q imbalance [38], superimposing noisy constellations [16, 24, 39, 40] or training and preamble sequences manipulations [23, 41]. These approaches, however, lack reliability as they are prone to demodulation errors and rarely support long-range communications, quintessential for many communication systems.

These reliability issues have been partially addressed at the higher layers of the protocol stack. Hamdaqa and Tahvildari describe a steganographic system for Voice-over-IP (VoIP) that encodes covert information by carefully delaying packet transmissions [42]. Although this approach is highly reliable and undetectable, it operates over large temporal windows, which considerably limits the achievable covert rate. Nain and Rajalakshmi develop a steganographic communication system that hides information over chip sequences of IEEE 802.15.4 networks integrating error-coding techniques to mitigate errors [43]. Although being reliable, this method only achieves low transmission rates. Overall, previous solutions either achieve *low covert throughput*, or are highly detectable through steganalysis [28, 44], or lack practical implementations demonstrating their effectiveness and feasibility.

SteaLTE is the missing answer to the high throughput, reliability, and undetectability requirements of PCCaaS applications characterized by mobility, time-varying channels and large distances. As a reliable end-to-end steganographic system *SteaLTE*: (i) Achieves high throughput through wireless steganography; (ii) provides reliable and channel-resilient communications through a combination of error-coding, retransmissions and adaptive covert modulation schemes, and (iii) can be seamlessly integrated in 3GPP-compliant cellular systems.

VI. Conclusions

This paper proposes *SteaLTE*, the first practical PCCaaS-enabling system for softwarized cellular networks. *SteaLTE* supports reliable, undetectable, high-throughput, and long-range covert communications. We have prototyped *SteaLTE* and implemented it on LTE-compliant testbeds, including indoor settings and the PAWR POWDER platform (ourdoors). We have extensively evaluated the performance of *SteaLTE* under diverse traffic profiles, distance and mobility patterns, highlighting the feasibility of undetectable transmissions and their negligible impact on primary traffic. Results over the multiplicity of the considered scenarios show that, in the vast majority of our experiments *SteaLTE* achieves a throughput of covert traffic that is at least 90% of the throughput of the primary traffic, affecting the latter negligibly ($<$ 6% loss).

References

[1] P. Rost, C. Mannweiler, D. S. Michalopoulos, C. Sartori, V. Sciancalepore, N. Sastry, O. Holland, S. Tayade, B. Han, D. Bega, D. Aziz, and H. Bakker, "Network slicing to enable scalability and flexibility in 5G mobile networks," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 72–79, May 2017.

[2] O-RAN Alliance, "O-RAN: Towards an open and smart RAN," https://www.o-ran.org/s/O-RAN-WP-FInal-181017.pdf, Oct. 2018.

[3] The Linux Foundation, "Open network automation platform architecture," https://www.onap.org/wp-content/uploads/sites/20/2018/11/ONAP_CaseSolution_Architecture_112918FNL.pdf, 2018.

[4] M. Moradi, W. Wu, L. E. Li, and Z. M. Mao, "SoftMoW: Recursive and reconfigurable cellular WAN architecture," in *Proceedings of ACM CoNEXT 2014*, Sydney, Australia, December 2–5 2014, pp. 377–390.

[5] L. Bonati, M. Polese, S. D'Oro, S. Basagni, and T. Melodia, "Open, programmable, and virtualized 5G networks: State-of-the-art and the road ahead," *Computer Networks*, vol. 182, pp. 1–18, December 2020.

[6] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven nextG cellular networks," *arXiv:2012.01263 [cs.NI]*, December 2020.

[7] L. Bonati, S. D'Oro, L. Bertizzolo, E. Demirors, Z. Guan, S. Basagni, and T. Melodia, "CellOS: Zero-touch softwarized open cellular networks," *Computer Networks*, vol. 180, pp. 1–13, October 2020.

[8] S. D'Oro, F. Restuccia, and T. Melodia, "The slice is served: Enforcing radio access network slicing in virtualized 5G systems," in *Proceedings of IEEE INFOCOM*, Paris, France, May 2019.

