This is Your President Speaking: Spoofing Alerts in 4G LTE Networks

Gyuhong Lee*
University of Colorado Boulder
gyuhong.lee@colorado.edu

Jihoon Lee*
University of Colorado Boulder
jihoon.lee-1@colorado.edu

Jinsung Lee
University of Colorado Boulder
jinsung.lee@colorado.edu

Youngbin Im
University of Colorado Boulder
youngbin.im@colorado.edu

Max Hollingsworth
University of Colorado Boulder
max.hollingsworth@colorado.edu

Eric Wustrow
University of Colorado Boulder
ewust@colorado.edu

Dirk Grunwald
University of Colorado Boulder
dirk.grunwald@colorado.edu

Sangtae Ha
University of Colorado Boulder
sangtae.ha@colorado.edu

ABSTRACT

Modern cell phones are required to receive and display alerts via the Wireless Emergency Alert (WEA) program, under the mandate of the Warning, Alert, and Response Act of 2006. These alerts include AMBER alerts, severe weather alerts, and (unblockable) Presidential Alerts, intended to inform the public of imminent threats.

Recently, a test Presidential Alert was sent to all capable phones in the United States, prompting concerns about how the underlying WEA protocol could be misused or attacked. In this paper, we investigate the details of this system, and develop and demonstrate the first practical spoofing attack on Presidential Alerts, using both commercially available hardware as well as modified open source software.

Our attack can be performed using a commercially-available software defined radio, and our modifications to the open source NextEPC and srsLTE software libraries. We find that with only four malicious portable base stations of a single Watt of transmit power each, almost all of a 50,000-seat stadium can be attacked with a 90% success rate. The true impact of such an attack would of course depend on the density of cell phones in range; fake alerts in crowded cities or stadiums could potentially result in cascades of panic.

Fixing this problem will require a large collaborative effort between carriers, government stakeholders, and cell phone manufacturers. To seed this effort, we also discuss several defenses to address this threat in both the short and long term.

*Both authors contributed equally to this research.

MobiSys '19, June 17–21, 2019, Seoul, Republic of Korea

ACM ISBN 978-1-4503-6661-8/19/06…$15.00
https://doi.org/10.1145/3307334.3326082

CCS CONCEPTS

• **Security and privacy → Mobile and wireless security**; *Spoofing attacks*;

KEYWORDS

Spoofing; Presidential Alert; WEA; CMAS; LTE; Security

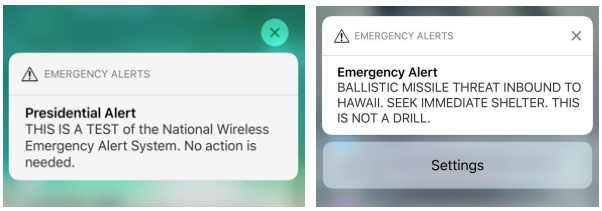
ACM Reference Format:
Gyuhong Lee, Jihoon Lee, Jinsung Lee, Youngbin Im, Max Hollingsworth, Eric Wustrow, Dirk Grunwald, and Sangtae Ha. 2019. This is Your President Speaking: Spoofing Alerts in 4G LTE Networks. In *The 17th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '19), June 17–21, 2019, Seoul, Republic of Korea.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3307334.3326082

1 INTRODUCTION

The Wireless Emergency Alerts (WEA) program is a government-mandated service in commercialized cellular networks in the United States. WEA was established by the Federal Communications Commission (FCC) in response to the Warning, Alert, and Response Act of 2006 to allow wireless cellular service providers to send geographically targeted emergency alerts to their subscribers. The Federal Emergency Management Agency (FEMA) is responsible for the implementation and administration of a major component of WEA called the Integrated Public Alert and Warnings System (IPAWS) [47]. IPAWS enables authorized public safety officials to send 90-character, geographically-targeted alerts to the public via commercial mobile service providers (CMSPs) [45].

This system can send three types of alerts: **Presidential Alerts** issued by the president to all of the United States; **Imminent Threat Alerts** involving serious threats to life and property, often related to severe weather; and **AMBER Alerts** regarding missing or abducted children. Considering the number of cell phone users and the nation-wide coverage of cellular networks, WEA over LTE was a natural step to enhance public safety *immediately* and *effectively*. In fact, recent rapidly moving fires have caused Emergency Services to consider using WEA instead of relying on opt-in alerting systems [36].

Lately though, a handful of widely publicized events has led to public scrutiny over the potential misuse of the alert system. On



(a) Presidential alert (b) Imminent threat alert

Figure 1: Snapshots of real WEA alerts received by cell phones: (a) the first national test of the Presidential Alert performed on Oct. 3, 2018 in the US, and (b) a false alert sent in Hawaii on Jan. 13, 2018.

Jan. 13, 2018, there was a geographically-targeted alert issued in Hawaii. The message, warning of an inbound missile, is shown in Figure 1b. Although caused by human error, the impact to the residents of Hawaii was huge, as it led to panic and disruption throughout the state [49]. This event was followed on Oct. 3, 2018, by the first national test of a mandatory Presidential Alert. The alert, captured in Figure 1a, was sent to all capable phones in the United States [42].

These recent high-profile alerts, have prompted us to assess the realizability and impact of an alert spoofing attack. In this paper we demonstrate how to launch a Presidential Alert-spoofing attack, and evaluate its effectiveness with respect to attack coverage and success rate.

To answer this question we start by looking into the alert delivery method used by WEA. WEA sends alerts via the commercial mobile alert service (CMAS) standard[1]. These alerts are delivered via the LTE downlink within broadcast messages, called System Information Block (SIB) messages. A cell tower (referred to as eNodeB) broadcasts the SIB to every cell phone (referred to as user equipment, or UE) that is tuned to the control channels of that eNodeB. A UE obtains necessary access information, like the network identifier and access restrictions, from SIB messages and uses it for the eNodeB selection procedure. Among the 26 different types of SIB messages, SIB12 contains the CMAS notification, which delivers the aforementioned alert messages to the UEs (greater detail in §2).

The eNodeB broadcasts SIB messages to the UE, independently from the mutual authentication procedure that eventually occurs between them. Thus, all SIBs, including CMAS, are intrinsically *vulnerable* to spoofing from a malicious eNodeB. More importantly, even if the UE has completed its authentication and securely communicates with a trusted eNodeB, the UE is still exposed to the security threat caused by the broadcasts from other, possibly malicious, eNodeBs. This is due to the fact that the UE periodically gathers SIB information from neighboring eNodeBs for potential eNodeB (re)selection and handover.

The UE's connection state can be classified into *active* and *idle* modes, depending on where the UE falls in the cell attachment process. Based on the UE's state, we analyze its vulnerability to the spoofing attack. Next, we develop the attack model from our novel analysis and implement the CMAS spoofing attack system using commercial off-the-shelf (COTS) software defined radio (SDR) hardware and modified open source srsLTE [24] and NextEPC [37] software libraries. We then evaluate those attacks in a responsible and controlled manner: all tested phones are put into a radio-isolated shield box and the signal emitted by our malicious eNodeB is completely isolated to the outside. To the best of our knowledge, this is *the first experimentally verified work* that discloses the potential risk of CMAS spoofing. Note that LTE networks currently deployed in most countries (e.g., countries in Europe, United States, and South Korea) have adopted public warning systems that follow the same architecture principles as CMAS [11], making them potential targets for the same attack.

We found via both experiment and simulation that a 90% success rate can be reached in 4,435m^2 of a 16,859m^2 building using a single malicious eNodeB of 0.1 Watt power, while in an outdoor stadium, 49,300 seats among the total 50,000 are hit with an attack, which itself has a 90% success rate using four malicious eNodeBs of 1 Watt power.

In summary, we make following major contributions:

- We identify security vulnerabilities of the WEA system and explain the detailed underlying mechanism stipulated by the LTE standard. We find that the CMAS spoofing attack is easy to perform but is challenging to defend in practice.
- We present our threat analysis on the CMAS spoofing attack, and implement an effective attack system using COTS SDR hardware and open source LTE software.
- We confirm that the CMAS spoofing attack can succeed in all 9 of the smartphones (from 5 manufacturers) that we tested.
- We evaluate our attack system using both SDR-based hardware prototype and measurement-based simulation. As one of the striking results, we demonstrate that four SDR-based malicious eNodeBs at 1-Watt of power can propagate their signal to 49,300 of the total 50,000-seat football stadium. Of the 49,300 seats affected, 90% will receive the CMAS message.
- We discuss possible solutions to prevent such a spoofing attack with thorough analysis and feasibility test, which can open the door toward collaborative efforts between cellular operators, government stakeholders, and phone manufacturers.

Responsible Disclosure. In Jan. of 2019, before public release, we disclosed the discoveries and technical details of this alert spoofing attack to various pertinent parties. These parties include the government and standardization organizations FEMA, FCC, DHS, NIST, 3GPP, and GSMA; the cellular network service providers AT&T, Verizon, T-Mobile, Sprint, and U.S.Cellular; and the manufacturers Samsung, Google, and Apple.

2 BACKGROUND

Here we review the background of the WEA service in the United States, whose underlying delivery architecture has been adopted in many other countries [11], and describe how these alert messages are delivered to a UE over the LTE network.

The 3GPP standardization body began a project in 2006 to define the requirements of CMAS in order to deliver WEA messages in the LTE network. The resulting technical specification, initially

[1]For clarity, we will use WEA to refer to the alert service and CMAS to refer to the underlying delivery technology.

