White-Stingray: Evaluating IMSI Catchers Detection Applications

Ravishankar Borgaonkar, Andrew Martin
Department of Computer Science
University of Oxford

Shinjo Park, Altaf Shaik, Jean-Pierre Seifert
TU Berlin &
Telekom Innovation Laboratories

Abstract

IMSI catchers, also known as fake base station threats, have recently become a real concern. There are currently a few freely available tools to detect such threats, most of which are Android apps that warn users when they are connected to the fake cellular base station.

In this paper, we evaluate these Android apps and test how resistant they are against various attacking techniques. Such an evaluation is important for not only measuring the available defense against IMSI catchers attacks but also identifying gaps to build effective solutions. We developed White-Stingray, a systematic framework with various attacking capabilities in 2G and 3G networks, and used it for our study. Our results of five popular Android apps are worrisome: none of these apps are resistant to basic privacy identifier catching techniques. Based on our results, we identify limitation of these apps and propose remedies for improving the current state of IMSI catchers detection on mobile devices.

1 Introduction

In the cellular network architecture, the mobile devices are permanently identified by two identities: *international mobile equipment identity* (IMEI) and *international mobile subscriber identity* (IMSI). These identities are exchanged over-the-air with the cellular base stations during a network attach procedure. In order to prevent the linking of a specified user with a particular IMSI, *temporary mobile subscriber identities* (TMSI) were proposed in the cellular security architecture. However, during the two cases – 1) when a device is turned on 2) when the serving network fails to recognize TMSI, IMSI is exchanged unencrypted with the network before a successful authentication procedure.

Besides, by exploiting the above two cases, an adversary can set up a fake network to steal IMSI, IMEI, and TMSI. As a result, this adversary can locate a certain set of mobile devices by co-relating permanent and temporary identities by operating a fake base station – such attacking techniques are termed as *IMSI catchers* (ICs). These ICs are also used for intercepting user traffic such as calls, SMS, etc. Recently there is a significant growth in the illegal use of ICs [47, 32, 19, 8, 36]. In addition, these devices can be easily purchased online at a low cost [11, 10].

As a response to privacy threats arising from ICs, research communities and companies released a variety of *IMSI Catchers Detector* (ICD) apps and techniques. While, there exist earlier studies on the detection of ICs using information from the cellular operator and customized devices [20, 34, 30], our domain of study is different in that we solely focus on the ICD apps available for Android devices. Note that we do not consider expensive and commercial products in this research paper. For example, as of May 2017, Google Play store offers several free ICD apps named as SnoopSnitch, Darshak, AIMSICD, Cell Spy Finder, GSM Spy Finder [37, 48, 9, 46, 25]. With popularity of more than 1500K installations combined, they are aimed at identifying fake base stations. Thus while many such ICD apps exist, it is unclear how effective they actually work.

There is little empirical data available on the accuracy of these apps or on the effectiveness of various approaches to detect ICs [45]. Towards that end, this paper focuses on evaluating popular Android-based ICD apps. We develop a systematic framework called White-Stingray with several common ICs attacking techniques supporting 2G and 3G network. Based on the framework, we generate different cellular radio protocol messages with malicious functionalities to steal IMSI, IMEI, and TMSI. We use these messages to evaluate the effectiveness and robustness of popular apps. While ICs attack is also possible in 4G [43], we exclude 4G in this paper as only one tested app [46] explicitly claims working with 4G ICs.

Our results show that all tested five ICD apps have little protection against common attacking techniques.

In addition, results give insights about detection models used in these tested apps and their capabilities, thus shedding light on possible ways for their improvements. We hope that our findings work as a stepping stone and motivation for the community to improve the current state of ICs detection.

To summarize, this paper makes three research contributions.

- We systematically evaluate ICD Android apps regarding their resistance against various malicious protocol messages revealing IMSI, IMEI, and TMSI. In order to achieve this, we developed White-Stingray to facilitate app evaluation.
- We have implemented a prototype of White-Stingray following ethical concerns and used to evaluate five popular apps. Our findings show that all of them are vulnerable to common circumventing techniques.
- Based on our results, we also explore possible ways to improve tested apps. In particular, we outline new requirements needed in the existing Android system APIs to improve detection capabilities.

Organization. We start from background knowledge to understand the operations of ICs and ICD apps in section 2. Next, we introduce our White-Stingray framework in section 3 followed by the evaluation of ICD apps with the framework in section 4. We discuss results and analysis of ICD apps in section 5, countermeasures for ICD app and mobile OS, baseband developers are followed in section 6. Finally, we conclude the paper in section 7.

2 Background

We present an overview of the cellular network architecture. For simplicity reasons and understandings of ICs attacks, we only describe the radio architecture and protocols of 2G and 3G networks. Then, we iterate how ICs operates and what kind of privacy sensitive information it gathers from the mobile devices. We go through previously proposed ICs detection methods and how they are implemented as mobile apps. Finally, we describe related works on ICs detection.

2.1 Cellular Network Architecture

The cellular radio network is geographically divided into *Location Areas* (LAs) in 2G and *Routing Areas* in 3G by cellular operators. Each LA or RA consists of several cells and each cell or a set of them are under the control of a *Base Station* (BS). A BS can serve up to several users depending on its power capacity and radio range. Each

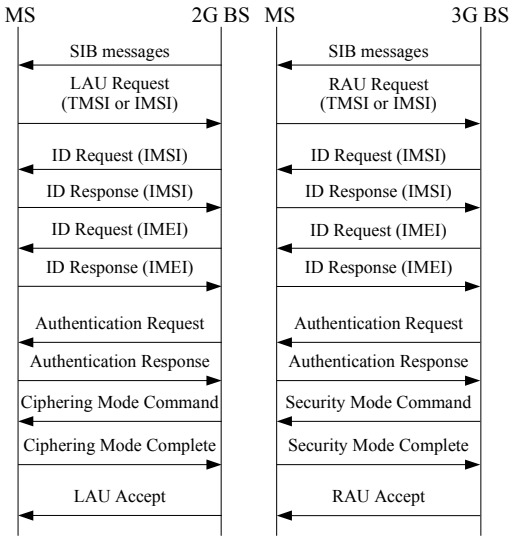

Figure 1: Basic network register procedure between MS and BS.

user carries a *Mobile Station* (MS) which refers to a mobile phone generally. A mobile phone is uniquely identified with IMEI and IMSI which is associated with a *Subscriber Identity Module* or *Universal Subscriber Identity Module* card.

