

# Security and Privacy in Large-scale Infrastructure

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

**Doktor der Technischen Wissenschaften**

by

**Ing. Dipl.-Ing. Adrian Dabrowski, BSc**

Registration Number 0025933

to the Faculty of Informatics

at the TU Wien

Advisor: Privatdoz. Mag.rer.soc.oec. Dipl.-Ing. Dr.techn. Edgar R. Weippl

The dissertation has been reviewed by:

---

Thorsten Holz

---

Stefanie Rinderle-Ma

Vienna, 1<sup>st</sup> June, 2018

---

Adrian Dabrowski



# Acknowledgements

I am thankful to have found that one special person – I wish to marry – that accepted the circumstances of my career choice and the deprivations and demands entailed with it. Additionally, I thank my parents for enabling me my studies.

Furthermore, I could not have mastered these demanding and eventful times without Katharina Krombholz, Martina Lindorfer, and Martin Schmiedecker. I really appreciate your openness and the ability to give constructive feedback. You probably do not know how important your help and our support group is to me.

When I started this endeavor, I was very naïve: *Writing my Master's Thesis was quite fun, so why not do it again – this time bigger and with proper science? Could not be that hard, could it?*

Subsequently, I engaged in a roller coaster ride, where I had to grow technically, academically, and personally. I will leave it up to the committee to determine its sufficiency.

Additionally I learned, how enigmatic academic funding – especially in security – can be. Luckily, I had an advisor, Edgar Weippl, with profound expertise in navigating the rough high seas of funding. Without him, I would not have been able to explore so many different research directions and expand my horizons. I still fell short to learn that short-term gratification should not hamper my long-term goals. In general, I need to learn to say *no*.

Albeit demanding times, they were very diverse and highly interesting. I do not like to miss them.

A friend once told me, a thesis is finished once you lose all aspirations. Here it is.



# Kurzfassung

Große, kritische Infrastruktursysteme entwickeln sich nur langsam weiter und sind schwer zu modifizieren. Das ist in einer vernetzten Welt, die nach immer kürzeren Aktualisierungszyklen verlangt, ein Nachteil und unter Umständen gefährlich. Besonders große und komplexe Systeme können sich anders Verhalten, als das Einzelverhalten der Komponenten vermuten ließe. Dieses Zusammenspiel kann zu einer destruktiven Interferenz bei den Sicherheitseigenschaften des Gesamtsystems führen. Die vorliegende Dissertation untersucht verschiedene emergente und systemische Sicherheitsherausforderungen von besonders großen oder kritischen Systemen sowie von Systemen mit einer großen Installationsbasis auf der Ebene ausgewählter Komponenten als auch des ganzen Systems. Sie wirft einen Blick auf Sicherheitsprobleme und Lösungen von Stromnetzen, Mobilfunknetzen, Mehrdeutigkeit bei der Datenkodierung, Seitenkanälen in Webbrowsern, sowie die Privatsphäre von Bildern in sozialen Netzwerken. Sie beschreibt nicht nur Angriffe, sondern auch Entdeckungs- und Abwehrmethoden. Die Ergebnisse wurden in Standardisierungsgremien eingebracht, um die Sicherheit zukünftiger Systeme zu verbessern.

Für *Stromnetze* beschreiben wir neuartige Angriffe, die den physischen Teil dieser cyber-physikalischen Systeme als Angriffsvektor nutzen. Wir demonstrieren, wie stromkonsumierende Geräte synchronisierte Verbrauchsspitzen produzieren können, welche die Regelungsleistung überstrapazieren. Dies kann durch automatische Notabschaltungen zu großflächigen Stromausfällen führen. Für *Mobilfunknetze* beschreiben wir Methoden, um falsche Basisstationen (IMSI Catchers) und andere Angriffe zu erkennen – sowohl von der Kunden- als auch von der Betreiberseite. Wir konnten unsere Ansätze auf einem echten Mobilfunknetz mit 4 Millionen Teilnehmern testen. Für *mobile Browser* untersuchen wir Seitenkanalangriffe, die den Browserverlauf verraten. Zum Schluss beschäftigen wir uns mit der *Privatsphäre von Bildern* in sozialen Netzwerken und anderen Bild-Datenbanken mit dem Ziel, Individuen die Kontrolle über ihre Abbildungen zu ermöglichen.

Alle fünf Teile dieser Arbeit behandeln Sicherheits- und Datenschutzprobleme, welche aus der Komposition von Features, falschen Annahmen oder impliziten Vertrauen entstehen. Kompositorische Unsicherheit resultiert z.B. aus der Zusammensetzung von Methoden oder Standards, die sich in ihren Sicherheitseigenschaften gegenseitig destruktiv beeinflussen. Falsche Unabhängigkeitsannahmen können die operativen Grundlagen von Systemen bedrohen. Nicht überprüftes, implizites Vertrauen gibt Angreifern, die vorgeben eine privilegierte Komponente zu sein, unnötige Macht über ihre Opfer.



# Abstract

Large-scale and critical infrastructure systems face unique challenges: they are slow to evolve and difficult to modify, which is especially problematic in a connected world that demands ever shorter update cycles. Additionally, complex systems might behave unexpectedly compared to the behavior of their components. The latter can destructively interfere with security properties of the system as a whole.

This thesis explores different emergent and systemic security challenges of large-scale systems by looking at security problems and solutions of power grids, mobile cell phone networks, building blocks such as ambiguity in data encoding, side-channels in privacy-enhancing systems, and privacy in social networks. However, we do not solely describe attacks, but also engage in detection, mitigation, and defense efforts as well as standardization to improve the security of future systems.

The five systems and components have been studied as follows: In *power grids*, we describe novel attack techniques via the physical part of these cyber-physical systems. We demonstrate how connected devices can produce synchronized power-usage peaks that outperform the grid's reaction abilities and can lead to blackouts. In *mobile phone networks*, we describe the ability to detect fake base stations (IMSI Catchers) and other attacks from both the customer's and the operator's side. We were able to evaluate our approaches on a real mobile network with 4 million subscribers. On *mobile browsers*, we examine side-channels that can lead to the leakage of private browsing history. With social networks and other picture databases rapidly proliferating photographs of individuals at large scale as well as enriching those pictures with metadata, we explore the possibilities to empower individuals to *gain more control over their pictures*.

In all five parts of this thesis, we encounter security and privacy problems due to a composition of subsystems, wrong premises, or unverified trust. Compositional insecurity stems from multiple methods or standards that, if combined, interfere destructively on security properties or guarantees. Wrong independence premises destroy statistical requirements, threatening the operational foundations of systems. Unverified trust gives impersonating attackers needlessly power over their victims.

We discuss these challenges and their impact on large-scale systems on the level of selected components as well as for the whole system.





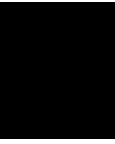
# Contents

<b>Kurzfassung</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Contents</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Power Grids . . . . .	2
1.2 Mobile Phone networks . . . . .	3
1.3 Mobile and Handheld Devices . . . . .	5
1.4 Individual Picture Privacy . . . . .	7
1.5 Other Publications . . . . .	8
<b>I Power Grids</b>	<b>11</b>
<b>2 Breaking the Independence Primitive</b>	<b>13</b>
2.1 Context . . . . .	14
2.2 Background . . . . .	15
2.3 Threat Scenario . . . . .	20
2.4 Evaluation: Power-Modulation . . . . .	24
2.5 Evaluation: Grid Effects . . . . .	28
2.6 Discussion . . . . .	34
2.7 Related Work . . . . .	37
2.8 Conclusion . . . . .	38
<b>II Mobile Phone Radio Networks</b>	<b>39</b>
<b>3 Background and Mobile Phone Networks Security Problems</b>	<b>41</b>
3.1 Overview and Introduction . . . . .	41
3.2 Mobile Network Background . . . . .	42
3.3 Systematization of Vulnerabilities . . . . .	48
3.4 Cause Overview . . . . .	55
	ix