[9] S. D'Oro, F. Restuccia, and T. Melodia, "Toward operator-to-waveform 5G radio access network slicing," *IEEE Communications Magazine*, vol. 58, no. 4, pp. 18–23, April 2020.

[10] S. D'Oro, L. Bonati, F. Restuccia, M. Polese, M. Zorzi, and T. Melodia, "Sl-EDGE: Network slicing at the edge," in *Proceedings of ACM Mobihoc*, Virtual Conference, October 2020.

[11] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429–2453, March 2018.

[12] C. Marquez, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Pérez, "Resource sharing efficiency in network slicing," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 909–923, September 2019.

[13] D. Bega, M. Gramaglia, A. Banchs, V. Sciancalepore, K. Samdanis, and X. Costa-Pérez, "Optimising 5G infrastructure markets: The business of network slicing," in *Proceedings of IEEE INFOCOM 2017*, Atlanta, GA, May 1–4 2017, pp. 1–9.

[14] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, May 2017.

[15] F. Y. Shih, *Digital watermarking and steganography: Fundamentals and techniques*. CRC Press, April 2017.

[16] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "Secret agent radio: Covert communication through dirty constellations," in *Proceeding of Springer IH 2012*, Berkeley, CA, May 15–18 2012, pp. 160–175.

[17] S. Grabski and K. Szczypiorski, "Steganography in OFDM symbols of fast IEEE 802.11n networks," in *Proceedings of the IEEE CS Security and Privacy Workshops 2013*, San Francisco, CA, May 24–25 2013, pp. 158–164.

[18] K. Szczypiorski and W. Mazurczyk, "Hiding data in OFDM symbols of IEEE 802.11 networks," in *Proceedings of IEEE MINES 2010*, Nanjing, Jiangsu, China, November 4–6 2010, pp. 835–840.

[19] T. Kho, "Steganography in the 802.15.4 physical layer," *U.C. Berkeley*, December 17 2007.

[20] E. Zielinska and K. Szczypiorski, "Direct sequence spread spectrum steganographic scheme for IEEE 802.15.4," in *Proceedings of IEEE MINES*, Shanghai, China, November 2011, pp. 586–590.

[21] B. A. Bash, D. Goeckel, D. Towsley, and S. Guha, "Hiding information in noise: Fundamental limits of covert wireless communication," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 26–31, December 2015.

[22] D. Kahn, "The history of steganography," in *Proceedings of Springer IH 1996*, Cambridge, U.K., May 30–June 1 1996, pp. 1–5.

[23] J. Classen, M. Schulz, and M. Hollick, "Practical covert channels for WiFi systems," in *Proceedings of IEEE CNS 2015*, Florence, Italy, September 28–30 2015, pp. 209–217.

[24] S. D'Oro, F. Restuccia, and T. Melodia, "Hiding data in plain sight: Undetectable wireless communications through pseudo-noise asymmetric shift keying," in *Proceedings of IEEE INFOCOM 2019*, Paris, France, April 29–May 2 2019, pp. 1585–1593.

[25] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, M. Hibler, D. Johnson, S. Kumar Kasera, E. Lewis, D. Maas, A. Orange, N. Patwari, D. Reading, R. Ricci, D. Schurig, L. B. Stoller, K. Van der Merwe, K. Webb, and G. Wong, "POWDER: Platform for open wireless data-driven experimental research," in *Proceedings of ACM WiNTECH*, September 2020.

[26] Platforms for Advanced Wireless Research. (2020) https://advancedwireless.org.

[27] J. F. Kurose and K. W. Ross, *Computer networking: A top-down approach*, 7th ed. Pearson, 2017.

[28] Z. Xia, X. Wang, X. Sun, and B. Wang, "Steganalysis of least significant bit matching using multi-order differences," *Wiley Security and Communication Networks*, vol. 7, no. 8, pp. 1283–1291, August 2014.