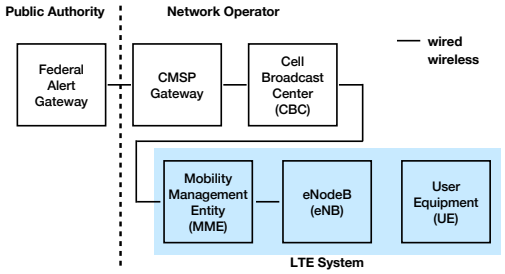


Figure 2: LTE CMAS network architecture.

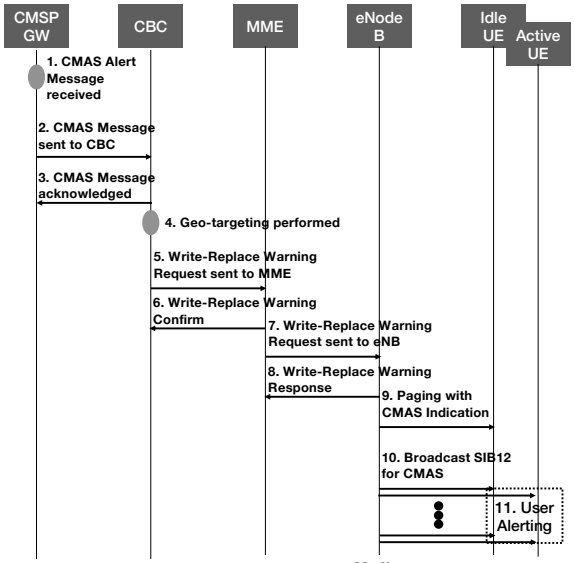
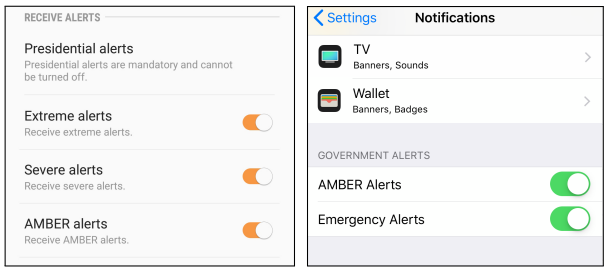


Figure 3: CMAS call flow.



(a) Android alert setting **(b) iPhone alert setting**

Figure 4: Government alert settings in mobile phones: (a) Android and (b) Apple's iPhones. Although AMBER and emergency alerts can be manually disabled, users cannot disable or block Presidential Alerts from being received or displayed.

released in 2009, describes general criteria for the delivery of alerts, message formats, and functionality of CMAS-capable UEs [3].

Figure 2 illustrates the LTE CMAS network architecture. During an emergency, authorized public safety officials send alert messages to Federal Alert Gateways. The participating CMSPs then broadcast the alert to the UEs, who will automatically receive the alert if they are located in or travel to the targeted geographic area. The cell broadcast center (CBC) is part of the core network and connected to the Mobility Management Entity (MME) which maintains the location information of the UEs attached to the network [5]. The eNodeB is the final step in communicating the alert to the UEs over the air.

Figure 3 shows a more detailed call flow of the CMAS procedure. An authorized official originates the alert with FEMA-approved alert origination software (step 1). The CMSP Gateway delivers emergency information to the CBC (steps 2 and 3), and the CBC performs geo-targeting which selects the eNodeBs where the alert is to be delivered (step 4) [46]. The CBC then identifies which MMEs need to be contacted and sends a *Write-Replace Warning Request* message containing the alert message and delivery attributes: Message Identifier, Serial Number, Tracking Area ID list, Warning Area, etc. (step 5) [3]. The MME sends a *Write-Replace Warning Confirm* message that indicates to the CBC that the MME has started to distribute the warning message to eNodeBs (step 6). If an eNodeB receives the *Write-Replace Warning Request* from its MME (step 7), it replies with a *Write-Replace Warning Response* message (step 8). A duplicated request can be detected by inspecting the Serial Number at the eNodeB. If it is identified as a new alert, the eNodeB sends a paging signal with a CMAS indication to wake up all UEs in idle mode (step 9). The alert message is broadcast via a SIB12 message over the air (step 10) [8], and finally all UEs will receive the alert, irrespective of whether their connection state is either idle or active (step 11).

Among the three types of emergency alerts listed in §1, UEs may choose to turn off the notification of emergency alerts and AMBER alerts. However, the 3GPP mandated that the reception of Presidential Alerts is obligatory. Thus, cell phones have no option to disable Presidential Alert, as seen in Figure 4.

Because it cannot be disabled, this paper focuses on spoofing Presidential Alerts. Moreover, the attack can be performed without involving any of the IPAWS architecture or protocol described above. Instead, the attack begins with the injection of a fake CMAS message at the wireless stage from a rouge eNodeB (steps 9 and 10 in Figure 3).

3 PRELIMINARY INVESTIGATION

In this section we describe our end-to-end LTE CMAS testbed, as well as the current security issues discovered using our testbed. Next, we explain the threat methodology depending on the UE state and further derive the threat impact as a function of the signal power from a malicious eNodeB relative to the signal power from a trusted eNodeB.

3.1 Building a CMAS-Enabled LTE Testbed

To analyze the specifics of CMAS alerts in detail, we built an LTE network in our lab and augmented it with CMAS capabilities.

The testbed, shown in Figure 5, consists of UEs, eNodeB, and MME. For the eNodeB, we use a COTS LTE small cell, Juni JL620 [27], which supports CMAS and 2x2 MIMO. We located the UEs inside an RF-isolated shield box, which prevents our experiments from unintentionally interfering with real devices. Our UEs communicate

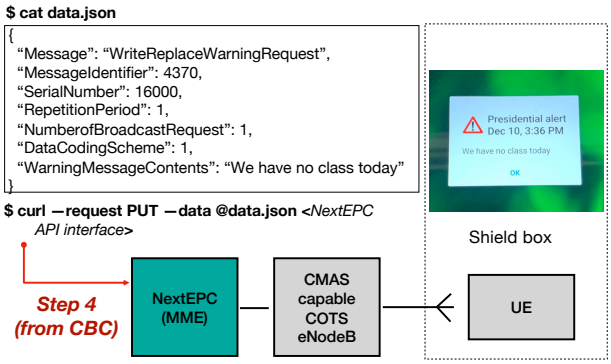


Figure 5: LTE testbed setup for a mobile phone to receive a CMAS notification.

Table 1: CBC configuration

Field	Description
MessageIdentifier	Type of CMAS alert [3]. '4370' is the Presidential Alert, '4379' is an AMBER alert.
SerialNumber	Identifier of a CMAS message to detect duplicates.
RepetitionPeriod	Defines the duration in seconds between broadcasts over the air.
NumberBroadcastRequests	Defines the number of times to broadcast over the air.
DataCodingScheme	Encoding scheme of the message content [2].
WarningMessageContents	Is the alert text to be shown to users.

with the eNodeB via a pair of antennas also inside the box. For the LTE core including MME, we use the open source NextEPC [37] software.

In the testbed, the alert process starts with the CBC. The CBC is the go-between for the alert originator and the LTE core, or Evolved Packet Core (EPC). We implemented our own CBC with a REST (Representational State Transfer) API and changes to the EPC. Our implementation provides a number of configurable items summarized in Table 1.

3.2 Identifying the Vulnerability

An eNodeB broadcasts LTE system information through the Master Information Block (MIB) and SIB. Specifically, when a UE searches for an eNodeB, it searches for the eNodeB's physical cell identifier (PCI) within a dedicated synchronization channel specified by the LTE standard [7]. After finding the PCI, the UE unscrambles the MIB which contains essential information such as the system bandwidth, system frame number (SFN), and antenna configuration in order to decode the SIB Type 1 message (SIB1). There are several SIB messages but only SIB1 has a fixed periodicity of 80 msec. Other SIB messages are dynamically scheduled by the eNodeB, and the scheduling information for other SIBs is encoded in the periodic SIB1. Each SIB message has a different role. For example, SIB2 has information about random access for initiating a data transfer and uplink power control.

3GPP specifies that the broadcast of CMAS messages is over the air through SIB12 [8]. However, unlike point-to-point messages in LTE, broadcasts of SIB messages are not protected by mutual cryptographic authentication or confidentiality, since the SIB contains essential information the UEs use to access the network before any session keys have been established. The contents of a CMAS message is a simple GSM 7-bit encoded text (the same format used in the traditional Short Message Service (SMS) [35]). Once a CMAS message has been received, there is no verification method for the message content. If an attacker can imitate eNodeB behavior closely enough to broadcast false CMAS notifications, then the UE will display them to the user.

A UE's vulnerability to a fake CMAS alert depends on whether it's in an *active* or *idle* state, illustrated in Figure 8. To affect the most UEs an attacker must consider different approaches for each state. Here we discuss idle UEs and active UEs separately:

***Idle* mode UEs.** Reference Signal Received Power (RSRP) is the power of an eNodeB-specific reference signal recognized by the UE, which is typically used to make an eNodeB selection and handover decision. Normally whenever a UE in idle mode performs eNodeB selection (or reselection), it will associate with the eNodeB having the highest RSRP. Therefore, we can formulate the selection of a specific eNodeB (indexed by k^*) done by the idle UE (indexed by i) is as follows:

$$k^* = \arg\max_{k \in C_i} \{\text{RSRP}_k\}, \quad (1)$$

where C_i is the set of eNodeBs that are observed by the UE_i. If the RSRP of a malicious eNodeB is the strongest, the UE decodes the SIBs transmitted by the malicious eNodeB. The attacker does not need to have any user information (including security keys) of UE_i, which would be stored in the database of the network operator. Without having such user information, UE_i will eventually reject the authentication process with the malicious eNodeB. However, UE_i can receive a CMAS message transmitted by the malicious eNodeB during this process, as shown in Case 3 of Table 2.