A BS controls and manages over-the-air communication with all the MS's inside a cell. The BS periodically broadcasts network information as *System Information Block* (SIB) messages for MS to connect to the network. For e.g., SIB messages contain *Location Area Code* (LAC) for 2G, *Routing Area Code* (RAC) for 3G, *cell ID* (Cell-ID) that uniquely identify a LA, RA and a cell respectively. They also contain *Mobile Country Code* (MCC) and *Mobile Network Code* (MNC) that uniquely identify a cellular operator network, and neighboring cell lists.

Figure 1 illustrates the procedure followed by a MS to register to a network. MS identifies its own network via SIB messages based on MCC and MNC. When a MS is turned on or moved to a new LA, it initiates a *LA Update (LAU) Request* to the network. If the MS was previously connected to the network, it sends TMSI as an identifier or else it uses IMSI. In case the network failed to recognize TMSI, it calls for a *Identification (ID) Request* procedure. The MS sends an *ID Response* message which contains IMSI in plaintext. Also, certain networks request for IMEI from the MS before authentication procedure, however, this is an optional feature. Upon receiving a valid IMSI, the network sends *Authentication Request* [6] to the MS which includes a random number. MS calculates a response using the random number and sends it back to the network in a *Authentication Response* message. After successful authentication, network initiates a *Ciphering Mode Command* and also includes the

cryptographic algorithm to be used (e.g. A5/1, A5/3) to encrypt the user data. MS generates ciphering keys and responds with an encrypted *Ciphering Mode Complete* message. From this step, the subsequent communications between MS and BS will be encrypted. However, both authentication and ciphering are optional in 2G [6]. The network sends a *LAU Accept* message to the MS with a new TMSI to be used by the MS.

In 3G, BS is referred as *Node B*. But, for simplicity reasons, we refer both 2G BS and 3G *Node B* as BS throughout this paper. Similar to 2G, the MS performs *RA Update* procedure to register to a network. The 3G registration procedure is similar to 2G but differs in the authentication procedure. MS and network perform mandatory mutual authentication, where network authenticates the MS and vice-versa. MS receives an *Authentication Request* with a random number and another parameter called *Authentication Token* (AUTN). MS authenticates the network using AUTN. Upon successfully validating the AUTN, MS and the network establish security algorithms and complete the RAU procedure. In case of an invalid AUTN, the MS responds with an *Authentication Reject* message along with a cause and the RAU procedure is terminated.

2.2 IMSI Catchers (ICs)

An adversary uses ICs for two main purposes: 1) to locate and track certain set or a particular MS, 2) to perform Man-in-the-Middle (MITM) attacks [17] in order to listen to the communication of a targeted MS. Further, an adversary can operate ICs in two modes: passive and active.

In active mode, ICs exploits weaknesses of 2G and 3G. These weaknesses are the combination of architectural design issues, and lack of mutual authentication between MS and BS in 2G [39, 48]. To locate and track MS, an active mode ICs can be operated in 2G and 3G by impersonating a real network operator. Once a targeted MS is attached to ICs, IMSI and IMEI can be obtained by an adversary. Then subsequent communication including calls or SMS of this MS can be intercepted by ICs with 2G support. In order to intercept, ICs can force MS to use no or weaker encryption algorithm (A5/1 [15]) over the radio link. As in 2G security architecture, radio encryption ends at BS, thus adversary can easily intercept encrypted communication. In case the MS is operating in 3G, ICs typically downgrade them to 2G and perform interception.

Unlike active mode ICs, passive mode ICs does not interact with a MS directly, and in addition to IMSI, TMSI can also be used to track people by co-relating known mobile phone number of a particular target. It is further divided into two subtypes: semi-passive ICs triggers silent call or SMS externally [43, 48, 35] to co-relate IMSI and TMSI with target's mobile phone number, while full-passive ICs has the capability of decrypting A5/1 or A5/2 security standard to intercept ciphered calls without interacting with the MS. There are few commercial products in the market for decrypting A5/1 [44].

Note that in this paper, we focus only on evaluating techniques used to detect active and semi-passive mode type of ICs, since the full-passive type of ICs are difficult to detect on mobile devices [40].

2.3 IMSI Catchers Detectors

The ICD tools or apps operate by monitoring and tracking the behavior of the BS for a period of time and/or based on certain known ICs attacking patterns. Currently, there are several types of ICD apps available based on mobile app-based methods [37, 48, 9, 46, 25], and network-based methods [20, 34]. Additionally, there are ICD devices like GSMK CryptoPhone [21, 30]. However, we are excluding the network-based and device-based method from our research since the former requires support from the operator, and the latter is expensive and not openly available for research. Therefore, we focus on app-based ICD in this paper.

For our analysis, we selected the popular free Android apps from the Google Play Store as listed in Table 1. The reason for selecting these apps is due to their number of installations and claims in detecting ICs. Further, we also surveyed AIMSICD [9], whose source code is available but not listed in the Google Play Store. In this paper, we will refer these apps as App1 to App5 according to Table 1.

ID	App Name	Downloads[1]	OSS?
App1	SnoopSnitch [37]	100k-500k	✓
App2	Cell Spy Catcher [46]	100k-500k	✗
App3	GSM Spy Finder [25]	100k-500k	✗
App4	Darshak [48]	10k-50k	✓
App5	AIMSICD [9]	[2]	✓

Table 1: Overview of the ICD apps discussed in this paper. Apps with available source code are given a check mark in the OSS (Open Source Software) column.

These ICD apps use the public and private interfaces provided by smartphone OS in order to access cellular network information. For e.g., LAC and Cell-ID [27], signal strength [29] and neighboring cell lists [28] are retrieved via public API whereas SIB messages and other dedicated signaling messages are accessed with a private API (requiring root access). As Apple iOS and Windows mobile OS lacks such type of APIs [13], ICD apps are only available for the Android platform as of now.

[1] Provided by Google Play Store, may not reflect actual app users
[2] Not listed in Google Play Store

Among the apps, App1 and App4 require root permission to operate, whereas App2 and App3 does not require. However, App5 makes a compromise, by enabling optional features when root permission is available. The advantage of App1 and App4 is that they have unrestricted access to the baseband information of Qualcomm and Intel baseband chip logs respectively.