3.5	Root Cause: Specification Issue . . . . .	56
3.6	Root Cause: Implementation Issue . . . . .	63
3.7	Root Cause: Protocol Context Discrepancy . . . . .	65
3.8	Root Cause: Wireless Channel . . . . .	67
3.9	Fake Base Stations Attacks in Detail . . . . .	67
3.10	Capabilities of Fake Base Stations . . . . .	68
3.11	Mobile Communications Acronyms . . . . .	74
<b>4</b>	<b>Client-side Mobile Phone Network Attack Detection</b>	<b>77</b>
4.1	Preface . . . . .	78
4.2	History and Motivation . . . . .	78
4.3	Background . . . . .	79
4.4	IMSI Catcher Artifacts and Detectability . . . . .	80
4.5	Catching an IMSI Catcher . . . . .	84
4.6	Implementation . . . . .	85
4.7	Results and Discussion . . . . .	90
4.8	Related Work . . . . .	93
4.9	Exposing Large Scale Denial of Service Attacks . . . . .	94
4.10	Future Work . . . . .	95
4.11	Conclusion . . . . .	96
<b>5</b>	<b>Operator-side Fake Base Station Detection</b>	<b>99</b>
5.1	Context . . . . .	99
5.2	Background . . . . .	100
5.3	Detection System Design and Data Sources . . . . .	102
5.4	Tracking IMSI Catcher . . . . .	103
5.5	Capturing IMSI Catcher . . . . .	107
5.6	Discussion . . . . .	112
5.7	Related Work . . . . .	114
5.8	Conclusion . . . . .	114
	<b>IIIMobile phones and terminals</b>	<b>117</b>
<b>6</b>	<b>Side-channels Uncovering Browsers' History</b>	<b>119</b>
6.1	Overview . . . . .	120
6.2	Motivation . . . . .	121
6.3	Background . . . . .	121
6.4	History Stealing in Wi-Fi Captive Portals . . . . .	122
6.5	Avoiding Minimalistic Browsers for Captive Portal Login . . . . .	124
6.6	Assessment of Applicability . . . . .	126
6.7	Proof-of-Concept Implementation . . . . .	128
6.8	Limitations and Future Work . . . . .	132
6.9	Conclusion . . . . .	132

<b>7</b>	<b>Introducing Decoding Ambiguity with Error-Correcting Codes</b>	<b>135</b>
7.1	Overview and Context . . . . .	136
7.2	Background . . . . .	136
7.3	Error Correction as a Hideout . . . . .	138
7.4	Implementation using Barcodes . . . . .	139
7.5	Experimental Results . . . . .	143
7.6	Discussion, Countermeasures and Future Work . . . . .	148
7.7	Conclusion . . . . .	150
 <b>IV Individual Picture Privacy</b>		<b>153</b>
<b>8</b>	<b>Fabric-based Passive Picture Privacy</b>	<b>155</b>
8.1	Preface . . . . .	155
8.2	Motivation . . . . .	156
8.3	Contribution . . . . .	157
8.4	Related Work . . . . .	157
8.5	Structure and Environmental Considerations . . . . .	158
8.6	Enforcement . . . . .	162
8.7	Technical Architecture . . . . .	162
8.8	Limitations . . . . .	167
8.9	Future Work and Enhancements . . . . .	167
8.10	Reception and Evaluation . . . . .	168
8.11	Proof of Concept Implementation . . . . .	171
8.12	Conclusion . . . . .	172
 <b>V Conclusion</b>		<b>175</b>
<b>9</b>	<b>Conclusion</b>	<b>177</b>
9.1	Systemic View . . . . .	177
9.2	Offensive versus Defensive Approaches . . . . .	178
9.3	Resume . . . . .	179
 <b>List of Figures</b>		<b>181</b>
 <b>List of Tables</b>		<b>185</b>
 <b>Bibliography</b>		<b>187</b>





# Introduction

Large-scale systems with a great number of dependent or independent sub-systems tend to behave in surprising ways – especially in corner cases. Security and privacy often are such corner cases: they are more than often not considered the core functionality and integrated later. However, considering security early on in the design process is no guarantee for a secure system, either.

In system theory, *emergence* describes larger, complex, or surprising phenomena caused by many smaller and simpler entities. They create new system properties by interaction with each other without a central orchestration or guidance. These effects are studied in biology, chemistry, multi-agent theory, and many other fields. For example, complex and symmetrical fractal snowflakes form from the much simpler flat hexagonal structure of the water ice molecular grid: the six corners are slightly more likely to encounter surrounding water molecules than the flat edges. Thus, new material is preferably deposited at those, forming new – even stronger attracting – peaks and eventually fractal snowflakes.

*Emergent insecurity* and *privacy leakage* can arise from many smaller entities interacting in a surprising or *unexpected* manner – *unexpected*, because the system designers did not anticipate it and those properties are absent in the individual entity, i.e., the designers' model of the comprising entities is incomplete. In a stable system, the individual benefits of system-preserving behavior must outweigh destructive behavior. Thus, selfish behavior substantially overlaps with the goals of the whole system. Additional resilience is achieved by the ability to tolerate a certain number of misbehaving entities.

The purest form of structured (destructive) collaboration is collusion. However, this typically involves beneficial properties for all colluding parties, i.e., a selfish benefit. In contrast, today many cyber-physical systems can be weaponized without the owners' knowledge in large quantities and certainly not to the owners' advantage.

Insecure effects can also emerge in large-scale systems through individual interactions of entities, which have not been anticipated by the designers. These specific interactions

can be triggered or otherwise be used by an attacker (e.g., as a side-channel, a protocol deficiency in a distributed system). If a weak specification causes a vulnerability, it can quickly affect a significant portion of entities – if not all.

Large decentralized, multi-stakeholder, or multi-vendor systems have another major weakness: fixing problems is not easy. Updates are often distributed as single-issue fixes to minimize side-effects. They must not break current protocols or behavior as not all systems will transition (i.e., upgrade) at the same time. If the vulnerability is not fixable without breaking backward compatibility, of systemic, or fundamental nature, mitigation strategies are employed. They can either significantly (e.g., stochastically) minimize the chance of an attacker to be successful in the first place, or detect an attack and react in real-time.

This thesis sheds light on multiple security and privacy aspects of large-scale systems: from the system as a whole down to its single components. We take a look at power grids, mobile phone networks from the operator to the individual participant, data encoding for portable hand-held devices, privacy in browsers, and social networks.

### 1.1 Power Grids

Power grids are a prime example of a large critical infrastructure, and their reliable operation became of utter importance for our economy and society. To stabilize the nominal frequency, power production and consumption have to be continuously kept in balance. As consumers are predominantly uncontrolled, operators have to continuously adapt power plants' output to the demanded power and predict future use with elaborated models including parameters such as weather, season, and time of the day. These models are based on the premise of a large number of small consumers averaging out their energy consumption spikes and thus forming a steady demand-change over time.

A power grid is a volatile system with constant changes in topology, usage, and thus flow patterns. Self-regulation and stabilization works only in narrow limits. A sophisticated technical and regulatory framework ensures a decentralized control system – probably the largest control systems that predate global communication networks (such as the Internet) by decades. However, runaway situations can easily trigger a cascade of secondary failures leading to large-area blackouts.

In Chapter 2, we show how insecurity emerges by breaking the fundamental assumption of all power grids: the independence of small-scale consumers. By gaining control over a large number of Internet-connected computers, IoT devices, and other electric devices (e.g., with a botnet), an adversary can stealthily modify their power consumption in a synchronized fashion. Such sudden load changes can outperform the power grid's countervailing mechanisms, i.e., primary and secondary reserve, and push the power grid into an unstable state. Thus, eventually triggering automatic load shedding or tie-line tripping for safety reasons, leading to large-scale blackouts. Reassembly of power-grids is a slow process, taking hours and sometimes days.

**Contribution:** This work is the first to demonstrate an attack on a large-scale cyber-*physical* system (CPS) over its *physical* interface instead of the electronic ones. It also demonstrates, that basically all non-mobile computing devices are part of a CPS through their power connection. We developed three attack scenarios: static, dynamic, and inter-control-zone attacks. The latter cleverly plays off the the independent secondary control systems (of adjacent control zones) against each other.

As real-world evaluation is – at best – ethically questionable, we simulated the results and derived key figures describing the quantity structure of this attack under different compositions of the infected device classes. The Matlab Simulink models based on Ulbig et al. [299] were released with the paper.

**Impact:** It sparked international response in media (e.g., Heise, futurezone, Der Standard). Furthermore, we got an invitation to an international meeting of reinsurance companies in Zürich, Switzerland to asses infrastructural threats.

This chapter is based on an extended version of our ACSAC 2017 paper:

- [93] Adrian Dabrowski, Johanna Ullrich, and Edgar Weippl. **Grid shock: Coordinated load-changing attacks on power grids.** In *Proceedings of the Annual Computer Security Applications Conference (ACSAC 2017)*. ACM, 12 2017.

**Context:** A follow-up paper estimating the effect of power-hungry Proof-of-Work cryptocurrency mining for a similar attack has been accepted to RAID 2018.

- [301] Johanna Ullrich, Nicholas Stifter, Aljosha Judmayer, Adrian Dabrowski and Edgar Weippl. **Proof-of-Blackouts? How Proof-of-Work Cryptocurrencies Could Affect Power Grids.** In *Research in Attacks, Intrusions, and Defenses: 21th International Symposium (RAID 2018)*. unpublished, to appear September 2018.

## 1.2 Mobile Phone networks

Mobile phone networks are a highly heterogeneous stack of networks and interfaces on multiple levels. On the customer-facing level, multiple radio front-end systems (GSM, UMTS, LTE) face a vast number of handsets from a large number of vendors and implementations. Likewise, the core network starting from the radio front-ends, continuing to the wide-area routing and signaling network, up to the gateways can also be built with systems of different vendors. And last but not least, mobile networks itself, form an international network to handle roaming. This system is held together by a large set of specifications and rule-sets for internal and external interfaces by the 3GPP, as well as operational guidelines of the GSMA.