Since a UE tries its authentication to the network up to five times before it listens to other frequency channels, the malicious eNodeB can leverage this period to send a CMAS message. Figure 6 shows the S1AP[2] message exchange between the MME and eNodeB when there is no user information in the Home Subscriber Server (HSS)[3]. After the MME replies with *Attach Reject* due to the failure in user authentication, the UE re-sends *Attach Request* four more times (totaling 5 requests), which takes 42.06 secs. This means a UE stays up to 42 seconds in the vulnerable *"Searching"* state shown in Figure 8. This allows an attacker plenty of time for the CMAS spoofing; assuming a 160–msec periodicity of SIB12 transmissions, 262 fake alerts can be received by a UE during this period.

***Active* mode UEs.** When a UE is in active mode, it securely communicates with one serving eNodeB. If it finds another eNodeB with a higher power level than the existing serving eNodeB, a handover procedure can be triggered: the UE measures the RSRP of the candidate eNodeB and sends the measurement report to the serving eNodeB. The serving eNodeB then makes a handover decision based on the received measurement report. However, if the target

[2]S1 Application Protocol (S1AP) is the 3GPP standard protocol between the MME and eNodeB [9].
[3]HSS is the user subscription database located in the LTE core network.

Time	Protocol	Length	Info
1.575193274	S1AP/NAS-	106	SACK id-downlinkNASTransport, Attach reject (IMSI unknown in HSS)
1.580086657	S1AP	82	id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]
1.581279499	S1AP	82	id-UEContextRelease, UEContextReleaseComplete
11.691305719	S1AP/NAS-	210	id-initialUEMessage, Attach request, PDN connectivity request
11.695100769	S1AP/NAS-	106	SACK id-downlinkNASTransport, Attach reject (IMSI unknown in HSS)
11.699968948	S1AP	82	id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]
11.701197824	S1AP	82	id-UEContextRelease, UEContextReleaseComplete
21.811210376	S1AP/NAS-	210	id-initialUEMessage, Attach request, PDN connectivity request
21.812220606	S1AP/NAS-	106	SACK id-downlinkNASTransport, Attach reject (IMSI unknown in HSS)
21.819861859	S1AP	82	id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]
21.821122628	S1AP	82	id-UEContextRelease, UEContextReleaseComplete
31.931121534	S1AP/NAS-	210	id-initialUEMessage, Attach request, PDN connectivity request
31.932141027	S1AP/NAS-	106	SACK id-downlinkNASTransport, Attach reject (IMSI unknown in HSS)
31.939778089	S1AP	82	id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]
31.941033246	S1AP	82	id-UEContextRelease, UEContextReleaseComplete
42.851063506	S1AP/NAS-	210	id-initialUEMessage, Attach request, PDN connectivity request
42.852149210	S1AP/NAS-	106	SACK id-downlinkNASTransport, Attach reject (IMSI unknown in HSS)
42.859696404	S1AP	82	id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]

Figure 6: S1AP log messages of authentication failure: after five consecutive authentication failures, which takes about 40 seconds, the UE starts to search for a new eNodeB in other frequency channels.

Info
id-initialUEMessage, Attach request, PDN connectivity request
SACK id-downlinkNASTransport, Authentication request
id-uplinkNASTransport, Authentication response
SACK id-downlinkNASTransport, Security mode command
id-uplinkNASTransport, Security mode complete
SACK id-downlinkNASTransport, ESM information request
id-uplinkNASTransport, ESM information response
id-InitialContextSetup, InitialContextSetupRequest , Attach accept, Activate default EPS bearer con
id-InitialContextSetup, InitialContextSetupResponse
id-uplinkNASTransport, Attach complete, Activate default EPS bearer context accept
id-downlinkNASTransport, EMM information
id-UEContextReleaseRequest, UEContextReleaseRequest [RadioNetwork-cause=failure-in-radio-interface
SACK id-UEContextRelease, UEContextReleaseCommand [NAS-cause=normal-release]
id-UEContextRelease, UEContextReleaseComplete

Figure 7: S1AP log message of an RLF: this UE was *active*, but it becomes disconnected due to a sudden RLF.

eNodeB is not identified by the serving MME, the handover will eventually fail. Therefore, the handover procedure, even if caused by a malicious eNodeB, does not make a UE vulnerable to the CMAS spoofing attack.

As a consequence, the attacker first needs to disconnect the UE from its serving eNodeB. After the UE is released from the serving eNodeB, it will immediately try to attach to the strongest eNodeB, and thereafter, it can be attacked in the same way as idle mode UEs described in the section above.

When a communication error is detected on the established radio link between the UE and its serving eNodeB, it is referred to as a Radio Link Failure (RLF). The RLF can be detected by either the UE or eNodeB for various erroneous cases. A typical RLF is caused by reaching the maximum number of packet retransmissions in the Radio Link Control (RLC) layer of the LTE protocol stack. Jamming LTE signals can easily lead to an RLF in active UEs [34, 40].

Without any special jamming technique, however, a malicious eNodeB can jam the communication between a UE and its serving eNodeB simply by transmitting at a much higher power than the serving eNodeB. The malicious eNodeB overwhelms the serving eNodeB's transmissions and causes an RLF. This transmission must be on the same frequency channel used by the serving eNodeB. Figure 7 shows a UE releasing resources allocated by its serving eNodeB with the cause of 'failure-in-radio-interface-procedure'. This is a failure caused by an RLF. Once the RLF occurs, the radio connection of the UE releases. The UE attempts to attach to the higher-powered eNodeB and, thus, becomes vulnerable to spoofed CMAS messages just as an idle mode UE is.

3.3 Cases for CMAS Reception and Trust

With the LTE testbed, we performed a CMAS reception test. The results break down into three possible outcomes for CMAS alert reception, those being: the CMAS is not received, the CMAS is received and is known to be trustworthy, the CMAS is received and may be malicious. From those results we have identified three possible cases that determine whether the CMAS is received and is trustworthy. The cases and results are summarized in Table 2. Each case depends on where the UE currently is in the idle/active life-cycle. While testing each case, we continuously transmit the CMAS message once a second.

Simply put, if a UE is not listening to the eNodeB transmitting the CMAS message, the CMAS message will not be received by the UE. This is illustrated as the blue portion in Figure 8. It may seem obvious, but a necessary condition for the UE to receive a CMAS message is that it is tuned to the synchronization channels of the eNodeB that is transmitting the CMAS message. If the UE is listening to other frequency channels, or selects another cell which is not transmitting the CMAS message, then the message will not be delivered. We will not consider this scenario from now on.

Secure CMAS. In Case 1, the UE attaches to an eNodeB and is safely in the active state. To do this, the UE must be equipped with a valid Service Identity Module (SIM) card that is registered to that EPC. This case is the general scenario for phones receiving normal service from their provider. Because mutual authentication between the UE and the network has been successfully made, the UE can trust that the eNodeB is not malicious [6]. The CMAS reception is successful as we would expect, and we know that this CMAS message is trustworthy.

Unsecured CMAS. In Case 2, the UE has failed or is in the midst of failing to attach when the eNodeB transmits the CMAS message. The CMAS message will still be received by the UE; this is the crux of the vulnerability. In order to demonstrate this, we deleted the SIM information from the EPC so that the user authentication would be unsuccessful. The UE is now in the unsecured range between the idle and active states[4] due to the authentication failure. Even though the UE fails to reach the active state, we observe that the CMAS message is successfully received. This is because once the UE completes decoding SIB12 it delivers the contents to the application layer to be shown to the user. This is possible even after the authentication process has finally failed. Case 2 can lead the potential threat that *any malicious eNodeB can deliver fake CMAS messages while the UE is in between the eNodeB search and authentication procedures*. The red area in Figure 8 depicts this exploitable state of the UE.

Finally, in Case 3, the UE roams to an eNodeB which sends a CMAS message. To demonstrate this we removed the SIM card from the UE. No authentication is possible but the UE can make emergency calls such as 911. Even in this situation, we verified that the UE still receives the CMAS message which is potentially malicious.

[4] In the 3GPP standard [4], there exists a corresponding state model on EPS connection management (ECM) consisting of ECM-CONNECTED and ECM-IDLE states. The idle mode is defined to conserve power due to the UE's radio and network resources. Initially, a UE performs 'network attach' and then it can go to the idle mode by inactivity. For simplicity, we do not differentiate between 'network attach' and 'wake-up from idle', both of which have the same problem at this phase of the threat.