2.4 Related Work

We divide the related work into two categories. The first is on evaluation of ICD apps. Brenninkmeijer [18] presented a user experience of IMSI catchers detection apps by simulating a fake network and comparing how those apps are acting. Although we selected similar sets of apps, we study their detection models and test them against our White-Stingray framework to evaluate the effectiveness and robustness.

The latter work is about the evaluation of network-based and device-based ICD applications. Dabrowski et. al [20] and Li et. al [34] presented ICs detection methods using network-based data. Dabrowski et. al analyzed signaling messages and timing fingerprint of authentication from multiple users to detect and catch the ICs. Li et. al used exact location of a legitimate base station with its ID as one of the factors in their detection method. However, their detection methodology is not evaluated in the literature as of yet. The Norwegian Police Security Services and Simula Research Laboratory [45] evaluated device-based ICD apps such as GSMK's CryptoPhone [30] and Delma's [21] Network Guard to investigate the presence of ICs in Oslo city. Their results concluded that measurements from both device-based ICD apps do not constitute a compelling case that ICs were in use in Oslo city. Similar conclusions are indicated by our study but against Android-based ICD apps in this paper.

3 White-Stingray Framework Design

In this section, we describe our White-Stingray framework design, capabilities, and implementation. First, as a part of the design, we investigate and discuss techniques used by ICD apps to identify the presence of ICs in 2G and 3G. In particular, we present the parameters that are detected by ICD apps based on the available source code and documentation. Additionally, we introduce new undetectable parameters that can be used by ICs to covertly collect IMSI and IMEI. Lastly, we discuss the implementation of our framework including hardware and software modifications.

3.1 Detectable Parameters by ICD apps

We describe 2G and 3G radio protocol information used by ICD apps for detection purpose. In this paper, we refer them as detectable parameters and divide into following three types – layer 1, broadcast and dedicated signaling. We list a superset of detectable parameters in Table 2. The signaling messages are defined by 3GPP standards, covering both 2G and 3G.

Layer 1

Generally, layer 1 radio protocol messages are decoded by baseband chips and a subset of parameters are relayed to mobile OS. Important layer 1 information is as follows:

Rx power. The Rx in MS is the value of power in decibels per milliwatt of the received signal from the BS. According to standards [2], MS always prefers to attach to a BS with the strongest Rx value. Thus, ICs transmits higher signal value to attract the MS to be attached. App3 and App5 monitor such type of Rx signal strength.

Broadcast Signaling

In 2G and 3G, the BS broadcast different types of signaling messages within a particular range. These type of configuration messages are necessary for MS to select a particular BS. The following broadcast signaling parameters are used by ICD apps:

LAC. As soon as a MS moves to a different LA, it must perform LA update procedure with the network to inform its current location. Thus adversaries generally exploit this LA update mechanism to force MS to attach to their ICs. Accordingly, they configure their ICs with a LAC different from the location where the targeted MS is physically present. All the apps monitor this LAC change behavior and alert the user about the authenticity of the LAC. For example, App1 checks the validity of current LAC with its neighbor cells, while App2 warns for every LAC change. As a result, App1 alarms when the MS is close to the border of the LA, whereas App2 generates an alarm whenever user travels to a new LA, whose LAC is not stored in the app history. However, LAC parameters may not be a sign of ICs presence due to the fact that change in LA could be a legitimate network operation (handover) in which MS is traveling in a car or train.

Neighbor cell list. MS constantly measures the signal strengths of BSs received in the neighbor cell list, and inform the network to facilitate handovers between cells. The neighbor cell list is available to the MS from the SIB messages. If there are no neighbor cells, then MS does not attempt to scan more cells and lock itself to the serving cell. To exploit this fact, ICs broadcasts empty neighbor cell list to MS to prevent attaching to the real network. App1, App2, and App5 monitor the neighbor cell list and raise an alert when it is empty. However, in rural areas, there may be a case that there are no neighbor cells, thus generating an alert based on the empty list alone may not be a concrete evidence for ICs' presence.

Parameters	App1	App2	App3	App4	App5
Layer 1					
Rx power (P1)	✗	✗	✓	✗	✓
Broadcasted signaling					
SIB messages (P2)	✓	✗	✗	✓	✗
LAC and Cell-ID (P3)	✓	✓	✓	✓	✓
Neighboring cell lists (P4)	✓	✓	✓	✓	✓
Paging (P5)	✓	✗	✗	✗	✗
Dedicated signaling					
Identity requests (P6)	✓	✗	✗	✓	✗
Authentication procedure (P7)	✓	✗	✗	✓	✗
Ciphering and integrity protection (P8)	✓	✗	✗	✓	✓
Silent SMS (P9)	✓	✗	✗	✓	✓
Reject messages (P10)	✓	✗	✗	✗	✗

Table 2: Detectable Parameters used by ICD apps. Check mark means the parameter is implemented, cross mark means the parameter is not present.

Cell-ID. The Cell-ID parameter uniquely identifies each BS or a particular sector or coverage region within a LAC. App1 compares whether the current Cell-ID is not present in the adjacent cell's neighbor cell list. App1 also records the Cell-ID with its GPS coordinates and uses it for the further analysis. App2 treats Cell-ID similar to the LAC, where it generates an alarm whenever the user enters a new cell. App3 generates an alarm when current Cell-ID is in between 0 and 9. App5 can check the current Cell-ID with the external database like [49].

However, Cell-ID can be changed by the network operator due to operational change in their configuration policy, and hence they are not permanent identifiers. In addition, during a concert or football match cellular network operators deploy temporary BSs having a new Cell-ID (which may not be in [49]). In this case, alerts based on Cell-ID only may not conclude the presence of ICs.

Paging. To locate a MS (for delivering an incoming call, SMS or data), the network uses paging procedure. It involves broadcasting paging messages by the BS in a cell or a group of cells. Paging message contains TMSI(s), IMSI(s), or both. In a normal condition, the network uses a TMSI to page the MS. In case of a network failure, or when a MS does not respond for paging with TMSI, the network uses IMSI. When a MS receives the paging with its TMSI or IMSI, it connects to the network to receive the call or data setup. However, when paging is received with IMSI, it reconnects to the network, performs a LA update procedure to acquire a new TMSI and then proceeds to receive call or data.