In the past, a number of severe problems have been discovered in the specification and implementation of such networks. For example, some are based on race conditions

between different network messages, missing (mutual) authentication, the network stack's cross-layer information loss, joint key material usage by different algorithms, and global static identifiers.

However, corrections (i.e., modifications) to such a hyper-heterogeneous collection of handsets, radio interfaces, and core networks are not trivial. Instead of fixing classes of vulnerabilities, backwards compatibility requirements forced standardizers and manufacturers to specify and roll-out singular fixes and mitigations to minimize unwanted side-effects. In a few cases, feature were deprecated but accompanied with long transition periods.

Luckily – one might think – about once a decade a complete new network generation is released, allowing the specification to introduce new bold (security) concepts and non-backward compatible fixes and enhancements. Thus, improving general security, in contrast to just bare-minimum patching.

Unfortunately the opposite is true. New network generations kept being deployed, but the older ones never went away. Necessary interoperability between these network generations actually accumulated the problems or introduced new ones. Attackers can downgrade connectivity to older standards as handsets nearly universally support all three network generations with their plentiful extensions – even in regions that never experienced the older ones (e.g., Japan never deployed GSM).

Our work focuses on one of the most severe deficiencies of the 3GPP mobile phone standards: The possibility of fake cell phone towers (or fake base station, IMSI Catcher, Stingray) to selectively or at large attract legitimate customers. The top-down approach of telecommunication infrastructure, gives a (real or fake) higher network layer unprecedented control over the lower layer (i.e., the handset, smart phone, data modem). Fake base station are a fundamental step stone in a large number of low and high level attacks on mobile phone users. They can identify, track, localize, or cut off legitimate customers as well as intercept and manipulate data, text, and voice; or any service running on top of that (such as SMS-TAN or SMS-2FA). Furthermore, they can exploit the phone's baseband and SIM, reconfigure the handset or deliver localized text spam.

**Contribution:** We tackled this topic in a threefold approach:

**Categorization and Standardization** In Chapter 3, we analyze past mobile phone network vulnerabilities and categorized them in multiple ways. As a follow-up [263] we combined them with an analysis of current countermeasures and an identification of open research questions to take into the next generation of networks (5G). Thus allowing to fix whole classes of vulnerabilities not just individual problems.

**Customer-side detection and mitigation** In Chapter 4, we analyze client-side possibilities to detect fake base stations and their operational artifacts. We implement two distinct systems, one for smart phones and one based on stationary monitoring



stations to identify running attacks. Furthermore, we describe how to detect another denial of service attack, based on race conditions.

**Operator-side detection** As last approach, we analyze the possibilities of network operators to detect fake base stations operating inside the geographical region of their own networks. In Chapter 5, we propose methods based on analysis of hand-over transactions with shadow location tracking and consistency checks, as well as a method based on distinct latencies of different phone models. We evaluated our approach based on real network monitoring data from T-Mobile Austria.

Source codes of our client-side detection implementations were released together with the paper.

**Impact:** The first paper [91] (see list below) received wide academic recognition as well as media attention (including the Washington Post). Additionally, it won the *Best Student Paper Award* at ACSAC 2014. The subsequent paper on operator side detection still received considerable academic and media attention. Among other things, we received notification that our papers have been used in a criminal justice case in Canada. The analysis of vulnerabilities spanning multiple network generations was presented during the 5G-Security week at European Telecommunications Standards Institute (ETSI) in 2017, and at the International Telecommunication Union (ITU) in 2018 with the goal of improving the upcoming standard.

These chapters are based on updated and extended versions of our publications from ACSAC 2014, RAID 2016, and parts of an upcoming journal publication in IEEE Communications Surveys & Tutorials (CST).

- [91] Adrian Dabrowski, Nicola Pianta, Thomas Klepp, Martin Mulazzani, and Edgar Weippl. **IMSI-Catch me if you can: IMSI-Catcher-Catchers.** In *ACM Annual Computer Security Applications Conference (ACSAC)*, pages 246–255. ACM, 2014. doi:10.1145/2664243.2664272. **Best Student Paper Award.**
- [90] Adrian Dabrowski, Georg Petzl, and Edgar R. Weippl. **The Messenger Shoots Back: Network Operator Based IMSI Catcher Detection.** In *Research in Attacks, Intrusions, and Defenses: 19th International Symposium (RAID)*, pages 279–302. Springer, 2016. doi:10.1007/978-3-319-45719-2\_13.
- [263] David Rupprecht<sup>†</sup>, Adrian Dabrowski<sup>†</sup>, Thorsten Holz, Edgar Weippl, and Christina Pöpper. **On Security Research Towards Future Mobile Network Generations.** *unpublished, accepted with minor revisions to IEEE Communications Surveys & Tutorials*, 2018. Preprint available: <https://arxiv.org/abs/1710.08932>.  
<sup>†</sup>) Both authors contributed equally.

## 1.3 Mobile and Handheld Devices

On the low end of the cellular network we discover very similar security-threatening interactions between well-meant extensions to standards and multiple simultaneously

supported standards. They either amend each other in unfortunate ways or the not-standardized interplay of multiple, well-specified standards leads to undefined behavior and implementation specific ambiguities.

In web browsing, several extensions aim to improve the security of their users. We show that the interaction between cookies and the HTTP Strict Transport Security (HSTS) extension lead to a new side-channel allowing man-in-the-middle (MITM) attackers to enumerate web pages visited in the past (i.e., history stealing). The technique is so powerful that apart from interests and cultural background – in some cases – it is able to retrieve the real name of a device’s owner. The scenario especially affects mobile users in public Wi-Fi hotspots. Many employ a so-called captive portal, which is effectively a MITM-system by design.

Another very successful technology are barcodes. They are so successful, that a large number of different symbologies were developed. Some are used in a very narrow field (e.g., *Interleaved 2-of-5* code is used for baggage tags in commercial aviation, EAN and UPC codes for resale), others are used for a wide range of applications (e.g., PDF417, QR-Code, Code-128). For universality and convenience barcode readers (smart phone applications as well as dedicated hardware) understand multiple codes. For additional robustness, these codes include error detection and sometimes also error-correction mechanism. We demonstrate, that a combination of multiple standards and error-robustness can be used to embed multiple different messages within one message (in our demonstration, a 2D barcode). The introduced decoding ambiguity is not limited to barcodes. It is an general artifact occurring every time when a receiver employs a synchronizer to lock onto a signal without special provisions. In general, ambiguity is not desired, but in the security context of an application it might turn out dangerous: as different network layers, software modules, or communication parties might decode the same data to different payloads.

This part of the thesis at hand is based on the following publications.

- [83] Adrian Dabrowski, Isao Echizen, and Edgar R. Weippl. **Error-correcting Codes as Source for Decoding Ambiguity.** In *Proceedings of Workshops at IEEE Security & Privacy 2015, Workshop on Language-Theoretic Security (LangSec)*. IEEE, 05 2015.
- [87] Adrian Dabrowski, Katharina Krombholz, Johanna Ullrich, and Edgar Weippl. **QR Inception: Barcode-in-barcode Attacks.** In *Proceedings of the 4th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM 2014)*. ACM, 11 2014.
- [89] Adrian Dabrowski, Georg Merzdovnik, Nikolaus Kommenda, and Edgar Weippl. **Browser History Stealing with Captive Wi-Fi Portals.** In *Proceedings of Workshops at IEEE Security & Privacy 2016, Mobile Security Technologies (MoST)*. IEEE, 05 2016.

## 1.4 Individual Picture Privacy

Long before the General Data Protection Regulation (GDPR), it became apparent that collections of information fragments (or snippets) connected to each other on a large scale pose a threat to individual privacy. This effect can be observed in the wild at many places, of which social networks are one of the most popular. Snippets such as a photograph, time stamp, a geo-location tag, digital identities, friends, textual context, etc are linked together without the control or knowledge of the subject. In contrast to former book-based photo albums, search engines index the data and make them available for a broad audience. Even before GDPR, legislation in many countries demanded consent of affected subjects. This requirement turns out to be impracticable in most real-world situations and thus it is often ignored. This especially affects involuntarily or unintentionally captured individuals.

Our work aims to mend the analogue message or consent gap between depicted subject and the photo utilization processes by the photographer and/or social networks. An individual is able to (digitally) express their consent, lack thereof, or usage-restrictions of their own depiction. This information is decoded by the camera, the photographers software, the newspaper publisher, or the social networks prior to processing, data-enrichment, linkage, indexing, or publication.