Table 2: Cases for CMAS reception and trust

Case	SIM Equipped	Auth. Success	CMAS Reception	Trustworthy
1	Yes	Yes	Yes	Yes
2	Yes	No	Yes	No
3	No	No	Yes	No

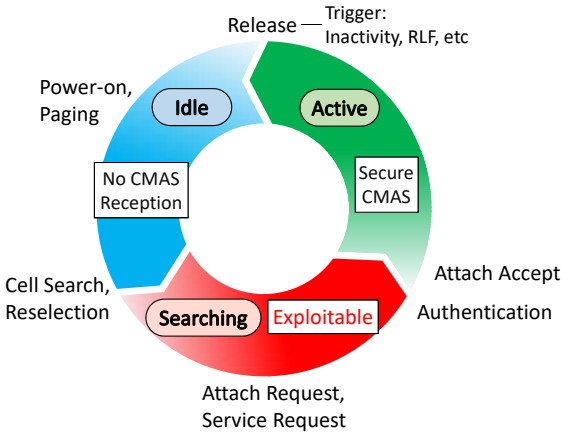


Figure 8: The Idle/Active life-cycle of a UE. The state of the UE continues counterclockwise around the chart. CMAS spoofing is possible while the UE performs an eNodeB search, prior to successful authentication with a trusted eNodeB.

As shown in Cases 2 and 3, CMAS spoofing can be done while the UE performs an eNodeB search, prior to successful authentication with a trusted eNodeB. These results are verified using 1 × JL620 COTS LTE small cell (no modification) [27], 1 × open source NextEPC (modified with the CBC) [37], and 9 different commercial LTE phones (Apple iPhone 8, X, and XS; Google Pixel 1; Huawei Nexus 6P; Motorola G5 Plus, and G6; Samsung Galaxy S7 Edge, and S8). Considering that the majority of UEs in cellular networks are in the idle state [18] and UEs often transition from the active to idle state due to an inactivity timer (around 10 seconds [25]), *almost all UEs are susceptible to this attack.*

3.4 Impact of Signal Strength

Here, we provide an analysis to estimate the expected number of UEs who are attacked by the methods described in §3.2 as a function of the difference in the received power strength between the malicious eNodeB and the serving eNodeB originally chosen in Eq. (1).

Let r_i be the RSRP of UE$_i$ from its strongest normal eNodeB, while ρ_i be the RSRP of UE$_i$ from the malicious eNodeB. Let δ_i be

$$\delta_i = r_i - \rho_i.$$

A UE that observes $\delta_i \leq 0$ has a higher or equal RSRP value from the malicious eNodeB than the serving eNodeB. Let $Pr(\delta_i \leq x)$ be the probability that δ_i is equal to or less than x, and $F(x)$ is drawn from the cumulative distribution function (CDF) of $Pr(\delta_i \leq x)$.

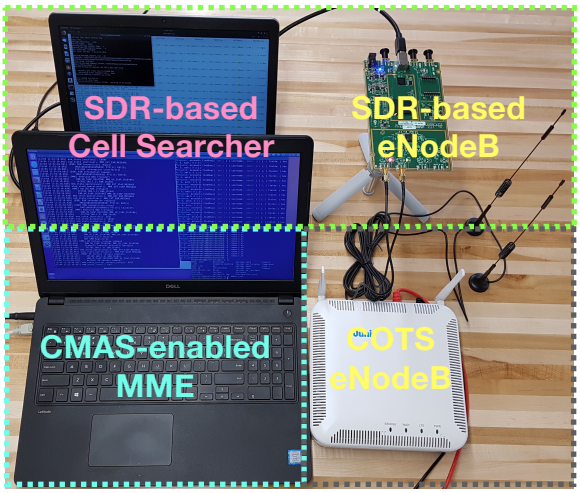


Figure 9: The Presidential Alert Spoofer scans for an eNodeB, gathers operator information, and sends a fake Presidential Alert to both idle and active UEs. The UEs may be FDD or TDD. This setup consists of one SDR device, one COTS LTE eNodeB, and 2 laptops.

Further, let N be the total number of UEs which is given by

$$N = N_{\text{idle}} + N_{\text{active}},$$

where N_{idle} and N_{active} are the number of idle UEs and active UEs. Let N^f be the random variable representing the number of UEs which successfully receive a *fake* alert from the malicious eNodeB. Similarly, N^f_{idle} and N^f_{active} represent the number of UEs receiving the fake alert in idle mode and active mode. Then, the expected number of N^f can be expressed as

$$E[N^f] = E[N^f_{\text{idle}}] + E[N^f_{\text{active}}] \quad (2)$$

$$= F(\alpha) \times N_{\text{idle}} + F(\beta) \times N_{\text{active}}, \quad (3)$$

where α is the RSRP difference required for an idle UE to select the malicious eNodeB, and β is the RSRP difference required to trigger the RLF for active UEs. From Eqs. (2) and (3), we can derive the following:

$$\alpha = F^{-1}\left(\frac{E[N^f_{\text{idle}}]}{N_{\text{idle}}}\right), \quad (4)$$

$$\beta = F^{-1}\left(\frac{E[N^f_{\text{active}}]}{N_{\text{active}}}\right). \quad (5)$$

Equations (4) and (5) show that the ratio of the number of UEs receiving a fake alert to the total number of UEs can determine the threshold values of α and β.

4 PROOF-OF-CONCEPT ATTACKS

In this section, we present the details of our *Presidential Alert Spoofer* system and describe how it works. Our system can be built with either an SDR device or a COTS eNodeB, and the list of hardware and software systems we used is summaried in Table 3.

Table 3: HW and SW systems used for implementation

System	Hardware	Software
Attack Preparation	BladeRF 2.0‡ ($500) USRP B210‡ ($1,300) Laptop (< $1,000)	OWL (modified) [16]
SDR-based Spoofer	BladeRF 2.0† ($500) USRP B210† ($1,300) Laptop (< $1,000)	srsLTE (modified) [24]
COTS eNodeB-based Spoofer	JL620ϟ (FDD) JLT621ϟ (TDD) Laptop (< $1,000)	NextEPC (modified) [37]

ϟ, †, ‡ The system requires only one among these.

Table 4: Spoofing attack preparation

Operator	EARFCN	Duplex	PCI	PLMN	RSRP
AT&T	5110	FDD	415	310410	-100
Sprint	41374	TDD	265	310120	-102
T-Mobile	5035	FDD	312	310260	-120
Verizon	5230	FDD	229	311480	-105

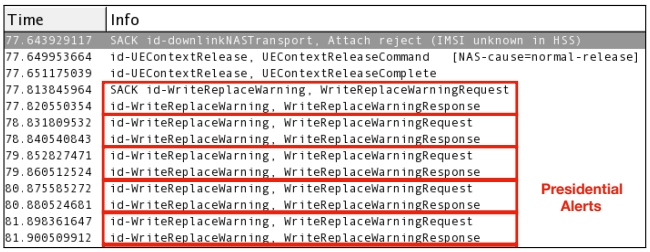


Figure 10: S1AP log messages: our modification on NextEPC provides an interface to inject a user-defined Presidential Alert.

Attack preparation. Our Presidential Alert Spoofer must first identify the existing eNodeBs in a given licensed frequency band. Each eNodeB can be uniquely identified at a given geographical position by the pair of 'E-UTRA Absolute Radio Frequency Channel Number (EARFCN)' and 'Physical Cell ID (PCI)'. For each EARFCN, our Spoofer finds the eNodeB, and associated PCI, of which the RSRP is the strongest. Once the existing eNodeBs are listed, the Public Land Mobile Network (PLMN) information of each eNodeB is collected. Every LTE network has its own PLMN, which contains a three digit country code and two or three digits to identify the provider. The PLMN is periodically broadcast by the eNodeB in the SIB1 message making it possible to passively collect all of the observable PLMNs within receiving range. To launch an attack, our Presidential Alert Spoofer uses the same PLMN as an existing eNodeB such that the UEs will select our Spoofer during a eNodeB search. We use the OWL software [16] with an SDR device (USRP B210 [20] and BladeRF 2.0 [39]), to gather all the PLMNs. Table 4 is the attack preparation results measured in our lab across the top four US LTE operators. For each EARFCN, our Spoofer's eNodeB tunes its radio to the same frequency with the same PLMN identifier, and starts to transmit a Presidential Alert continuously which contains a custom (attack) message.

Figure 11: Receiving multiple fake Presidential Alerts using a Samsung Galaxy S8(left), and an Apple iPhone X(right).

Attack execution with an SDR device. We implemented the Spoofer using a USRP B210 and BladeRF to attack Frequency Division Duplex (FDD) systems. With an SDR, we can change the transmission frequency easily to target every cellular band. We added SIB12 support to the open source eNodeB software [24] to transmit CMAS messages. Since the attack can last about 42 seconds (described in §3.2) and we can send a CMAS message every 160 msec and a victim UE may receive up to 262 SIB12 transmissions.

Attack execution with a COTS eNodeB. We use a COTS eNodeB (Juni JLT-621 [27]) to target Time Division Duplex (TDD) systems. Our modification of NextEPC provides an interface to inject a user-defined Presidential Alert that broadcasts each second as shown in Figure 10. With this configuration, a victim UE may receive up to 42 transmissions of SIB12 from the COTS eNodeB. Any commercial LTE FDD/TDD eNodeB hardware can perform this attack, which may play a key role if an attacker wants to control multiple malicious eNodeBs in a coordinated manner.

Attack verification. In our lab environment, we verified that the fake Presidential Alert sent by our SDR-based Spoofer was successfully shown in the FDD phones of AT&T, T-Mobile, and Verizon[5]. With a TDD Sprint phone, we verified that our COTS eNodeB-based Spoofer also works successfully[6]. Testing was performed on the nine different mobile phones listed in section §3.3 (two of which are shown in Figure 11). The detailed conditions regarding the attack will be described in §5.