Except for App1, no other apps monitor paging messages. When App1 observes paging with IMSI, it considers this as an ICs event. However, paging request with IMSI could be due to a case of network failure. For example, BS rebooted and lost existing database of a relation of IMSI and TMSI.

Dedicated Signaling

After a particular BS is selected by the MS for the network registration procedure, dedicated signaling messages are used for authentication and further network services. Following are the dedicated signaling parameters used by various apps:

Identity requests. According to [6] both IMSI and IMEI can be requested any time in plaintext in 2G and 3G. In certain cases, networks request IMEI only after the security session is established. In general, ICs lacks parameters required for a successful 2G and 3G authentication procedure, they request both IMSI and IMEI before initiating authentication. This behavior may indicate the presence of an ICs, however requesting IMEI before authentication is not a mandatory procedure [6] and depends on the network configuration. Such a type of behavior is only monitored by App1 and App4.

Authentication Procedure. In 2G, ICs can bypass the authentication procedure with MS due to lack of network authentication. However, MS authenticates 3G networks and vice-versa. Apps A1 and A4 monitor only if the authentication has been performed or not and generates a notification but not any alarm to the user.

Ciphering and integrity protection. In 2G, ciphering algorithms are A5/1, A5/3, and in 3G, ciphering algorithms are UEA1, UEA2 and integrity protection algorithms are UIA1, UIA2 [24]. Integrity protection is available only in 3G [4]. In case of null encryption, A5/0 is used in 2G, and UEA0/UIA0 is used in 3G. Thus ICs tends to force MS to use A5/0 in 2G and UEA0/UIA0 in 3G in order to be able to intercept the traffic.

App1, App4, and App5 track currently used ciphering

and integrity protection algorithms, and warn if a null security (all the keys are set to zero) algorithm is used. Note that, 3GPP TS 22.101 [7] mandates the use of ciphering indicator in MS, however, network operators can disable it, and moreover, the majority of the phone manufacturers do not implement it.

Silent SMS. The 'Short Message Type 0' as defined in [3] is known as a silent SMS. This type of SMS allows the user to send a message to another MS without the knowledge of recipient [23]. After receiving, the MS discards the message and does not produce any notification to the user. The police in various European countries use these type of SMS to locate the targeted MS. However, ICs are not required to send silent SMS. In few cases, adversary operating ICs do not know co-relation between IMSI and target's identity. Then if target's mobile phone number is known, silent SMS can be used to co-relate IMSI and mobile phone number using active or passive ICs. App1 and App4 both monitor silent SMS and alert the users.

Reject messages. In certain cases, the serving network can deny telephony services to any MS during the LAU procedure by sending LAU reject messages. For example, denying cause could be subscription issues (roaming not allowed or insufficient credits), network failure, etc. Upon receiving a reject message, the MS may or may not connect to a BS depending on the specified cause. ICs exploit these messages with various causes to deny service to the MS. Upon receiving this reject message, the MS may or may not attempt to attach to a new BS based on reject cause. ICs exploit these LAU reject messages to initiate *Identity Request* procedure for collecting IMSI or IMEI. Only App1 track such types of reject messages.

3.2 Undetectable Parameters

In this section, we describe 2G and 3G radio protocol information which is not used by tested ICD apps for detection purpose, but important in building effective heuristics to detect ICs on mobile devices. During evaluation of ICD apps using our framework, we do not expect surprising findings. However these undetectable features need to considered for the detection purpose, hence we implement in our framework. We refer such type of protocol information in rest of the paper as undetectable parameters.

RAND and AUTN. The 2G network send a random number called RAND to the MS to perform authentication and receive an authentication response called RES. This RES is unique for a particular RAND and can be generated by the same MS only. Hence, ICs can replay this RAND in a certain area to verify the presence of the MS in that area without using IMSI [14].

3G networks generate an AUTN based on a sequence number and send it to the MS. If MS identifies that the received AUTN is out of sequence, it considers this token as used and sends *Authentication Reject* message to the network with reject cause either *MAC Failure* or *Synch Failure* [4]. The former indicates that this AUTN does not belong for this IMSI, while the latter indicates that the token is previously used and cannot be reused. ICs can replay used token to the MS to track its presence in a certain area. However, if the token is valid and unused, the authentication process will be successful.

Figure 2: Our experimental setup with host and USRP. Phones are running the ICD apps.

Location. Radio Resource Location services Protocol (RRLP) [5] is used by the network to locate the MS during an emergency situation, for e.g. 911. RRLP does not mandate authentication and can be used during a voice call, SMS or LAU procedure. Upon RRLP request, the MS provides the GPS location information to the network, provided the location settings are enabled from the OS.

NITZ. Network can optionally announce its name, and current time and date in the form of *Network Information and Time Zone* (NITZ) message [1]. This is usually sent after successful security setup. Depending on the operator policy, NITZ could be sent during every LAU, selectively, or not sent at all [38].

Note that, NITZ messages do not facilitate ICs in stealing IMSI or IMEI. However, NITZ messages may be a parameter to detect anomalies in the network information. For example, an operator configures the network to send NITZ before starting authentication procedure. If ICs does not use such type of configuration, then alert can be raised.

3.3 Hardware and Software Implementation

We systematically analyze detectable and undetectable parameters of ICD apps, and implement them into our own ICs for the evaluation. Accordingly, we describe hardware, software, adversary model, and capability of our ICs framework White-Stingray in this section.

Hardware. Our hardware platform consists of a host embedded PC and a USRP [22] as shown in Figure 2. USRP acts as a radio frontend, and is connected to he PC via USB 3. The PC runs operational 2G and 3G network software components required for the White-Stingray framework. We use our own smartphones and commercial SIM cards to evaluate White-Stingray framework.

Software. We build an experimental 2G and 3G network setup using the open source software OpenBTS [41] and OpenBTS-UMTS [42] respectively.

Ethical concerns. We evaluated our White-Stingray framework inside the Faraday Cage [26].