Over the last years, we explored several different possible ways to transmit consent and usage restrictions in a machine-readable way. Chapter 8 gives a detailed description of our passive fabric based approach, where the consent information is unobtrusively encoded into the fabric pattern or print on cloths. It has to overcome multiple challenges, such as wrinkling, uneven illumination, picture compression artifacts, and camera distortions. Further, we conducted a user study and implemented a proof-of-concept in Matlab. Albeit, not included in this thesis, we also explored other solutions such as an active smartphone-based implantation in which subjects can decide on a case-by-case basis. The prototype was build upon decentralized techniques such as Wi-Fi-Direct.

**Impact:** The different ideas were explored, implemented, and evaluated in cooperation with Prof. Isao Echizen from the National Institute of Informatics, Tokyo, Japan and Prof. Peter Purgathofer, Technische Universität Wien. For a complete list, see the *Context* below. We received a total of € 72.000 non-academic grant money from Internet Privatstiftung Austria (IPA) and an invitation to Ars Electronica Festival 2015.

- [95] Adrian Dabrowski, Edgar R. Weippl, and Isao Echizen. **Framework Based on Privacy Policy Hiding for Preventing Unauthorized Face Image Processing.** In *Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC2013*, pages 455–461. IEEE SMC, 10 2013. DOI 10.1109/SMC.2013.83. doi:10.1109/SMC.2013.83.

**Context:** The work above is a slice from a much broader field we were contributing to. The following first- and co-authored papers are suggested for context but not part of this

thesis.

- [179] Katharina Krombholz, Adrian Dabrowski, Matthew Smith, and Edgar Weippl. **Exploring Design Directions for Wearable Privacy.** In *Proceedings of USEC Mini Conference 2017*. Internet Society, 02 2017. URL: <https://www.internet-society.org/doc/exploring-design-directions-wearable-privacy>.
- [198] Shimon Machida, Adrian Dabrowski, Edgar Weippl, and Isao Echizen. **Privacytag: A Community-based Method for Protecting Privacy of Photographed Subjects in Online Social Networks.** In Arpan Kumar Kar, P. Vigneswara Ilavarasan, M.P. Gupta, Yogesh K. Dwivedi, Matti Mäntymäki, Marijn Janssen, Antonis Simintiras, and Salah Al-Sharhan, editors, *Digital Nations – Smart Cities, Innovation, and Sustainability*, pages 261–275, Cham, 2017. Springer International Publishing. **Best Paper Award.**
- [88] Adrian Dabrowski, Katharina Krombholz, Edgar Weippl, and Isao Echizen. **Smart Privacy Visor: Bridging the Privacy Gap.** In *Proceedings of Workshop on Privacy by Transparency in Data-Centric Services (PTDCS) at 18th International Conference on Business Information Systems (BIS2015)*, Poznan, Poland, 2015. Springer.
- [180] Katharina Krombholz, Adrian Dabrowski, Matthew Smith, and Edgar R. Weippl. **Ok Glass, Leave me Alone: Towards a Systematization of Privacy Enhancing Technologies for Wearable Computing.** In *Proceedings of Workshop on Wearable Security and Privacy co-located with Financial Cryptography and Data Security 2015*, 01 2015.
- [165] Alberto Escalada Jimenez, Adrian Dabrowski, Noburu Sonehara, Juan M Montero Martinez, and Isao Echizen. **Tag Detection for Preventing Unauthorized Face Image Processing.** In *Proceedings of the 13th International Workshop on Digital-Forensics and Watermarking (IWDW 2014)*. LNCS, Springer, 10 2014.

## 1.5 Other Publications

Other peer-reviewed publications of the author.

- [84] Adrian Dabrowski, Markus Kammerstetter, Eduard Thamm, Edgar Weippl, Wolfgang Kastner **Leveraging Competitive Gamification for Sustainable Fun and Profit in Security Education** In *USENIX Summit on Gaming, Games and Gamification in Security Education (3GSE)*, co-located with USENIX Security Symposium 2016
- [94] Adrian Dabrowski and Edgar R. Weippl **Mobile Phone’s Wi-Fi Presence for Continuous Implicit Secondary Deauthentication** In *11th International Conference on Passwords (Passwords 2016)*
- [178] Christian Krieg, Adrian Dabrowski, Heidelinde Hobel, Katharina Krombholz, and Edgar Weippl **Hardware Malware** In *Synthesis Lectures on Information Security, Privacy, and Trust*, DOI 10.2200/S00530ED1V01Y201308SPT006, ISBN paperback 978-1627052511, ISBN ebook 978-1627052528
- [300] Johanna Ullrich, Katharina Krombholz, Heidelinde Hobel, Adrian Dabrowski, and Edgar Weippl. **IPv6 Security: Attacks and Countermeasures in a Nutshell.** In *USENIX Workshop on Offensive Technologies (WOOT)*. USENIX Association, 2014.





Part I

Power Grids





# Breaking the Independence Primitive

Electric power grids are among the largest human-made control structures and are considered as critical infrastructure due to their importance for daily life. When operating a power grid, providers have to continuously maintain a balance between supply (i.e., production in power plants) and demand (i.e., power consumption) to keep the power grid's nominal frequency of 50 Hz or alternatively 60 Hz. Power consumption is forecast by models refined over decades including multiple parameters like weather, season, and time of the day; they are based on the premise of many small independent consumers averaging out their energy consumption spikes.

In this chapter, we develop attacks violating this assumption, investigate their impact on power grid operation, and assess their feasibility for today's adversaries. In our scenario, an adversary builds (or rents) a botnet of zombie computers and modulates their power consumption, e.g., by utilizing CPU, GPU, hard disks, screen brightness, and laser printers in a coordinated way over the Internet. Outperforming the grid's countervailing mechanisms in time, the grid is pushed into unstable states triggering automated load shedding or tie-line tripping. We show that an adversary does not have to rely on smart grid features to modulate power consumption given that an adequate communication infrastructure for striking the (legacy) power grid is currently nearly omnipresent: the Internet to whom more and more power-consuming devices are connected.

Our simulations estimate that between 2.5 and 9.8 million infections are sufficient to attack the European synchronous grid – depending on the mix of infected devices, the current mix of active power plant types, and the current overall produced power. However, the herein described attack mechanisms are not limited to the European grid.

This chapter is based on our 2017 ACSAC paper [93]. It is structured as follows: Section 2.2 provides background on today's power grids from an engineering perspective.

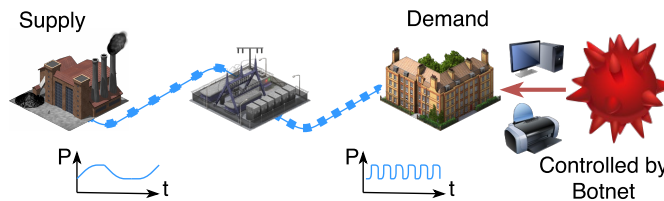


Figure 2.1: Visualization of Attacks 1 and 2: The botnet can modulate the power demand much faster than power plants can react.

Section 2.3 describes our attack scenarios and the anatomy of the adversary’s botnet for these attacks. It goes without saying that such an attack can never be ethically tested on a real power grid. Thus, we measure the capabilities for load modulation of a zombie and its peripherals in Section 2.4 and use simulations to predict the impact of large load changes on the power grid in Section 2.5. In Section 2.6, we combine the gained results into multiple scenarios and assess the number of infections needed considering parameters like time of the day, season, etc. Section 2.7 discusses related work.

## 2.1 Context

Electric power grids are among the largest human-made structures and by far the most important for technology-dependent societies. Without electricity, life as we know it would not function; there would be breakdowns in water and food supply, transport, medical aid, and communication infrastructures. For this reason, power grids are considered critical infrastructures, and operated with a high level of care to provide qualitative service, i.e., constant voltage and frequency. At the same time, power grids are legacy systems pre-dating modern telecommunication networks — such as the Internet — by decades, as is reflected in its structure: Electricity consumers are predominantly uncontrolled, i.e., consuming electric power whenever they need thereby causing fluctuations in consumption. However, on a macro scale fluctuations average out: for each consumer turning a light bulb off there is most likely another one turning the light on. Energy suppliers have developed sophisticated models that reliably forecast power demand in dependence of time of the day, week day, season and many other parameters allowing (centralized) power plants to trace actual consumption best possible in order to keep the equilibrium of production and consumption; the remaining gap is placed at disposal by so called *control reserves* (*spinning reserve* in the U.S.), i.e., the activation of power plants in stand-by.

Power grids around the globe currently undergo substantial modifications commonly summarized under the term *smart grid*, and the included concepts put an end to the strict separation of controlled production and uncontrolled consumption. On the one hand, renewables like wind turbines and photovoltaics provide electric energy in dependence of weather conditions and are thus only to a certain extent predictable, not to mention arbitrarily controllable. On the other hand, demand-side management aims to shift certain types of consumption, e.g., heating or cooling, in time. Synchronized over a

communication channel, energy should then be consumed at the time of production by renewables. Due to such remote control of high amounts of power consumption, the smart grid is considered to be vulnerable to direct cyber attacks aiming to destabilize the system [191, 335].