Affected devices. Through discussions with 3GPP [1] of the SIB12 vulnerability described in §3.2, it became clear that the lack of authentication was a design choice by 3GPP, rather than an oversight. This design provides the best possible coverage for legitimate emergency alerts, but the trade-off leaves every phone vulnerable to spoofed alerts. As a consequence, all modem chipsets that fully comply with the 3GPP standards show the same behavior: the fake Presidential Alert is received without authentication. Once the

[5]Note that emitting over-the-air signals on a licensed band spectrum is illegal. Our experiments are carried out with proper RF shielding.
[6]At the moment, an SDR-based LTE TDD system implementation is not available.

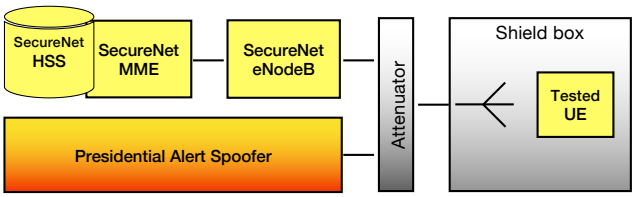


Figure 12: The testbed setup for evaluating the attack success rate. The transmission power levels of the SecureNet eNodeB and the Presidential Alert Spoofer can be controlled independently.

LTE modem of the UE receives the fake alert, the operating system[7] will display the alert to the user. Since our attack verification tests included many Android and iOS phones, we conclude that most (presumably all) LTE phones will be affected by the attack, regardless of the phone's vendor or model[8].

5 EVALUATION

Now, we evaluate the attack performance of our Presidential Alert Spoofer system. First, we show that the attack success rate depends on the spoofing signal strength. Second, we take propagation measurements of the AT&T LTE network and our SDR eNodeB in both indoor and outdoor environments. With these propagation measurements and the known success rate of the attack, we then evaluate the attack's coverage.

5.1 Experimental Setting

Figure 12 illustrates our experimental testbed setup, which consists of an EPC and eNodeB for a normal LTE system, a malicious eNodeB for spoofing, and cell phones for victim UEs. A signal attenuator receives the broadcast signals from two different sources and delivers the combined signal to a UE in a shielded box. We built an LTE test network, with an EPC and eNodeB, named SecureNet that assumes the role of the user's original network. On the other hand, the malicious eNodeB, part of the Presidential Alert Spoofer, is installed solely without any LTE core support. By using the signal attenuator, the signal power received at the UE can be precisely controlled for various practical scenarios. Within the experiments, all UE measurements were gathered with the Samsung Galaxy S8 and Motorola G6.

5.2 Success Rate

We evaluate the success rate of the Presidential Alert Spoofer as a function of α (or β), which is the RSRP difference between the SecureNet eNodeB and Presidential Alert Spoofer for an idle UE (β for an active UE). We first attach the UE to SecureNet. For the idle UE case, we wait for the UE to enter the idle mode due to inactivity. The Spoofer broadcasts each new Presidential Alert message with a new Serial Number and different message content (described in Table 1). In doing so we can determine whether each Presidential Alert is successfully received[9] and at what power configuration of α or β. We conducted 20 experimental trials for each value of α (or β) ranging from 0 dB to -25 dB.

The Spoofer may elect to use a different PCI than that of the serving eNodeB, appearing to be a new eNodeB. Or, the Spoofer may use the same PCI, appearing to be the existing eNodeB and interfering with the existing eNodeB's PHY-layer control channel information [51]. This decision has different impacts on the performance of the spoofing attack depending on the UE state (*idle* or *active*).

Figure 13 shows the empirical cumulative distribution function (CDF) of successful receptions of fake alerts as a function of α for idle UEs. When the Spoofer uses a different PCI and the received signal strength from the Spoofer is higher than that from SecureNet ($\alpha < 0$), the idle UE will consider the Spoofer as a new serving eNodeB by Eq. (1). Our experimental results verify this expectation; 50% of idle UEs can receive a fake message even at $\alpha = -1$, more than 90% of idle UEs can receive a fake message when $\alpha \leq -6$.

However, if the same PCI is used, the attack performance is significantly degraded. Because the PCI is used to generate cell-specific reference signals [7], using the same PCI value will cause channel estimation errors at the UE due to collisions from the two transmitters. This, in turn leads to more decoding errors when receiving the SIBs. As a result, using the same PCI requires much higher attack power, as no UE is affected when α is greater than -12 dB. With $\alpha \leq -17$, 90% of idle UEs can still be attacked.

Figure 14 shows the CDF of successful fake message receptions as a function of β (*i.e.,* forcing disconnect) for active UEs. When the Spoofer uses a different PCI and the received signal strength from the Spoofer is higher than that from the SecureNet eNodeB, the active UE will start to consider the Spoofer as a target eNodeB for a handover, as described in §3.3. Because the Spoofer is not identified by SecureNet, a handover cannot be performed. Instead, we observed a RLF will occur when $\beta \leq -10$, which eventually leads to the reception of a fake alert. 90% of active UEs can receive a fake message when $\beta \leq -20$ assuming that a different PCI value is used for the Spoofer. Unlike the idle UE case, using the same PCI value results in higher decoding errors (and more RLFs) at a receiver. Thus, it shows better attack performance; 90% of receptions are successful with $\beta \leq -13$.

5.3 Attack Coverage

To determine the attack coverage, we performed measurements of RSRP over various distances between the UE and the eNodeB. We transmitted with a COTS eNodeB in the Educational Broadcast Service (EBS) band and at 0.1 Watt transmit power[10]. The measurement was done with 70m of indoor space and 120m of outdoor space. From the measurements, we observed that the RSRP tends to decrease as the distance increases and the indoor RSRP is higher than that of outdoor. This is due to the fact that multiple signals from indoor reflections along with a line-of-sight signal can provide receiver diversity, thus exhibiting a smaller path-loss exponent than an outdoor environment [23, 43].

From Figures 13 and 14, we show the following results: If the malicious eNodeB's PCI is different from SecureNet and the idle UE's

[7] According to a mobile market report, the worldwide market share of Android is 75.39% and that of iOS is 22.35% on March 2019 [41].
[8] Since much of the LTE public warning system is inherited from 2G/3G, a similar attack is also possible in 2G/3G. Unfortunately, this is out-of-scope for this paper.
[9] CMAS messages with repeated content may be ignored by the UE.
[10] We have the spectrum license for the EBS band in our campus area.

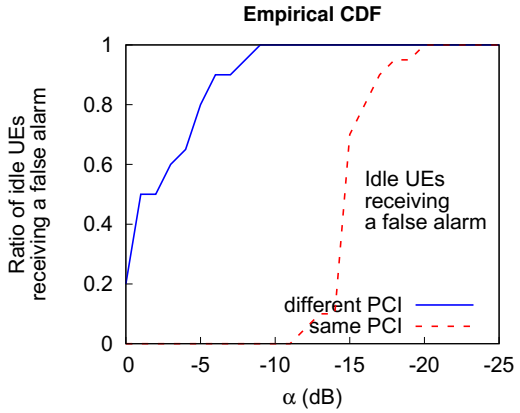


Figure 13: The CDF as a function of α for only idle UEs. Because eNodeB reselection happens when idle UEs wake up, the spoofing attack performs better when using a different PCI.

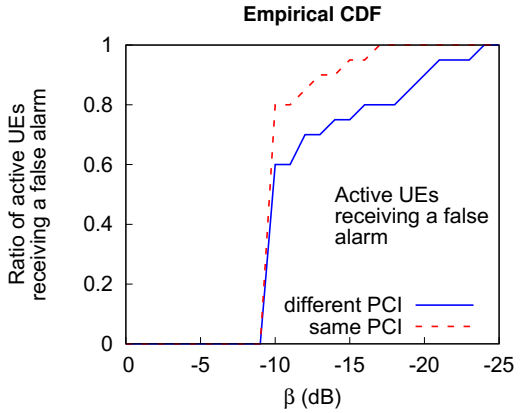
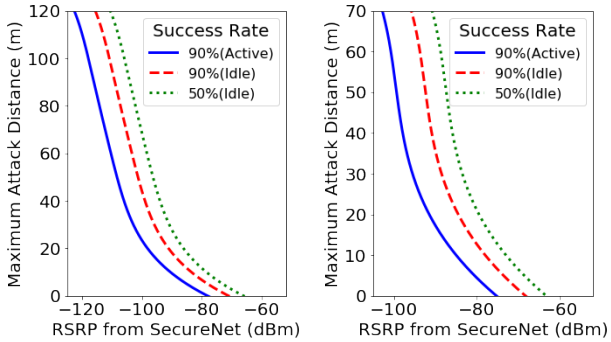


Figure 14: The CDF as a function of β for only active UEs. Using the same PCI leads to more decoding errors observed by the UE. This results in a slightly more effective attack.

RSRP from the malicious eNodeB is 6dB higher than SecureNet, the attack will be successful with 90% probability. When the RSRP difference is 1dB, the attack to the idle UE is successful with 50% probability. On the other hand, in the case of the same PCI configuration, the active UE will be attacked with 90% probability when the RSRP difference is 13dB.