Adversary Model. For performing ICs attack against ICD apps, we consider following adversary model. The adversary can operate in two modes: passive and active. In passive mode, the adversary can listen to 2G or 3G broadcast channels and decode information such as IMSI, BS configuration details etc. In active mode, the adversary can operate both 2G and 3G rogue BSs that can impersonate any operator network. Further, the adversary can inject modified radio protocol messages into the MS. In this mode, adversary can also behave as a MITM between the MS and network. Specifically, the attacker's ICs is capable of decoding the messages exchanged between the MS and the network. We only cover active mode in this paper.

White-Stingray Capabilities. Our White-Stingray framework generates several 2G and 3G ICs patterns to study the detection behavior of ICD apps. In particular, it generates detectable and undetectable parameters as discussed earlier Section 3.1 and Section 3.2.

We modify OpenBTS and OpenBTS-UMTS software to incorporate test patterns. To generate test ICs patterns, we operate the framework with both normal and abnormal network configurations and unexpected protocol sequences in signaling messages. Specifically, the framework allows us to operate the network with odd LAC and Cell-ID values. Further, it assists in bypassing authentication and ciphering related signaling messages, requests for IMEI and IMSI with sudden connection releases, initiate paging with IMSI and TMSI, send null *Ciphering Mode Command* and downgrade messages, send silent SMS etc.

Additionally, we implement a silent calling mechanism into our framework. A silent call is a procedure to dial a certain number and hang up the call before the target's MS starts to ring a notification of the incoming call. The purpose of a silent call is to trigger paging messages to a certain MS. According to [35], there is a mean delay of 8 seconds between the user dialling a number and the phone ringing. Accordingly, calls aborted before 5 seconds following the call initiation would result in no rings, but by that time, a paging request would already broadcasted by the BS. A passive adversary will be able to capture these paging message and by repeating this procedure the adversary can reveal the presence of the MS in a certain location.

Further, the framework can generate undetectable patterns such as normal network configurations, unexpected protocol sequences, and unauthenticated messages in order to escape the detection by the ICDs. Specifically, the framework generates paging messages with TMSI, request IMSI and IMEI with sudden connection releases, send ineffective *Ciphering Mode Command* with null ciphering and downgrade messages, etc.

4 ICD Apps Evaluation and Results

Our evaluation phase involves the operation of White-Stingray framework where various network parameters are attributed and altered continuously with several types of configurations and tested against ICD apps. This allows us to generate 2G and 3G ICs patterns listed in Table 3 to probe both detectable and undetectable parameters. During this, the parameters encounter several normal and abnormal values including non-standard protocol sequences. We analyze the detection behavior of all the apps.

During the evaluation, we assume that our test MS is initially registered to a real 2G or 3G network. We configure our White-Stingray framework to impersonate as 2G and 3G ICs. Then we perform following pattern variations and non-standard protocol sequence changes against ICD apps:

Fluctuating Rx power and duration. We operate our framework in 2G and 3G networks with varying power levels and duration. The goal is to verify if ICD apps detect short and long lived high power fake base stations.

App2 only stores the duration of the MS residing on the particular base station or ICs. App3 presents and stores the Rx power of the ICs along with its LAC and Cell-ID. App5 stores both Rx power and duration of ICs. But none of the apps perform ICs detection by comparing stored data with new measurements.

Unusual network configurations. We set up our framework to operate with following three different cases of network configurations: 1) LAC and Cell-ID not observed in the surroundings, 2) neighboring LAC and non-neighboring Cell-ID and 3) neighboring LAC and Cell-ID. In all the cases, we operate with and without a neighbor cell list.

We observe that in the first case, App1 identifies the framework as a lonesome LA (LA with the single cell), whereas App2 warned about the change in LAC, hence both considered White-Stingray framework to be an ICs. In the second case, App1 considers the framework as normal and App2 alarmed about the new Cell-ID. In the third case, App1 identifies our framework as a ICs pattern when the frequency is different from the real BS. For

ICs Patterns	App1		App2		App3		App4		App5	
	2G	3G	2G	3G	2G	3G	2G	3G	2G	3G
Fluctuating Rx power and duration (C1)	✗	✗	✓	✓	✓	✓	✗	✗	✓	✓
Unusual network configurations (C2)	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
Unusual identity requests (C3)	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗
Unusual paging messages (C4)	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗
Authentication token replay (C5)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Zero or weak security (C6)	✓	✓	✗	✗	✗	✗	✓	✓	✓	✓
Silent SMS (C7)	✓	✓	✗	✗	✗	✗	✓	✓	✓	✓
Silent calls (C8)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Fake NITZ (C9)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Location leaks (C10)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Unusual downgrade from 3G to 2G (C11)	–	✗	–	✗	–	✗	–	✗	–	✗

Table 3: Results of White-Stingray pattern evaluation against ICD apps. Check mark means the pattern is detected, cross mark means the pattern is not detected. Dash means the pattern is technically not possible.

all above cases, when the neighboring cell list is empty, only App1 considers this as ICs pattern.

However, App3 exhibited a strange behavior, where it detects LAC or Cell-ID between 0 and 9 as an ICs. Any other values are considered as normal. App5 stores every visited Cell-ID for corresponding LAC, and it warns if the stored Cell-ID is found with a LAC different from the stored one.

Unusual identity requests. The White-Stingray framework sends identity requests by not only following standard 2G and 3G radio protocol sequence but also modifying these radio messages. However, note that our modification to the standard radio protocol sequence still allows the framework to communicate with MS effectively.

We configure the LAC and RAC of White-Stingray in 2G and 3G network mode respectively to be different from the neighboring BS. This step triggers a LAU procedure during which our framework obtains (before authentication) both IMSI and IMEI from the MS. Further, MS can either accept or reject the LAU procedure. Similarly, in the 3G RAU procedure, framework receives both IMSI and IMEI and can only reject the connection since authentication cannot be bypassed.

Subsequently, we repeated the above procedures and introduce protocol sequence modifications. Upon collecting the IMSI and IMEI, we abruptly release the radio connection without issuing any kind of accept or reject messages thereby breaking the protocol sequence. None of the apps detected modified protocol sequence and leakage of IMSI and IMEI. Similarly, all apps fail to recognize standard protocol sequence except App1.

Unusual paging messages. We configure White-Stingray to use the same LAC and RAC of the nearby BS respectively in 2G and 3G network mode. When MS detects the high Rx power, it does not trigger any LAU or RAU procedure, but rather listens to White-Stingray's broadcast channels. Further, our framework broadcast three types of paging messages that contain: 1) only TMSI, 2) only IMSI, 3) both TMSI and IMSI of the targeted MS. In all the cases, MS initiates a connection to our framework to receive call/SMS, but however, none of them are delivered to the MS. Consequently, the connection is released.