In this chapter, we show that an adversary does not have to rely on explicit (or future) smart grid features to modulate power consumptions, as the communication infrastructure to attack the *legacy grid* is already available: the Internet. An adversary might compromise a large number of Internet-facing power-drawing devices, e.g., computers, TVs, or thermostats controlling heating systems, and modulate their power consumption in a coordinated way (Figure 2.1<sup>1</sup>). As these fluctuations are at a large scale, fast and unpredictable, power plants are not able to trace power consumption any more causing an imbalance of production and consumption and eventually load-shedding, disconnection of power plants, disconnection of transmission lines, or a split of one synchronous power grid into multiple areas. Our attack benefits from the fact that the power grid is substantially slower in reaction than information technology, and will become even more vulnerable in the future, as controllable power consumption (with a potentially low level of security protection) increases due to the spread of the Internet-of-Things (IoT).

In this chapter, we focus primarily on the synchronous grid of Continental Europe (also known as UCTE grid) as it is the largest of its kind spanning over 23 countries, including large parts of Europe, North Africa as well as Turkey, and cite the respective UCTE/ENTSO-E policies. While terminology and details might differ in other synchronous grids, e.g., in the United States, we want to stress that the general principles, attacks, and conclusions apply to AC power grids all over the world.

## 2.2 Background

This section provides background on the power grid from an engineering perspective and an introduction into control theory, discussing feedback loops and resonance frequencies.

### 2.2.1 Producer-Consumer Equilibrium

Electric power cannot be stored at large scale, i.e., must be generated and consumed at the same time. In consequence, the challenge when operating a power grid is to maintain an equilibrium of electric power supplied by power plants and power consumed by electric loads. Apart from a few consumers with extraordinary high consumption — e.g., aluminum foundries and steel mills — are uncontrolled, i.e., they turn their power consumption on and off whenever they need or feel to. Thus, keeping a balance between supply and demand has become the suppliers' tasks – their power plants' production has to trace current consumption.

Scheduling power plants in order to deliver enough electric power at all times is a non-trivial task, which is fulfilled by applying a two-fold approach: elaborated models were

<sup>1</sup>Isometric icons from the *Lincity-ng* project in Figure 2.1 distributed under CC-BY-SA-v2.

Table 2.1: Emergency routines in case of under-frequency in Germany [313, p65] similar to the ENTSO-E policies [303, p26]

	<i>Frequency</i>	<i>Action</i>
1	49.8 Hz	Alerting, activation of plants, shedding of pumps
2	49.0 Hz	Load-shedding of 10-15% of total load
3	48.7 Hz	Load-shedding of further 10-15% of total load
4	48.4 Hz	Load-shedding of further 15-25% of total load
5	47.5 Hz	Disconnection of all power plants

developed describing overall power consumption in dependence of type of load (commercial or residential), time of the day, week day, season, weather and many parameters more allowing a quite accurate prediction of power consumption. Secondly, the remaining gap is handled by control reserve, i.e., additional power production capacities that are kept in stand-by and activated if needed [303, 304].

If production and consumption are imbalanced, frequency deviates from its nominal value  $f_0$  (in Europe  $f_0 = 50 \text{ Hz}$ , in the US  $f_0 = 60 \text{ Hz}$ ): If there is more supply than demand, the frequency increases; if there is less supply than demand, the frequency decreases. This happens, because large spinning turbines produce the vast majority of electricity in today's power grids and store rotational energy, i.e., kinetic energy due to rotation. In case of over-supply, conservation of energy produces additional torque on the generator's spinning axis and accelerates the turbine, i.e., energy supplied to the turbine is converted into mechanical energy instead of electric energy. As the turbine speed and the grid frequency are rigidly coupled, the grid frequency increases as well. Vice versa, higher power consumption slows down the generator due to a counter-torque on the spinning axis and lowers the output frequency. In fact, a grid's frequency deviation  $\Delta f = f - f_0$  with  $f$  being the current value is used as the primary indicator for an imbalance in demand and supply and triggers the control reserve, bringing the power grid back into equilibrium.

Due to minor imbalances, frequency is fluctuating around the nominal value even under normal operational conditions due to minor imbalances. If deviations are larger than a pre-defined threshold, emergency routines are performed to bring the power grid back into balance. For example, German regulations define a five-step plan for load-shedding in case frequency drops under certain values [122], see Table 2.1. These routines protect turbines and other physical devices from damage, e.g., due to resonant frequencies.

### 2.2.2 Continental Synchronous Grid Area

Historically, power grids were "islands" with a single power generator which were then stepwise integrated into larger grids for reasons of reliability and costs. Also, consumption

spikes are likely to be handled better by multiple power plants. Cheaper (but typically less controllable) power plants are able to produce the base load, more expensive (and dynamic and more controllable) plants handle peak loads. Nowadays, networks are operated on a national, even continental level.

A parallel operation of generators requires coherence, i.e., operation at exactly the same frequency and in phase, leading to *synchronous grid areas*. Misalignment, e.g., in extreme case, one generator is at the positive peak of a sine, while another is at the negative peak, will result in major short-circuit like currents potentially leading to fire or physical destruction. The biggest synchronous area is the *continental synchronous grid area*, also called *synchronous grid of continental Europe*, comprising most of the European Union, Switzerland, many Balkan countries as well as three North African states; there are also plans for further expansion. This implies that the sine at a power plug in Athens, Greece is the same as another one obtained in Lisbon, Portugal or Tunis, Tunisia. It has a total production capacity of more than 600 GW and a nominal frequency of 50 Hz.

The continental synchronous grid is organizationally split into control zones which are led by a transmission system operator (TSO) [305]. Control zones are the size of a smaller European country like Austria or Switzerland and mostly follow national borders or geographical landmarks. Larger countries are split into multiple control zones, e.g., Germany has four. Control zones have connections with adjacent zones via transmission lines. However, their capacity covers only a fraction of the power consumption and is mostly meant for the compensation of power imbalances.

TSOs are unified in the *European Network of Transmission System Operators for Electricity (ENTSO-E)* which defines regulations on how to jointly operate the grid. Among these regulations, ENTSO-E specifies the provision and application of control reserve in three steps to balance production and consumption, namely *primary*, *secondary* and *tertiary control* as described in the following paragraphs [303,304].

On the physical level, before any control system kicks in, the rotational energy stored in the spinning turbines stabilizes the frequency to a certain extent.

**Primary control** is activated within seconds after an incident – i.e., frequency deviation is exceeding a certain threshold – and the first to actively react to a power imbalance. Primary control is applied in proportion to the frequency deviation, i.e.,  $K \cdot (f - f_0)$ , and does not bring the frequency back to nominal, it rather stabilizes the frequency at a stationary value. In practice, a control system (governor) within the power plant observes the grid frequency and decides whether to increase or decrease power output. In primary control, all generators in the synchronous area participate simultaneously.

**Secondary control** is activated after 30 seconds and takes some minutes until full activation. Its task is to replace primary control and return the frequency to its nominal value. This type of control reserve has to be carried out by the TSO whose control zone is imbalanced. The respective zone is recognized by the *Area Control Error (ACE)* which

is calculated for each zone according to Equation 2.1.

$$ACE = P_{measured} - P_{planned} + K \cdot (f - f_0) \quad (2.1)$$

$P_{measured}$  is the sum of measured power transfers on transmission lines,  $P_{planned}$  the sum of planned power exchanges with adjacent zones, and  $K$  is the network power frequency characteristic of the primary control. If all produced primary control is exported into other control zones, ACE is zero and secondary control remains inactive in the respective area. If the imbalance occurred in its own zone, a TSO's ACE differs from zero and secondary control is initiated.

**Tertiary control** frees up resources from primary and secondary control after their sustained activation. In contrast to the prior two control mechanisms, tertiary control also allows for manual intervention by the TSO.

### 2.2.3 Feedback Loops and Resonance Frequencies

Control theory distinguishes open-loop systems from closed-looped systems. In an open-loop system, the controller aims to achieve the output reaching a set point without monitoring the output; in consequence, accurate system models are necessary while still not being able to adapt in case of unexpected disturbances. Meanwhile, closed-looped systems are measuring the system's output  $y$  (e.g., via a sensor), comparing it with the set point  $w$  and reacting upon the control deviation  $e = w - y$ . The output counteracts the deviation from the set point; this behavior is also known as *negative feedback*. This way, a disturbance influencing the output is measured, and counteracted.

Closed-loop controls frequently incorporate delays, as it takes time to measure, calculate and physically react, e.g., when accelerating physical masses. This implies that feedback is not instantaneous and the system might swing when excited at certain frequencies. A signal's phase shift is dependent on the delay, and a shift of 180 degrees changes negative feedback into positive. The feedback does not counteract the deviation anymore, but rather reinforces it, leading to self excitation and an increasing amplitude. Such a situation is potentially damaging and thus to be avoided; as a rule of thumb, the control should be faster than the monitored physical system.