In order to figure out the maximum attack distance, polynomial regression was applied to the measured values for both outdoor and indoor conditions. Figure 15 shows the relationship between the RSRP from SecureNet and the attack distance by the malicious eNodeB in order to achieve a 90% success rate for active UEs, a 90% success rate for idle UEs, and a 50% success rate for idle UEs. For example, in the outdoor case shown in Figure 15a, when the RSRP of an active UE from SecureNet is −100dBm, the attack can reach up to 23.4m centered at the malicious eNodeB of 0.1 Watt power with 90% probability. For the idle UE, when the RSRP is −100dBm, it can be attacked up to 44.1m and 68.5m away from the malicious eNodeB



(a) Outdoor **(b) Indoor**

Figure 15: The maximum attack distance between the UE and the malicious eNodeB according to the RSRP from the SecureNet eNodeB. The transmit power of the malicious eNodeB is 0.1 Watt.

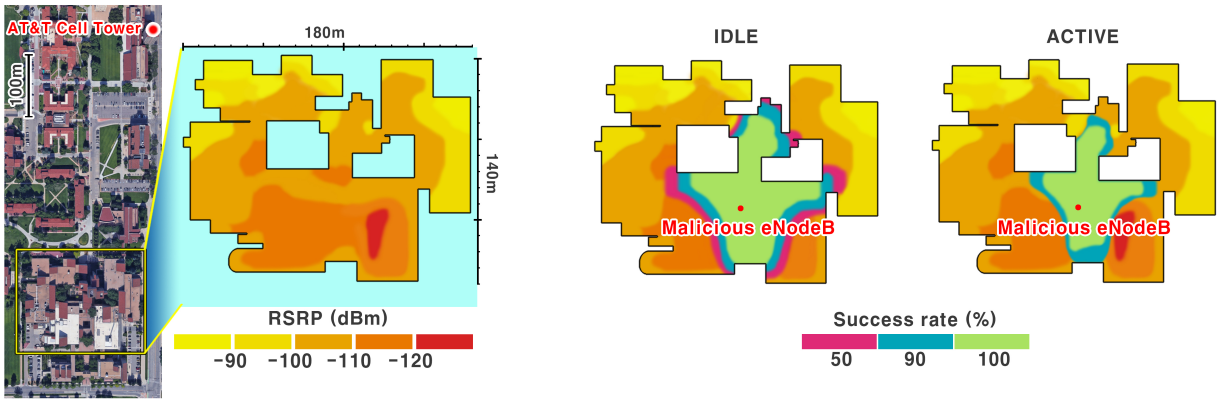
of 0.1 Watt power with 90% and 50% probability, respectively. Figure 15b shows the result in open space indoors, where the maximum attack distance can be obtained similarly with the outdoor case. We observe that the attack distance can be much greater than the outdoor case due to the smaller attenuation characteristic of indoor buildings. For the active UE with −100dBm RSRP, the attack radius is 55.2m with a 90% probability of success.

5.4 Practical Scenarios: Indoor and Outdoor

Since we do not use our spoofer outside of a shield box, we cannot directly measure its effect on a large number of people. To evaluate the attack coverage according to its success rate, we use actual RSRP measurements in the indoor and outdoor environments.

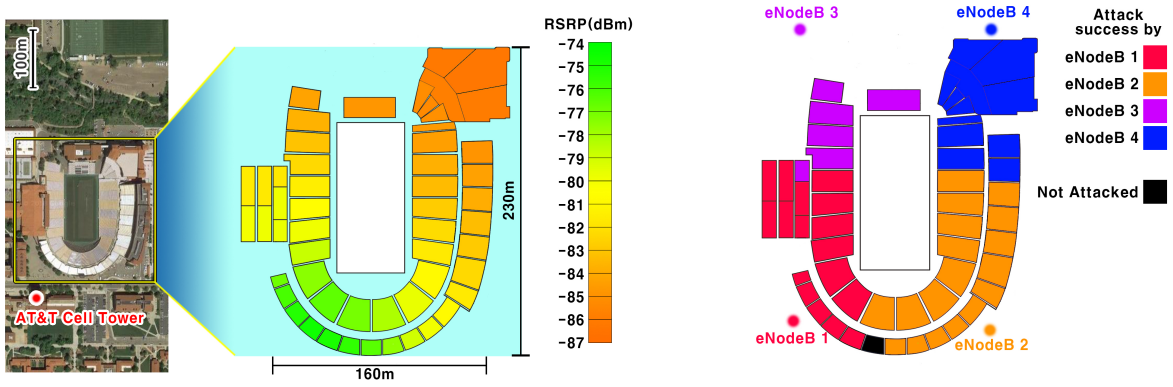
Indoor Attack. We placed our malicious eNodeB inside a campus building and measured the RSRP of a dummy LTE signal (containing no CMAS message) in the EBS band with 0.1 Watt transmit power. We also measured the RSRP of a nearby AT&T eNodeB, shown in Figure 16a. The RSRP does not attenuate consistently due to various obstacles, but generally the RSRP tends to decrease as the distance from the AT&T eNodeB increases. We compared the two RSRPs throughout the building and indicated the attack coverage using measurements obtained from §5.2, depicted in Figure 16b. As a result, in a building with a total area of about 16,859m^2, for idle UEs, the coverage for a 90% success rate was about 4,435m^2, whereas for active UEs, the coverage for a 90% success rate was about 2,955m^2.

Outdoor Attack. Without access to outdoor LTE equipment, we simulate the RSRPs of the spoofing eNodeB and the AT&T eNodeB with the NS-3 v3.29 network simulator [38]. For a scenario we assume a football game in which a large number of people are gathered in a restricted region. A group of attackers send fake alerts to the spectators inside the football stadium. We measured the RSRP of an actual AT&T eNodeB around the perimeter of our campus' football stadium, shown in Figure 17. We used the simulator to estimate the RSRPs at the centers of each section in the stadium (Figure 17a). We simulated spoofers in four corners around the stadium, near but still outside of the ticketed area. Figure 17b shows



(a) Indoor RSRP measurements of AT&T (b) Attack coverage

Figure 16: The indoor attack simulation: (a) The satellite image of the Engineering Center at the University of Colorado Boulder shows the nearest AT&T eNodeB. The graph shows the indoor RSRP distribution of that eNodeB. (b) The attack coverage for idle and active UEs are shown when a 1×0.1 Watt malicious eNodeB is used.



(a) Outdoor RSRP measurements of AT&T (b) Attack coverage

Figure 17: The outdoor attack simulation: (a) The satellite image of Folsom Field at the University of Colorado Boulder shows the location of the AT&T eNodeB. The stadium graph represents the RSRP distribution of the eNodeB measured at the center of each section. (b) When 4×1-Watt malicious eNodeBs are located outside the four corners of the stadium, the simulated attack coverage hits all but one section. This means that 49,300 among the total 50,000 seats are hit with the attack, which itself has a 90% success rate.

which malicious eNodeB with a 1-Watt transmit power attacked each section. We observe that all sections, except one, are attacked by the malicious eNodeBs. This means that 49,300 among the total 50,000 seats will be hit with the attack, which itself has a 90% success rate, given that all UEs are in the idle state.

6 POTENTIAL DEFENSES

Defending against CMAS spoofing attacks requires careful consideration of several challenges. First, updates to the CMAS architecture could require expensive changes by cell phone manufacturers, operating system developers, government bodies, and cellular carriers. Coordinating such an effort would be difficult due to the fragmented nature of the network. Furthermore, updates must still support outdated devices, both on the user (UE) and infrastructure (eNodeB) side, as it could take years to replace or update old equipment already in use. In addition, any comprehensive defense must trade-off the protections provided with the availability of the system: if users cannot receive valid alerts due to complex protections, it may be more hazardous than if we continued to use the existing (but vulnerable) system.

With these challenges in mind, we discuss two potential defenses: first, adding digital signatures to alerts, and second, client-side software solutions ignoring unsecured CMAS alerts and attempting to detect false alerts by fingerprinting characteristics of legitimate eNodeBs. We stress that neither of these defenses offer a magic solution, but instead hope they provide starting points for network operators and cell phone manufacturers to continue discussions.

6.1 End-to-End Approach: Digital Signature

We first consider digitally signing SIB12 messages to prevent spoofed messages, as discussed by 3GPP [1]. While conceptually simple, adding signatures is difficult for several reasons: First, operators and devices must agree on the key or keys that will be used to sign and validate messages. Second, devices must decide what to do if they receive a signature from an unknown key or an unsigned (e.g., legacy CMAS) message from the network. Finally, signatures must fit within the practical constraints of the network.

For key management, we leverage suggestions from 3GPP discussions [1], which suggest using the Non-Access Stratum (NAS) to send authenticated messages to the device. Because NAS provides message integrity between the eNodeB and UE (mediated by pre-shared keys in the UE SIM card), messages received in this way cannot be spoofed by a (physically) nearby adversary. However, sending alerts over this channel would limit their reception to devices that had established a NAS session. Instead, we recommend using this authenticated channel to send updates to the set of (public) keys that a device should trust. These keys could correspond to private ones held by the variety of local and national Cell Broadcast Entities (CBEs) that are authorized to issue such alerts, such as local law enforcement for AMBER alerts or the President for Presidential Alerts. If desired, these keys could be included in certificates that describe their scope, and be signed by central entities such as the carrier or traditional Certificate Authorities.

In the immediate term, networks or CBEs that do not support digital signatures will continue to send legacy ones, and it is also possible for devices to receive alerts from CBEs that they do not have the corresponding key for. If the device refuses to display such messages, it may leave the user in the dark about potentially legitimate alerts, while protecting them from false ones. If the alert is still displayed, then the addition of signatures provides no benefit. One solution is to allow legacy messages to be displayed until enough of the network has deployed signatures. Another approach could be to allow the user to see unverified alerts but with a warning indicating that the message could not be authenticated. We caution that such warnings must be designed carefully to appropriately inform the user of the risks; previous research on the usability of HTTPS warning messages in browsers may be a useful starting point [22].