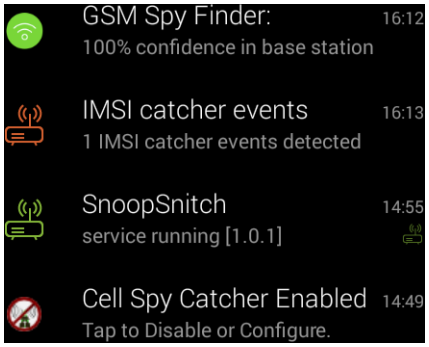

Figure 3: Paging by IMSI – Only App1 shows ICs event.

Figure 3 displays the behavior of the apps for this pattern. App1 detects the first type as a normal activity, whereas the second and third as ICs patterns. However, none of other apps are able to detect these unusual paging messages from our White-Stingray framework.

Authentication token replay. Assuming the target MS's IMSI or TMSI is known, we collect AUTNs with our White-stingray [14]. We then operate our framework in 2G and 3G network mode and allow the MS to initiate a LAU and RAU procedure respectively.

In the 2G mode, we replay the same random number for multiple times to the MS. Based on the authentication response, the framework decides the target's presence in the LA. However, the authentication itself will be always successful, ICD app needs to keep track of used random number by itself to check whether the number is replayed

or not. None of the apps are capable of this.

In the 3G mode, we replay the collected AUTN to the MS. MS could either reject or accept the authentication procedure, based on the validity of the AUTN thereby revealing targeted MS's presence in the corresponding RA. None of the apps detect this authentication token replay pattern based attack.

Null security. During network attachment procedure, we force test MS to select A5/0 security algorithm using our framework in a 2G mode. This can be achieved in following two ways:

1) We setup the framework such that it can trigger a LAU procedure from the MS. During this procedure, our framework bypasses both authentication and ciphering procedures and successfully finishes the LAU procedure. As a result, since no algorithm is specified the MS uses an A5/0 ciphering algorithm.

2) We configure the framework similar to C4 and initiate a call or SMS to the MS. This MS receives a paging message from the framework and responds to receive the call/SMS. Before actually delivering the call, we send a *Ciphering Mode Command* to the MS and indicate the use an A5/0 algorithm. Accordingly the MS selects A5/0 algorithm and continues to receive the call or SMS.

We observed that App1, App4, and App5 detect both the cases as suspicious ICs patterns and is only displayed within the app user interface. However, none of them alert the user with a warning.

Fake NITZ. We send a NITZ message to the MS similar to [38] during 2G and 3G registration procedure using our framework. Upon receiving, the MS displays a wrong time and date, and in certain phones, the Android OS crashes [38]. None of the apps monitor this pattern.

Silent calls and SMS. We initiate silent calls and SMS to the target MS. To deliver calls and SMS, paging messages are broadcasted to the MS. In case of a silent call, since the MS is paged by its TMSI, this is considered to be a normal activity. Surprisingly, none of the Apps consider whether the MS has received a special type of silent call or not. However in case of silent SMS, App1, App4, and App5 analyze the type of SMS and detect this as an ICs attacking pattern.

Location leaks. The framework allows us to send RRLP positioning request message to the test MS. After receiving such messages, the MS replies with the current GPS coordinate. According to 3GPP specifications, when an ICs sends RRLP positioning request message, RRLP capable MS replies with the current GPS coordinate. None of the apps monitor such type of RRLP messages. Note that, if MS does not contain GPS chip inside then it sends error message instead.

Unusual downgrade from 3G to 2G. We configure the RAC of the framework in the 3G mode to be different from the neighboring BS. This triggers a RAU request message from the MS. In response, our framework sends RAU reject message with the cause *GPRS services not allowed*. According to the specification [6], the MS switches to 2G network upon reception of this message. However, we could not find any ICD apps detecting this type of ICs attacking pattern. Note that, in order to regain 3G services, the MS has to be rebooted.

5 ICD Apps Strengths and Weaknesses

We begin by describing our findings after applying White-Stingray framework against all tested ICD apps. We divide our findings into two types: insufficient detection strategies and app's capability constraints. Based on our ICD app evaluation, we derive that none of the apps are able to detect our ICs circumventing techniques. For describing results, we refer false positive as ICD detecting a normal BS as an ICs and false negative as vice-versa. We refer detectable parameter as P1 to P10 (Table 2) and ICs patterns as C1 to C11 (Table 3) respectively for the following discussions.

Insufficient Detection Strategies

We found that strategies like improper parameter selection and unintended protocol messages cause false positives and false negatives.

Improper parameter selection. App2 and App3 use broadcast signaling information as their main detection parameter. Simple modifications on the messages as discussed in previous section 3 can cause false negatives.

Unintended protocol messages. By sending radio protocol messages with the unintended flow, known 2G and 3G flaws are exploited and ICD apps fail to detect them.

We introduced obscure variations in the attacking patterns used by ICs. For example, while evaluating C3, we terminate the execution of a radio protocol sequence by denying to send the accept or reject message. As a result, App1 was not able to detect this variation despite leaking IMSI and IMEI to our White-Stingray. While App1 monitors P10, it fails to interpret the all types of causes that can be sent by the network. As a result, in C11 the ICs was able to downgrade the MS from 3G to 2G.

Although paging with IMSI can be circumvented by paging with TMSI, the adversary should know the TMSI of the target MS beforehand. The link between target MS's identity and TMSI can be made by using silent calls [43] (assuming target's mobile phone number is known). But silent call patterns are not detected by ICD apps, thereby allowing a passive adversary to track the location of the MS. App1 treats all types of non-plaintext SMS (such as binary, silent) as malicious. However, this could trigger false positives in cases where the network sends binary SMS to the MS during mobile payment services like m-pesa [50].

While the absence of ciphering and integrity protection (P8, C6) usually indicates the presence of ICs, it is also possible that some base stations are pre-configured for no ciphering due to various reasons [31]. This could trigger a false positive. In case where an adversary operates a compromised femtocell [16], detecting by the ICD apps is not possible and trigger false negative.