Linear controllers exhibit proportional (P), integral (I) or derivate (D) behavior as well as respective combinations: Proportional control amplifies the control deviation  $e$  by a constant factor, integral control integrates the control deviation  $e$  over time, and derivate control differentiates. Proportional control shows permanent control offset, i.e., the output differs from its intended value by some offset. If the latter is undesired, proportional control has to be combined with integral behavior, forming a PI controller.

Power imbalance influences a grid's frequency; there are multiple controls reacting on frequency shifts, i.e, closed-loop controls [303, 304]. Load, in particular from induction engines, increases/decreases with frequency and thereby automatically reduces power imbalance. This effect is known as *self regulation of loads*, and is assumed to be 1%/Hz

in the continental synchronous grid. In addition, there are the operational measures of primary, secondary and tertiary control, rescheduling power production facilities. Primary control is specified to show proportional behavior, i.e., it cannot return frequency to its nominal value of 50 Hz, whereas, secondary is a combined proportional, integral (PI) controller returning the frequency to its nominal value. Both show delays, i.e., primary control reacts typically within a few seconds and secondary control within 30 seconds, replacing primary control, vulnerable to self-excitation. Since tertiary control can be manually scheduled, its behavior cannot be specified in a similar manner.

#### 2.2.4 Failure Modes

Power grids typically adhere to a *N-1 criteria* (or N-1 redundancy).  $N$  is the number of all installed components, whereas  $N - 1$  is sufficient to manage a situation (current or planned). This is a measure to successfully cope with simple operational disturbances. Nonetheless, power grids tend to cascading (or chain) reactions that threaten grids on a continental scale that elude simulation models.

The main difficulties with failure management in power grids are:

- Electrical power can not be routed such as water; it flows in all lines simultaneously in relations to their resistance.
- If the operation deviates too much from the set-point, the grid becomes uncontrollable, sending more and more parts of it into shut-down
- Cascading effects are hard to control and to predict. Ripple effects can take effect within seconds.

##### 2.2.4.1 2003 U.S. & Canada Northeast Blackout

On August 14 [307], the second largest grid failure in history affected 55 million people in Ontario, Canada, and eight northeastern U.S. states. At 12:15 p.m., incorrect telemetry data, triggers a software bug, preventing Ohio-based grid operator to receive alarms on the grid's condition. During the next three hours, one power plant shuts down, and one overloaded line tripped due to sagging lines touching a tree. At 2:32 p.m. a second line, now overload itself, sagged into a tree and disconnected. Within minutes, a cascade of an increasing number of tie lines breaking down under the heavy load started a disastrous chain reaction. In some areas, as customers got disconnected and load fell off the grid, generators started spinning faster, going out of sync and being shutdown itself. Large electric currents trying to find their way through the grid tripped one line protection after another. Until the end of the ripple effect at 4:13 p.m., 256 plants went off-line. It took two days to restore the majority of the grid, and up to a week for some remote areas.

### 2.2.4.2 2006 European Grid Split and Blackout

On November 4th [306], two transmission lines crossing the Ems river near Hamburg were scheduled for a brief disconnect to enable an over-sized cruise ship to be put to sea from a shipyard upstream. However, the schedule changed last-minute and the operator E.ON had no time to change contractual energy transfer operations. At 9:39 p.m. the second line crossing the Ems-River was taken off-line, increasing the load on a shared line between E.ON and RWE, another grid operator in Germany. Both operators were unaware that they configured different warning and trip points on both ends of the line (3,000 A vs. 2,100 A). Thus, flow and safety calculations were wrong and in an uncoordinated attempt by E.ON to create another interconnect at a distribution center generated unanticipated high loads. Two seconds later, the distribution center went off-line due to tripping overload protection circuits. The flows rerouted uncontrolled over different grid segments. Within the following 17 seconds, a cascade of failing transmission lines tore the European grid into three distinct networks. The fault lines cut right through the control zones of Germany and Austria, making their control even harder. Suddenly, the northern part had an overproduction of 10,000 MW, while the two southern part of the network were missing that energy. The networks began to desynchronize rapidly, taking engineers many hours and several attempts to resynchronize.

In the southern grid load-shedding automatically disconnected consumers from power, while in the north-eastern part, power plants were shut down, affecting about 10 million customers. Additionally, large-scale infrastructure, such as trains, were profoundly affected.

## 2.3 Threat Scenario

For our attacks, we assume a botnet controlling a high amount of computers and their peripherals. Each bot can trivially modulate the power consumption of the CPU, the GPU (Graphics Processing Unit), hard drives, and the screen backlight. Laser printers — an peripheral common — are also large power consumers due to the high temperatures used in their fusion units. In some cases, the botnet might find other locally accessible Internet-of-Things (IoT) devices on the network, which often incorporate less security protection or default passwords, for load modulation.

While each of the devices only contributes several hundred to thousands Watt, their effect multiplies by the botnet producing a large leverage on power consumption within the grid. It can modulate this power consumption in a coordinated fashion and in a sub-second range. This way, the adversary aims to negatively affect the power grid.

In the first part of this section, we introduce different kinds of load modulation attacks. In the second part, we specify the botnet in detail.

### 2.3.1 Attack Types

We consider an attack successful if of the following effects occurs:



- Customers or power plants become disconnected from the grid, e.g., by automatic load shedding due to under-frequency or frequency protection protocols for power plants.
- Transmission lines (tie lines) become disconnected, e.g., by overload-protections, or adjacent control zones become disconnected.

### **Attack 1: Static Load Attack**

The attacker increases the power consumption of all bots to the maximum; this action shifts power generation and consumption out of the equilibrium by increasing the consumption faster than the producers can react. Just a brief violation of the frequency thresholds, triggers load shedding (see Table 2.1), i.e., the automatic disconnection of parts of the grid. To enlarge the amplitude of load changes, the adversary might piggyback their attack on power spikes and oscillations that usually happen in the grid [163, 185]. This attack targets the *primary control*.

### **Attack 2: Dynamic Load Attack**

Closed-loop control systems with negative feedback and non-zero latency tend to over- and undershoot when reacting to changes. This effect can be used to increase the amplitude of Attack 1 by measuring the reaction times and modulate the power consumption so that the highest production peak is met with a low modulated demand and vice versa. Since the attacker is reacting on the grid, s/he needs a return channel to measure the state of the grid, i.e., the current frequency. In particular, the adversary increases the load to the maximum and waits for the full primary control to be activated; then, decrease the load to the minimum wait for the primary control to deactivate, and so on. This attack targets the behavior of the primary control.

Eventually, the attacker might find a resonant frequency that leads to a much larger frequency swing than appropriate for the load change. The ENTSO-E synchronous area is known to have eigenfrequencies that manifest in several post-incident reports [115], [306, p.77], [185, p.3].

### **Attack 3: Inter-Zone Attacks**

This attack aims to trip tie lines that are connecting areas by putting large loads on them. A naïve way to increase the load on a tie line is to find a line that is operating near the maximum and increase power consumption in the target area of that transfer. Some TSO's publish their line state on the web [32]. Even though they are delayed in time, it gives an attacker a good insight on when the line is usually loaded the most. However, since primary control detects the increase in load, a part of the additional load will be produced in the targeted area (control zone), leaving only the rest to the tie line.

Reducing power consumption in one area while simultaneously increasing it in the target area would further increase the burden on the line, but decreasing load (of mostly idle

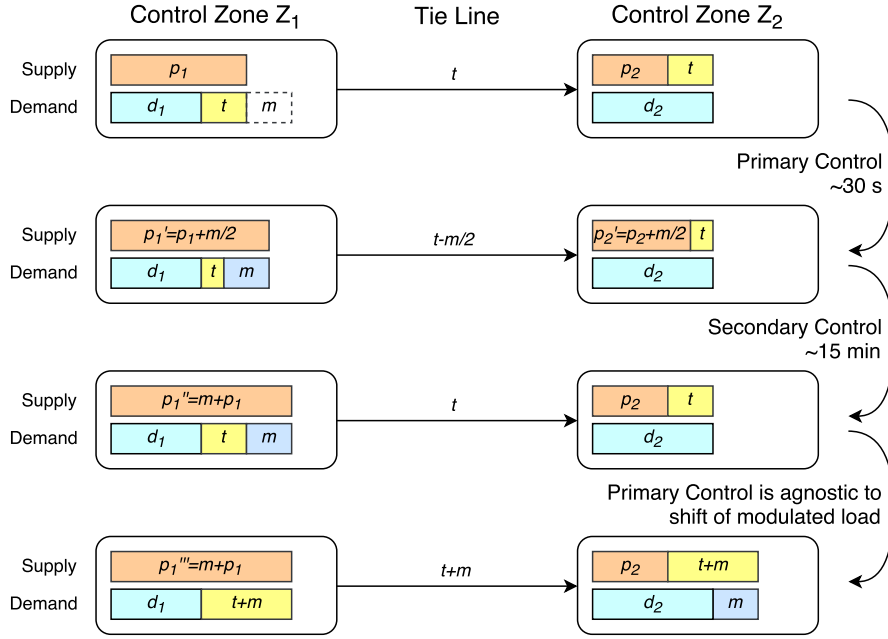


Figure 2.2: Simplified schematic of attack 3

electronic appliances) is only possible in rare cases. However, the attacker can wait for the automatic secondary control to equalize for changes between both zones to meet scheduled transfers; then change the load modulation between zones to achieve the same effect.