Finally, signatures must fit within the constraints of the network without adding significant overhead. CMAS alert messages are sent in an 82-byte field of the SIB12 message [3]. Even adding a short signature to this could limit the length of a useful message. Instead, we propose using additional pages in the existing SIB12 message to send a corresponding signature of a CMAS alert message, allowing for up to 82-bytes for the signature or any metadata (such as a time stamp or sequence number to prevent replay attacks). This is more than enough for a 64-byte ECDSA signature, though if smaller signatures are desired, BLS signatures could also be used to reduce the size in half [15]. As of May 2019, the FCC mandated that commercial mobile service operators must support alert messages that contain up to 360 characters of alphanumeric text on their LTE and future networks in the US [21]. As a result, adding a digital signature becomes applicable for the existing and future wireless emergency alert systems.

Signature implementation. To verify the feasibility of this scheme, we implement a simple digital signature for the Presidential Alert. We used the ed25519 digital signature [14] to sign a 4-byte time stamp along with the CMAS alert message (68 bytes overhead in total), and defined a new Message Identifier (see Table 1) which indicates that a digital signature is added. Once a signed message is received, the alert message can be displayed after verifying its signature. We implemented this by modifying the open source UE implementation running on a USRP B210 [24]. The resulting UE is not affected by the spoofing attack because it only accepts signed messages.

Suggestion: We recommend that the digital signature only be required for Presidential Alerts. Unlike AMBER and other alerts, the Presidential Alert has only one originator, making the key distribution much simpler. The private key need only be known by a single entity, rather than a large number of CBEs. In addition, Presidential Alerts are the only alerts that cannot be disabled by users, potentially making them a larger target for abuse.

6.2 Client-only Approaches

To enhance the resiliency of LTE networks against such attacks, we also consider a network profiling technique which can be solely implemented on the UE without modifications to the network.

Accepting only secure CMAS. Since the Presidential Alert Spoofer exploits the unauthenticated CMAS delivery in LTE, a UE may ignore all received SIB12 messages before it successfully authenticates the network. It could be implemented either 1) by the LTE modem firmware, or 2) by the operating system (e.g., Android and iOS) of the UE. As the LTE modem firmware maintains its own state including the UE authentication, it could decide whether each SIB12 reception is secure or not. Similarly, the OS could keep track of the authenticated state of the network, and it may choose not show an unsecured CMAS messages to the user. The major disadvantage of this approach is that the accessibility of trustworthy CMAS messages will be limited. Secure CMAS alert messages may not be received by:

- those who do not have a valid subscription from a home or visited network service provider,
- those who are temporarily in the unauthenticated state due to the UE attaching, idle exit, handovers, etc.

Nevertheless, the risk of the CMAS spoofing attacks could be significantly mitigated by this approach.

LTE eNodeB fingerprinting. Since every eNodeB broadcasts its network configuration through SIB messages, a UE may leverage such information to construct a fingerprint of the eNodeB during its normal operation. In particular, we observe that each eNodeB uses different SIB contents and patterns. The transmission patterns of SIB messages in terms of message types and periodicity can vary, and we observed that each operator exhibits its own transmission pattern[11]. By monitoring the SIB transmission pattern of a certain eNodeB, we can link it to a specific operator that runs it. In addition, channel quality (e.g., RSRP) and cell load information can be measured by the UE [31, 50] and combined with location information to additionally classify the validity of the attached eNodeB. While an

[11] Unique SIB transmission patterns across top four US LTE carriers have been only verified in the Boulder area of Colorado, USA.

attacker may be able to mimic these signals, it may nonetheless be a significant engineering hurdle to perform a coordinated attack against a geographically diverse set of users.

Providing an eNodeB's location. As a related approach, we can leverage the received signal strength (RSS) at the UE to determine if the eNodeB to which we are connected is a feasible distance away. Using a widely used path-loss model [23], we can estimate the distance to the eNodeB using the RSS value. Then compare this with the location provided by an Internet database [17] to determine whether the alert could have come from a trusted eNodeB or not.

We emphasize that client-only solutions are not as robust as a digital signature-based one, as a motivated attacker may still be able to spoof messages to some users. However, it has the advantage of not requiring any modifications on the network side, and can be implemented entirely through software updates on the UE, offering a potentially attractive short-term stopgap.

7 RELATED WORK

LTE security leaks: Existing LTE threats can be distilled into three general attacks: jamming, sniffing, and spoofing. LTE jamming attacks can be made more efficient by pairing with sniffing attacks, which collect network configurations to pinpoint the most vital spots of the signal [34]. Within the sniffing and spoofing genres, there exists IMSI catching, a notorious issue in cellular networks [48]. When a phone authenticates itself to the network, the SIM card has to reveal its identity over an insecure plaintext transmission. This unique identifier (i.e., IMSI) can be intercepted by adversaries that mount a passive or active attack, thus leading to a violation of privacy and traceability. The work in [30] proposes a new cell selection procedure that mitigates denial of service attacks but does not address the UE's vulnerability to fake CMAS messages. With the advent of open source LTE software, the authors of [40] demonstrate inexpensive and practical attacks about fine-grained location leaks and denial of service of the LTE device by implementing the attack platform.

IMS-related threats: The 4G LTE network requires IP Multimedia Subsystem (IMS) for enabling VoLTE (Voice over LTE). In this context, apart from traditional 2G/3G networks, recent research [28, 32] identified several vulnerabilities of the VoLTE service and further demonstrated that the adversary can easily gain free data channels by delivering packets via the voice channel. Also, [44] disclosed the insecurity of IMS-based SMS by devising several SMS attacks such as SMS spoofing and spamming. [19] propose an effective and practical callee-only solution against caller ID spoofing. Given an incoming call, it leverages a callback session and its associated call signaling observed at the phone to infer the call state of the other party. It further compares with the anticipated call state, thus verifying whether the incoming call comes from the originator.

LTE protocol verification: MobileInsight [33] is a software tool that collects, analyzes, and exploits runtime cellular network information, which exposes protocol messages on both the control and data planes from the 3G/4G chipset. As one application, it can detect security loopholes that 4G mobility management protocol configures the UE to not encrypt the signaling message during an attach procedure. Several works have exposed the risk of potential spoofing attacks on LTE emergency alerts [1, 26, 29]. LTEInspector [26] modeled the LTE control plane procedures including attach, detach, and paging to identify LTE design problems. LTEFuzz [29] generated dynamic test cases to investigate the security aspects of LTE control plane procedures. Prior to these studies, the investigation of spoofing attacks on 3GPP public warning system was also reported [1]. In contrast to our verification of the CMAS attack vulnerability, which has been done in systematic and extensive way, none of these validated such an attack in practice. The contributions of our work are as follows: 1) the detailed analysis of CMAS spoofing attack, 2) design and implementation of such systems by using the COTS eNodeBs as well as SDRs, 3) possible counter measures, and most importantly 4) the impact of such attacks (indoor and outdoor environments).

5G security analysis: As 5G networks will accommodate new concepts such as edge computing and network slicing, their security challenges can become more threatening than 4G LTE [12]. For example, [13] provides the first comprehensive model of the 5G authentication protocol via a systematic security evaluation, makes explicit recommendations, and proposes provably secure fixes for the traceability attack. More importantly, according to the 5G RRC (Radio Resource Control) standard [10], SIB12 is likely to be deployed in 5G systems, so CMAS spoofing will be one of the critical security threats well into the future.

8 CONCLUSION

In this paper, we have identified security vulnerabilities of WEA over commercial LTE networks and found that a spoofing attack with fake alerts can be done very easily. Specifically, we presented our threat analysis on the spoofing attack, and implemented an effective attack system using COTS SDR hardware and open source LTE software. Our extensive experimentation confirmed that the CMAS spoofing attack can succeed in all tested smartphones in the top four cellular carriers in the US. Further, we have discussed several defenses, from which we believe that completely fixing this problem will require a large collaborative effort between carriers, government stakeholders, and cell phone manufacturers.

ACKNOWLEDGMENTS

We would like to thank the MobiSys reviewers and our shepherd Stefan Saroiu for their feedback on earlier versions of this paper. This work was supported by the NSF under Grants CNS-1525435 and CNS-1738097.