Capability Constraints

We identified two types of capability constraints that are responsible for the non-effectiveness of tested ICD apps.

Design constraints. The ICD apps operating without root permission are limited to the parameters provided by Android API. Both App2 and App3 are not using root access, therefore they lack access to lower level cellular radio protocol information parameters processed by the baseband chip. Insufficient support from Android OS further limits the capability of apps. For example, the support of ciphering indicator on Android was first discussed in 2009, but this is still not implemented [12]. Lack of standardized API to access the baseband information is another limiting factor.

Further, from the usability perspective, only App2 and App3 generate an alarm (play a sound and vibrate) to the user for certain network configurations in C2. However, they are false positives. Besides, no other app raises an alarm when the ICs operate the pattern C4, C5, and C6 but rather only display the ciphering and authentication information inside the app.

Missing parameters. None of the ICD apps are implementing the parameters listed in Section 3.2. Hence, they fail to detect them, for example when ICs injects incorrect time information and obtain the MS's GPS coordinates. Further, location tracking attacks based on AUTN and RAND can be performed by our ICs and still remain undetected.

6 Countermeasures

We propose countermeasures to improvise ICs detection methodologies for ICD apps and discussion on support needed from mobile phone vendors.

ICD app developers. First of all, the public APIs provided by the Android OS are not functionally enough to design an effective ICD app. They provide only broadcast parameters, which ICD apps must validate with other external and non-reliable public sources (for example, OpenCellID project). Relying merely on the broadcast parameters can result in false positives and negatives.

Due to different types of network configurations by operators and complexity of the cellular protocol, radio protocol messages could occur in any unexpected order and protocol parameters may contain uncommon values. These uncommon values are not interpreted by ICD apps, thus resulting failure in alerting suspicious activities. In addition to standard protocol procedures, ICD apps need to consider non-standard protocol behaviors. All radio protocol messages in 2G and 3G carrying privacy sensitive parameters should be considered and implemented into their detection engines.

On usability aspects, once an ICs is detected, ICD apps must provide clearly understandable notifications to the user in a reasonable time. Apps generating alarm too frequently by monitoring broadcasted parameters alone will annoy the user, making them loose trust on the app and ignore real threat alarms. On the other hand, some apps do not trigger notification alarms on possibly the most noteworthy parameter like null ciphering. However due to limitation of Android APIs, developers can not warn users in real time or during the call establishment about non-encrypted 2G or 3G connectivity.

Mobile OS and baseband developers. Because all the cellular communication data including cryptographic session keys are processed within the baseband processor, it is acceptable that opening up the access to the baseband can potentially be risky. However, ICD apps must rely on the radio protocol information received from baseband to detect suspicious ICs patterns. Thus having access to a secure API between smartphone OS and baseband OS would assist ICD developers. However what kind of information such type of secure API contains is a debatable issue since cellular network providers would not like to disclose weaknesses of serving base stations.

As an example towards this direction, recently Xiaomi announced its own smartphone processor with integrated baseband chip, which claims that it has ICD capability inside the baseband [33]. Even though 5G networks may fix IMSI catchers issues by improving security by introducing new architecture, the ICs for 2G to 4G would still exist. Thus baseband or device vendors need to make extra efforts to solve existing ICs problem - either by opening up new mechanisms within baseband or allowing more access to app developers without root access.

7 Conclusion

We have studied and analyzed the detection capabilities of 5 popular ICD apps. Further, we developed a White-Stingray framework to perform 2G and 3G ICs attacks against the apps. We learned that none of the tested apps are able to detect the ICs effectively and many of their techniques can be circumvented by our framework. We discussed the strengths and weaknesses of selected ICD apps. Finally, we propose countermeasures to app developers and phone manufacturers so as to improve the detection state of the ICD apps.

Acknowledgments

This research was partly performed within the 5G-ENSURE project (www.5GEnsure.eu) the EU Framework Programme for Research and Innovation Horizon 2020 under grant agreement no. 671562.

References

[1] 3GPP. Network Identity and TimeZone (NITZ); Service description; Stage 1. TS 22.042, 3rd Generation Partnership Project (3GPP).

[2] 3GPP. Radio subsystem link control. TS GSM 05.08, 3rd Generation Partnership Project (3GPP).

[3] 3GPP. Technical realization of the Short Message Service (SMS). TS 23.040, 3rd Generation Partnership Project (3GPP).

[4] 3GPP. 3G security; Security architecture. TS 33.102, 3rd Generation Partnership Project (3GPP), Dec. 2010.

[5] 3GPP. Location Services (LCS); Mobile Station (MS) - Serving Mobile Location Centre (SMLC) Radio Resource LCS Protocol (RRLP). TS 44.031, 3rd Generation Partnership Project (3GPP), Mar. 2010.

[6] 3GPP. Mobile radio interface Layer 3 specification; Core network protocols; Stage 3. TS 24.008, 3rd Generation Partnership Project (3GPP), Sept. 2012.

[7] 3GPP. Service aspects; Service principles. TS 22.101, 3rd Generation Partnership Project (3GPP), July 2013.

[8] AFTENPOSTEN NEWS. New report: Clear signs of mobile surveillance in Oslo, despite denial from Police Security Service. http://www.aftenposten.no/norge/New-report-Clear-signs-of-mobile-surveillance-in-Oslo_-despite-denial-from-Police-Security-Service-61149b.html.

[9] AIMSICD: Android-IMSI-Catcher-Detector. https://github.com/CellularPrivacy/Android-IMSI-Catcher-Detector/.

[10] ALIBABA. IMSI Catcher. https://www.alibaba.com/product-detail/IMSI-catcher_135958750.html.

[11] ALIBABA. Tracking suspect police use IMEI and IMSI. https://www.alibaba.com/product-detail/tracking-suspect-police-use-IMEI-IMSI_60257431818.html.

[12] ANDROID PUBLIC TRACKER. Ciphering Indicator. https://issuetracker.google.com/issues/36911336.

[13] APPLE DEVELOPER FORUMS. How can I get CellID, LAC and others device information? https://forums.developer.apple.com/thread/21018.

[14] ARAPINIS, M., MANCINI, L., RITTER, E., RYAN, M., GOLDE, N., AND REDON, K. New Privacy Issues in Mobile Telephony : Fix and Verification. *Proceedings of the 2012 ACM conference on Computer and communications security* (2012), 205–216.