Figure 2.2 depicts the scheme step by step. First, load is added to Zone  $Z_1$ , effectively lowering the transmission on the line (in- and outgoing transmissions cancel each other out). However, secondary control will compensate for the overproduction in one zone and the underproduction in another and adjust output power accordingly to meet the scheduled transfer on the line. The attacker waits until this happens and inverts the modulation between the zones, recreating the imbalance with reversed sign, again triggering substantial compensation currents over transmission lines. For simplicity, we assume  $m_1 \approx m_2 = m$ , so that an extra of  $m$  is added to the transmission line. Since the total load of the grid does not change, the primary control will not kick in.

### 2.3.2 Anatomy of a Grid-Attacking Botnet

A botnet is a set of hijacked computers (called *bots* or *zombies*) on the Internet that is set up to perform tasks on behalf of the botnet owner [259]. Among other, botnets gained infamous popularity by traffic-based denial-of-service attacks, mass-hacking, sending spam, spying on the computer owners, online fraud, mining crypto-currencies, stealing secrets from presidential candidates, and infecting other computers. Some botnets operate for years until they are detected. The following paragraphs provide details on how an

adversary is able to built an adequate botnet for power-load attacks.

### Acquisition

Prices of botnets vary depending on the country the zombies are placed in. A 2013 report [99], named USD 1,000 for 10,000 U.S.-based bots, and between USD 400 and USD 600 for European-based bots. Large botnets contain up to tens of million devices [246].

### Synchronization

For power grid attacks, a timely communication structure is in order to coordinate precise load manipulations. Modern protocols such as NTP [209,343] compensate for round-trip time, delivering sub-millisecond performance if allowed to run for extended periods of time [210].

### Geographical Estimation

For our attacks, the botnet has to coarsely estimate the position of the zombie machines. For attacks 1 and 2 the granularity can be as low as the continent as central Europe is an interconnected supergrid. For attacks on the US grid, the granularity should be at least on state level as there are multiple synchronous grids. There are various ways to identify the geographical position of a bot:

- *GeoIP lookup*: Maxmind [203] and other databases provide at least a state/country level localization – even in the free version.
- *Wi-Fi localization*: Coarse location by BSSIDs of Wi-Fi access points is now a standard technique for mobile phones. Some stand-alone PCs certainly almost all notebooks come with a Wi-Fi receiver. Some databases are available free of charge [326].
- *Keyboard layout*: Malware such as the Conficker worm [76,246] uses the keyboard layout to determine the country of the computer to avoid targeting the own country. This works on language-fragmented continents like Europe, but not in North America.

### Frequency Measurement

Attack 2 and 3 (Section 2.3.1) benefit from the frequency feedback channel. In case the attacker and bot-master is sitting anywhere within the attacked grid, s/he can invest into a low-cost power grid frequency measurement unit, such as from open-source projects [79,97], measuring the frequency at an ordinary power outlet. Since the frequency is identical in all parts of the network (until it breaks up), one measurement station is sufficient. Attacks on remote grids might approximate measurements by analyzing audio/microphone hum, or Webcam light flickering on target machines — similar to its

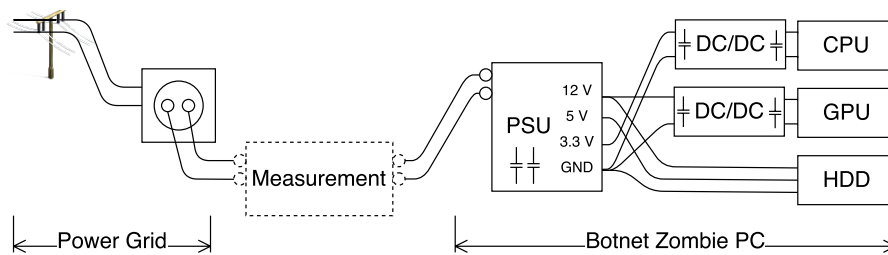


Figure 2.3: Model of botnet zombie and method of measurement

use in multimedia forensics [80, 170]. Furthermore, some websites [98, 105, 129, 163, 220] offer live data for certain grids.

## 2.4 Evaluation: Power-Modulation

To understand the attack and estimate the effects we have to answer two questions. First, to which amount can a bot zombie influence its power consumption and at which pace. Secondly, use simulations to predict the outcome of such a load attack on the power grid. The former is described in this section, the latter in Section 2.5.

In a lab experiment, we measured a bot's capability for software-driven load modulation. In a first step, we analyzed the dynamics of a PC's load increase/decrease in order to determine their capabilities for fast load changes. Then, we categorized different types of devices that might become part of our attack and investigated the increase of load from an idle to a fully utilized state.

### 2.4.1 Electric Model of a Load-Altering PC

Since PCs and servers appear to have great potential for load control, we took a closer look and asked how fast they can modulate their power consumption.

The components of a PC (or Server) do not directly draw power from the mains. Instead, a series of power conversions takes place before reaching the relevant components, i.e., CPU, GPU and hard drive. Our model is depicted in Figure 2.3: We must assume that each conversion step through the power supply unit (PSU) and subsequent DC/DC converters incorporates power-stabilizing capacitors which will dampen the artificially produced load spikes. To measure the effective load amplitudes and times as dispensed to the grid, we had to measure at the power socket (Figure 2.4).

As a conservative assumption of an office PC, we chose an Intel Core2 Duo (Figure 2.5). For a high-end gaming PC we tested an Intel i7-6700 with an NVIDIA Geforce GTX 1070 graphics card (Figure 2.7). Both were connected with an LG 24" TFT screen which was measured separately (Figure 2.6). On Linux, we used command line tools `hdparm -t` for inducing stress to hard disks, `stress -c` for the CPU, and `glmemperf` for the GPU. On Windows, we used ZCPU for CPU stress and 3D Mark to measure the GPU.

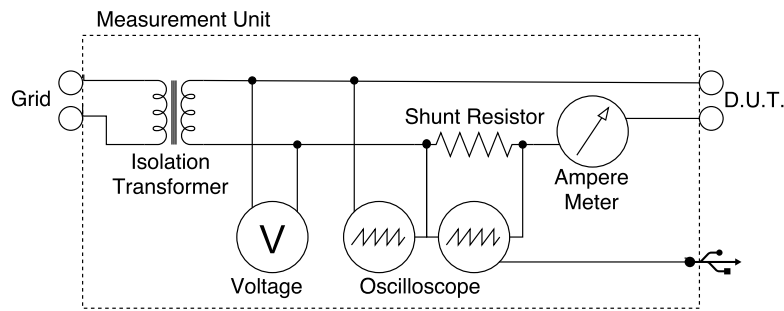


Figure 2.4: Measurement setup in detail

As expected according to our model, the capacitors soften the steep current edges, especially in the low-end range. Thus, the PSU in the old office PC ramped up the consumption within 2-3 AC cycles, i.e., 40-60 ms. In contrast, the gaming PC can multiply its power consumption and the PSU ramps up the usage within a single AC cycle. Hard disk consumption turned out to be negligible: most of the power is used for the disk rotation which is independent from head movements.

Laser printers are without question the heaviest power consumers of all computer peripherals due to the high temperatures involved when fixating the toner to the paper. The fuser's surge current is a multiple of its already high power consumption (Figure 2.8a). In our setup with a small office/home office (SOHO) printer, the heat-up process started within a second when printing via USB, and several seconds when printing over the network. The high power usage continues for 8 seconds for the first page and 5 seconds for all following pages. On stand-by, the printer reheats the fuser every 35 seconds, until it goes to sleep mode after several minutes.