REFERENCES

[1] 3GPP TR 33.969. 2014. Technical Specification Group Services and System Aspects; Study on Security aspects of Public Warning System (PWS) (Release 15). http://www.3gpp.org/DynaReport/33969.htm. (2014).
[2] 3GPP TS 23.038. 2018. Technical Specification Group Core Network and Terminals; Alphabets and language-specific information (Release 15). http://www.3gpp.org/dynareport/23038.htm. (2018).
[3] 3GPP TS 23.041. 2018. Technical Specification Group Core Network and Terminals; Technical realization of Cell Broadcast Service (CBS) (Release 15). http://www.3gpp.org/dynareport/23041.htm. (2018).
[4] 3GPP TS 23.401. 2018. Technical Specification Group Services and System Aspects; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (Release 15). http://www.3gpp.org/dynareport/23401.htm. (2018).
[5] 3GPP TS 29.168. 2018. Technical Specification Group Core Network and Terminals; Cell Broadcast Centre interfaces with the Evolved Packet Core (Release 15).

http://www.3gpp.org/dynareport/29168.htm. (2018).
[6] 3GPP TS 33.401. 2018. Technical Specification Group Services and System Aspects; Security architecture (Release 15). http://www.3gpp.org/dynareport/33401.htm. (2018).
[7] 3GPP TS 36.211. 2018. Technical Specification Group Radio Access Network; Physical channels and modulation (Release 12). http://www.3gpp.org/dynareport/36211.htm. (2018).
[8] 3GPP TS 36.331. 2018. Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC) (Release 15). http://www.3gpp.org/dynareport/36331.htm. (2018).
[9] 3GPP TS 36.413. 2018. Technical Specification Group Radio Access Network; S1 Application Protocol (S1AP) (Release 15). http://www.3gpp.org/dynareport/36413.htm. (2018).
[10] 3GPP TS 38.331. 2018. Technical Specification Group Radio Access Network; NR; Radio Resource Control (RRC) (Release 15). http://www.3gpp.org/dynareport/38331.htm. (2018).
[11] 5G Americas. 2018. Public Warning Systems in the Americas. (2018). https://goo.gl/yZ4R4L.
[12] I. Ahmad, T. Kumar, M. Liyanage, J. Okwuibe, M. Ylianttila, and A. Gurtov. 2018. Overview of 5G Security Challenges and Solutions. *IEEE Communications Standards Magazine* 2, 1 (March 2018), 36–43.
[13] David Basin, Jannik Dreier, Lucca Hirschi, Saša Radomirovic, Ralf Sasse, and Vincent Stettler. 2018. A Formal Analysis of 5G Authentication. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18)*.
[14] Daniel J Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang. 2012. High-speed high-security signatures. *Journal of Cryptographic Engineering* 2, 2 (2012), 77–89.
[15] Dan Boneh, Ben Lynn, and Hovav Shacham. 2001. Short signatures from the Weil pairing. In *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 514–532.
[16] Nicola Bui and Joerg Widmer. 2016. OWL: a Reliable Online Watcher for LTE Control Channel Measurements. In *ACM All Things Cellular (MobiCom Workshop)*.
[17] CellMapper. 2018. Cellular Coverage and Tower Map. https://www.cellmapper.net/. (2018).
[18] Xiaomeng Chen, Abhilash Jindal, Ning Ding, Yu Charlie Hu, Maruti Gupta, and Rath Vannithamby. 2015. Smartphone Background Activities in the Wild: Origin, Energy Drain, and Optimization. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*.
[19] Haotian Deng, Weicheng Wang, and Chunyi Peng. 2018. CEIVE: Combating Caller ID Spoofing on 4G Mobile Phones Via Callee-Only Inference and Verification. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom '18)*.
[20] Ettus Research. 2018. USRP B210. https://www.ettus.com/product/details/UB210-KIT. (2018).
[21] Federal Communications Commission (FCC). 2016. Wireless Emergency Alerts; Amendments to Rules Regarding the Emergency Alert System. https://www.gpo.gov/fdsys/pkg/FR-2016-11-01/pdf/2016-26120.pdf. (2016).
[22] Adrienne Porter Felt, Robert W. Reeder, Alex Ainslie, Helen Harris, Max Walker, Christopher Thompson, Mustafa Emre Acer, Elisabeth Morant, and Sunny Consolvo. 2016. Rethinking Connection Security Indicators. In *Proceedings of the Twelfth USENIX Conference on Usable Privacy and Security (SOUPS'16)*.
[23] Andrea Goldsmith. 2005. *Wireless Communications*. Cambridge University Press.
[24] Ismael Gomez-Miguelez, Andres Garcia-Saavedra, Paul D. Sutton, Pablo Serrano, Cristina Cano, and Douglas J. Leith. 2016. srsLTE: An Open-Source Platform for LTE Evolution and Experimentation. In *ACM WiNTECH (MobiCom Workshop)*.
[25] Junxian Huang, Feng Qian, Alexandre Gerber, Z. Morley Mao, Subhabrata Sen, and Oliver Spatscheck. 2012. A Close Examination of Performance and Power Characteristics of 4G LTE Networks. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys '12)*.
[26] Syed Rafiul Hussain, Omar Chowdhury, Shagufta Mehnaz, and Elisa Bertino. 2018. LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE. In *Proceedings of the Network and Distributed System Security Symposium (NDSS '18)*.
[27] Juni. 2017. Enterprise Small Cell JL620. http://www.juniglobal.com/product/jl-620fdd-jlt-621tdd/. (2017).
[28] Hongil Kim, Dongkwan Kim, Minhee Kwon, Hyungseok Han, Yeongjin Jang, Dongsu Han, Taesoo Kim, and Yongdae Kim. 2015. Breaking and Fixing VoLTE: Exploiting Hidden Data Channels and Mis-implementations. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS '15)*.
[29] Hongil Kim, Jiho Lee, Eunkyu Lee, and Yongdae Kim. 2019. Touching the Untouchables: Dynamic Security Analysis of the LTE Control Plane. In *40th IEEE Symposium on Security and Privacy*.
[30] M. Labib, V. Marojevic, J. H. Reed, and A. I. Zaghloul. 2017. Enhancing the Robustness of LTE Systems: Analysis and Evolution of the Cell Selection Process. *IEEE Communications Magazine* 55, 2 (February 2017), 208–215.
[31] Jihoon Lee, Jinsung Lee, Youngbin Im, Sandesh Dhawaskar Sathyanarayana, Parisa Rahimzadeh, Xiaoxi Zhang, Max Hollingsworth, Carlee Joe-Wong, Dirk Grunwald, and Sangtae Ha. 2019. CASTLE over the Air: DistributedScheduling for Cellular Data Transmissions. In *The 17th Annual InternationalConference on Mobile Systems, Applications, and Services (MobiSys '19)*.
[32] Chi-Yu Li, Guan-Hua Tu, Chunyi Peng, Zengwen Yuan, Yuanjie Li, Songwu Lu, and Xinbing Wang. 2015. Insecurity of Voice Solution VoLTE in LTE Mobile Networks. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS '15)*.
[33] Yuanjie Li, Chunyi Peng, Zengwen Yuan, Jiayao Li, Haotian Deng, and Tao Wang. 2016. Mobileinsight: Extracting and Analyzing Cellular Network Information on Smartphones. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom '16)*.
[34] M. Lichtman, R. P. Jover, M. Labib, R. Rao, V. Marojevic, and J. H. Reed. 2016. LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation. *IEEE Communications Magazine* 54, 4 (April 2016), 54–61.
[35] Collin Mulliner, Nico Golde, and Jean-Pierre Seifert. 2011. SMS of death: from analyzing to attacking mobile phones on a large scale. In *Proceedings of the 20th USENIX conference on Security*.
[36] National Public Radio. 2018. Officials Assess Response To Camp Fire In Northern California. https://goo.gl/iF12Vo. (2018).
[37] NextEPC Inc. 2019. Open source implementation of LTE EPC. https://www.nextepc.com/. (2019).
[38] Nsnam. 2018. NS-3: A discrete-event network simulator for internet systems. https://www.nsnam.org. (2018).
[39] Nuand. 2018. bladeRF 2.0 micro xA4. https://www.nuand.com/product/bladerf-xa4/. (2018).
[40] A. Shaik, J. Seifert, R. Borgaonkar, N. Asokan, and V. Niemi. 2016. Practical attacks against privacy and availability in 4G/LTE mobile communication systems. In *23nd Annual Network and Distributed System Security Symposium, NDSS*.
[41] StatCounter. 2019. Mobile Operating System Market Share Worldwide. http://gs.statcounter.com/os-market-share/mobile/worldwid. (2019).
[42] The Washington Post. 2018. Cellphone users nationwide just received a 'Presidential Alert.' Here's what to know. https://goo.gl/KRfDjf. (2018).
[43] Michael Tsai. 2011. Path-loss and Shadowing (Large-scale Fading). https://goo.gl/QD7wwn. (2011).
[44] Guan-Hua Tu, Chi-Yu Li, Chunyi Peng, Yuanjie Li, and Songwu Lu. 2016. New Security Threats Caused by IMS-based SMS Service in 4G LTE Networks. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*.
[45] U.S. Department of Homeland Security (DHS). 2015. Wireless Emergency Alerts (WEA) CMSP Cybersecurity Guidelines. https://goo.gl/X9X3cY. (2015).
[46] U.S. Department of Homeland Security (DHS). 2016. Geo-Targeting Performance of Wireless Emergency Alerts in Imminent Threat Scenarios. https://goo.gl/41s3CE. (2016).
[47] U.S. Federal Emergency Management Agency (FEMA). 2016. IPAWS Architecture. https://www.fema.gov/media-library/assets/documents/113642. (2016).
[48] Fabian van den Broek, Roel Verdult, and Joeri de Ruiter. 2015. Defeating IMSI Catchers. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS '15)*.
[49] Wikipedia. 2018. Hawaii false missile alert. https://goo.gl/oD9ofx. (2018).
[50] Xiufeng Xie, Xinyu Zhang, and Shilin Zhu. 2017. Accelerating Mobile Web Loading Using Cellular Link Information. In *Proceedings of ACM MobiSys*.
[51] Hemin Yang, Anpeng Huang, Ruipeng Gao, Tammy Chang, and Linzhen Xie. 2014. Interference Self-Coordination: A Proposal to Enhance Reliability of System-Level Information in OFDM-Based Mobile Networks via PCI Planning. *IEEE Transactions on Wireless Communications* 13, 4 (April 2014), 1874–1887.