[15] BIRYUKOV, A., SHAMIR, A., AND WAGNER, D. Real time cryptanalysis of a5/1 on a pc. In *International Workshop on Fast Software Encryption* (2000), Springer, pp. 1–18.

[16] BORGAONKAR, R., REDON, K., AND SEIFERT, J.-P. Security analysis of a femtocell device. In *Proceedings of the 4th International Conference on Security of Information and Networks* (New York, NY, USA, 2011), SIN '11, ACM, pp. 95–102.

[17] BOTT, R., AND FRICK, J. Method for identifying a mobile phone user or for eavesdropping on outgoing calls, July 25 2001. EP Patent EP1051053.

[18] BRENNINKMEIJER, B. Catching IMSI-catcher-catchers: An effectiveness review of IMSI-catcher-catcher applications, 2016.

[19] CBC NEWS. Someone is spying on cellphones in the nation's capital. http://www.cbc.ca/news/politics/imsi-cellphones-spying-ottawa-1.4050049.

[20] DABROWSKI, A., PETZL, G., AND WEIPPL, E. R. The Messenger Shoots Back: Network Operator Based IMSI Catcher Detection. *Research in Attacks, Intrusions and Defenses* (2016).

[21] DELMA MSS LIMITED. Network Guard. http://www.delmamss.co.uk/products.

[22] ETTUS RESEARCH. USRP Software Defined Radio (SDR) online catalog. https://www.ettus.com/product.

[23] EUROPEAN DIGITAL RIGHTS (EDRI). Police frequently uses Silent SMS to locate suspects. https://edri.org/edrigramnumber10-2silent-sms-tracking-suspects/.

[24] FOSBERG, D., HORN, G., MOELLER, W.-D., AND NIEMI, V. *LTE Security*, 2nd edition ed. John Wiley and Sons Ltd., 2013.

[25] GALAN. GSM Spy Finder. https://play.google.com/store/apps/details?id=kz.galan.antispy.

[26] GAMRY INSTRUMENTS. The Faraday Cage: What Is a Faraday Cage & How Does It Work? https://www.gamry.com/application-notes/instrumentation/faraday-cage/.

[27] GOOGLE INC. CellIdentityGsm - Android API Documentation.

[28] GOOGLE INC. NeighboringCellInfo - Android API Documentation.

[29] GOOGLE INC. SignalStrength - Android API Documentation.

[30] GSMK. Cryptophone. http://www.cryptophone.de/.

[31] GSMK. Questions about the Interception of GSM Calls. http://www.cryptophone.de/en/support/faq/questions-about-the-interception-of-gsm-calls/.

[32] IBTIMES NEWS. Brexit Britain has turned a blind eye to the death penalty - we're busy trading morals for money. http://www.ibtimes.co.uk/someone-could-be-secretly-spying-mobile-communications-white-house-pentagon-who-1612274.

[33] JOE ROBERTS. Xiaomi Mi 5C: All you need to know about the Surge S1-powered phone. http://www.trustedreviews.com/news/xiaomi-mi5c-surge-s1-news-specs-release-date-price.

[34] LI, Z., WANG, W., WILSON, C., CHEN, J., QIAN, C., JUNG, T., ZHANG, L., LIU, K., LI, X., AND LIU, Y. FBS-Radar: Uncovering Fake Base Stations at Scale in the Wild. *NDSS* (2017).

[35] LIN, Z., KUNE, D. F., AND HOPPER, N. Efficient private proximity testing with GSM location sketches. In *Financial Cryptography and Data Security - 16th International Conference, FC 2012, Kralendijk, Bonaire, Februray 27-March 2, 2012, Revised Selected Papers* (2012), pp. 73–88.

[36] NAKEDSECURITY BLOG. Soldiers sent hate-SMS messages from rogue base stations. https://nakedsecurity.sophos.com/2017/05/12/soldiers-sent-hate-sms-messages-from-rogue-base-stations/.

[37] NOHL, K. Mobile selfdefense. *31st Chaos Communication Congress* (2014).

[38] PARK, S., SHAIK, A., BORGAONKAR, R., AND SEIFERT, J.-P. White rabbit in mobile: Effect of unsecured clock source in smartphones. In *Proceedings of the 6th Workshop on Security and Privacy in Smartphones and Mobile Devices* (2016), ACM, pp. 13–21.

[39] PESONEN, L. Gsm interception. http://www.tml.tkk.fi/Opinnot/Tik-110.501/1999/papers/gsminterception/netsec.html, 1999.

[40] PRIVACY INTERNATIONAL. Phone Monitoring. https://www.privacyinternational.org/node/76.

[41] RANGE NETWORKS. OpenBTS. http://openbts.org/.

[42] RANGE NETWORKS. OpenBTS-UMTS. http://openbts.org/w/index.php?title=OpenBTS-UMTS.

[43] SHAIK, A., BORGAONKAR, R., ASOKAN, N., NIEMI, V., AND SEIFERT, J.-P. Practical attacks against privacy and availability in 4G/LTE mobile communication systems. In *23rd Annual Network and Distributed System Security Symposium, NDSS San Diego, California, USA, February 21-24, 2016* (2016).

[44] SHOGHI COMMUNICATIONS LTD. A5.1 Decryptor. http://www.shoghicom.com/a5-decryptor.php.

[45] SIMULA RESEARCH LABORATORY. An investigation into the claims of IMSI catchers use in Oslo in late 2014. http://www.pst.no/media/76725/IMSI-report-SimulaResearch-Laboratory.pdf.

[46] SKIBAPPS. Cell Spy Catcher (Anti Spy). https://play.google.com/store/apps/details?id=com.skibapps.cellspycatcher.

[47] THE REGISTER NEWS. Fake mobile base stations spreading malware in China. http://www.theregister.co.uk/2017/03/23/fake_base_stations_spreading_malware_in_china/.

[48] UDAR, S., AND BORGAONKAR, R. Understanding IMSI Privacy. https://www.isti.tu-berlin.de/fileadmin/fg214/ravi/Darshak-bh14.pdf, 2014.

[49] UNWIRED LABS. OpenCellID. http://www.opencellid.org.

[50] VODAFONE INDIA. Vodafone mPesa. https://www.mpesa.in/portal/.