[29] I. Gomez-Miguelez, A. Garcia-Saavedra, P. D. Sutton, P. Serrano, C. Cano, and D. J. Leith, "srsLTE: An open-source platform for LTE evolution and experimentation," in *Proceedings of ACM WiNTECH 2016*, New York City, NY, October 3 2016, pp. 25–32.

[30] L. Bertizzolo, L. Bonati, E. Demirors, A. Al-Shawabka, S. D'Oro, F. Restuccia, and T. Melodia, "Arena: A 64-antenna SDR-based ceiling grid testing platform for sub-6 GHz 5G-and-beyond radio spectrum research," *Computer Networks*, vol. 181, pp. 1–17, November 2020.

[31] Iperf3. (2021) https://iperf.fr.

[32] Z. Liu, Y. Jiang, and P. Qian, "A data-hiding method based on TCP/IP checksum," in *Advances in Computer Science and its Applications*. Springer, 2014, pp. 1039–1044.

[33] S. J. Murdoch and S. Lewis, "Embedding covert channels into TCP/IP," in *Proceedings of Springer IH 2005*, Barcelona, Spain, June 6–8 2005, pp. 247–261.

[34] K. Ahsan and D. Kundur, "Practical data hiding in TCP/IP," in *Proceeding of the ACM IH&MMSec*, Juan-les-Pins, France, December 6 2002, pp. 1–8.

[35] I. Grabska and K. Szczypiorski, "Steganography in long term evolution systems," in *Proceedings of the IEEE CS Security and Privacy Workshops 2014*, San Jose, CA, May 17–18 2014, pp. 92–99.

[36] D. Martins and H. Guyennet, "Steganography in MAC layers of 802.15.4 protocol for securing wireless sensor networks," in *Proceedings of IEEE MINES 2010*, Nanjing, Jiangsu, China, November 4–6 2010, pp. 824–828.

[37] A. M. Mehta, S. Lanzisera, and K. S. J. Pister, "Steganography in 802.15.4 wireless communication," in *Proceedings of IEEE ANTS 2008*, Mumbai, India, December 15–17 2008, pp. 1–3.

[38] K. Sankhe, F. Restuccia, S. D'Oro, T. Jian, Z. Wang, A. Al-Shawabka, J. Dy, T. Melodia, S. Ioannidis, and K. Chowdhury, "Impairment shift keying: Covert signaling by deep learning of controlled radio imperfections," in *Proceedings of IEEE MILCOM 2019*, Norfolk, VA, November 12–14 2019, pp. 598–603.

[39] P. Cao, W. Liu, G. Liu, X. Ji, J. Zhai, and Y. Dai, "A wireless covert channel based on constellation shaping modulation," *Hindawi Security and Communication Networks*, vol. 2018, pp. 1–15, January 8 2018.

[40] V. Kumar, J.-M. Park, T. C. Clancy, and K. Bian, "PHY-layer authentication using hierarchical modulation and duobinary signaling," in *Proceedings of IEEE ICNC 2014*, Honolulu, HI, February 3–6 2014, pp. 782–786.

[41] Z. Hijaz and V. S. Frost, "Exploiting OFDM systems for covert communication," in *Proceedings of IEEE MILCOM 2010*, San Jose, CA, October 31–November 3 2010, pp. 2149–2155.

[42] M. Hamdaqa and L. Tahvildari, "ReLACK: A reliable VoIP steganography approach," in *Proceedings of IEEE SSIRI 2011*, Jeju Island, South Korea, June 27–29 2011, pp. 189–197.

[43] A. K. Nain and P. Rajalakshmi, "A reliable covert channel over IEEE 802.15.4 using steganography," in *Proceedings of IEEE WF-IoT 2016*, Reston, VA, December 12–14 2016, pp. 711–716.

[44] S. Grabski and K. Szczypiorski, "Network steganalysis: Detection of steganography in IEEE 802.11 wireless networks," in *Proceedings of IEEE ICUMT*, Almaty, Kazakhstan, September 10–13 2013, pp. 13–19.