Screens can easily be turned on and off via software as operating systems offer power saving controls and appropriate APIs. As seen in Figure 2.6, the screen first displays a goodbye message (3 seconds), then goes into time-out mode (5 seconds) and finally to

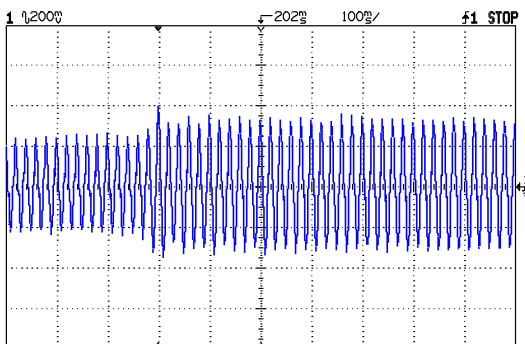


Figure 2.5: Example: low-end office PC. PSU ramps up power within 2-3 AC cycles

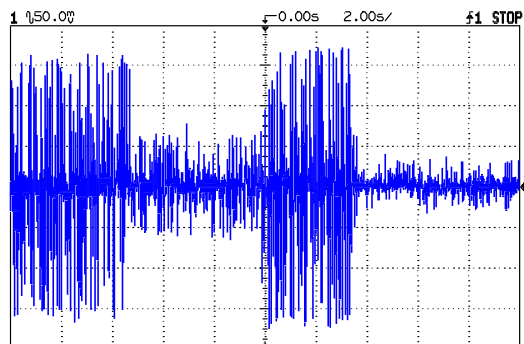


Figure 2.6: LG 24" TFT screen needs 8 s before going to sleep mode.

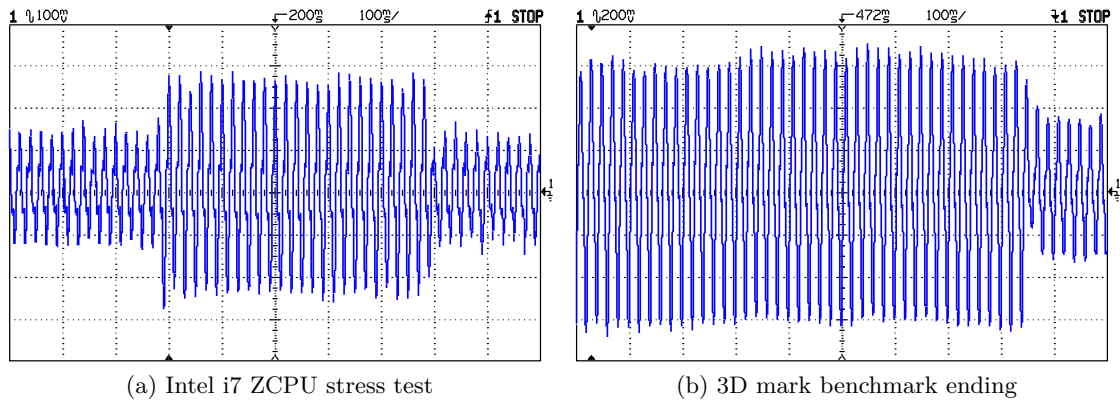


Figure 2.7: Gaming PC; the PSU ramps up the current within a single AC cycle to a multiple compared to idle usage.

sleep.

These measurements (Table 2.2) give us a preliminary insight regarding the achievable dynamics of load changing attacks performed by a botnet. As expected, capacitors in the power conversion units smear the hard edges of artificially produced power spikes. However, even in the worst case (60 ms per slope) an attacker can achieve modulation frequencies up to 8 Hz.

### 2.4.2 Categorization of Load-Altering Appliances

The second part of this Section looks at the question on the amount of controllable load by PC components and commonly found IoT devices and their usage.

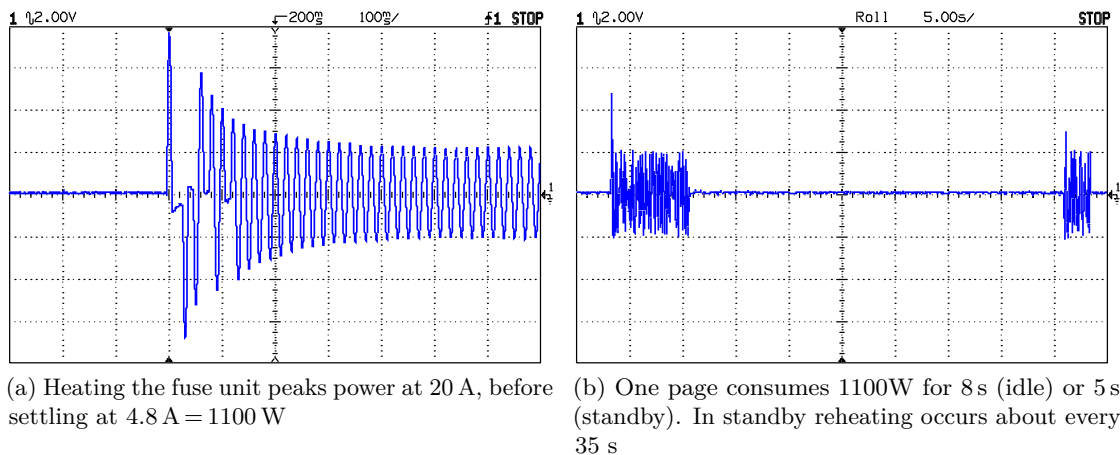


Figure 2.8: Power usage of a Brother HL2150 SOHO printer

Table 2.2: Modulated load by device

Device	Type	Pwr Control		Latency		$\Delta$ Load
		Inc.	Dec.	on	off	
CPU	Core2 Duo	●	○	20-60 ms	20-60 ms	35 W
	i3	●	○	20-60 ms	20-60 ms	55-73 W [75]
	i5	●	○	20-60 ms	20-60 ms	73-95 W [75]
	i7	●	○	20-60 ms	20-60 ms	77-95 W [75]
	i7-E	●	○	20-60 ms	20-60 ms	130-150 W [75]
GPU	Low-end	●	○	20-60 ms	20-60 ms	20-76 W [75]
	Mid-end	●	○	20-60 ms	20-60 ms	102-151 W [75]
	High-end	●	○	20-60 ms	20-60 ms	150-238 W [75]
	Top-end	●	○	20-60 ms	20-60 ms	201-297 W [75]
HDD		●	○	20-60 ms	20-60 ms	3-7 W [75]
Screen TFT	size dep.	●	●	1-5 s	5-10 s	60-100 W
Laser Printer	SOHO	●	○	1-3 s	5-10 s	800-1300 W
Smart Air Cond.		●	○	1-10 s		600-1000 W
Smart Thermostat	elec. Heating	●	○	1-10 s		1-15 kW
Smart Oven		●	○	1-10 s		2-3 kW
Smart Refrigerator		●	○	1-10 s		300-500 W
Smart Kettle		●	○	1-10 s		1000-1500 W

The last column of Table 2.2 comprises data from our own measurements, data sheets and PC power tutorials [75].

Since such an overview cannot depict the countless different models of hardware sold and installed around a world, its purpose is to estimate the impact of the attacks described above.

The  $\Delta$  *Load* column denotes the margin of controllable power consumption, e.g., the difference between idle state and full utilization. For example, desktop hard disks (typ. 5,400 RPM) have a lower base power consumption than server hard disks (typ. 7,200 - 10,000 RPM), but the difference between access and non-access is small.

In contrast to other appliances, screens can easily reduce power without much side effects by going to sleep. Major operating systems offer unprivileged API or command line calls to accomplish that. Hard disks can be sent to sleep as well (spin-down) but this typically needs administrator privileges. Furthermore regular background file system activity (book-keeping) will not make the effect lasting without putting the whole OS into sleep. Such a step withdraws the PC from the control of the botnet and is therefore not included.

As for printers, we did not consider office printers as they are usually shared by multiple users. Thus, print jobs are sequentialized and power consumption would not multiply with the number of infections, as it is spread over time.

Internet-of-Things devices are included in our list although they are still rare. The exceptions are smart thermostats [37, 223, 284] being increasingly sold in the U.S. [297], totalling 20 million devices since 2013 (U.S. has 126 Mio. Households [282]). Additionally, air conditioners [37, 125, 189]) and smart refrigerators [267] can be manipulated by changing the set-point temperature. Kitchen appliances such as smart ovens [266] and Wi-Fi-controlled water kettles [239] can also substantially draw (and alter) power.

## 2.5 Evaluation: Grid Effects

We investigated the effects of a botnet's load change on the continental synchronous grid. In particular, we seek to answer in which way and to what extent load has to be modulated by an adversary using the botnet. Furthermore, we studied whether the grid's state, i.e., total load or the mix of feeding power plants, influences the success of an attack. Such attacks against critical infrastructure can never be tested on a real system, specifically for a grid like the continental synchronous grid area providing power to more than 500 million people. Therefore, we developed a model in *Matlab/Simulink* that is based on the model of Ulbig et al. [299] and the ENTSO-E policies [303, 304]. In the remainder of this section, we describe in detail the model, the dependencies of grid parameters, and the success of each attack as presented in Section 2.3.

### Attack 1: Static Load Attack

The adversary suddenly increases a high amount of load; the raised demand leads to an imbalance of production and consumption, thus shifting the frequency from its nominal value to lower values. If the adversary's amount of load is high enough, the frequency decreases rapidly without the primary control being able to counteract in time. If the frequency goes down to 49 Hz, load is shed due to emergency protocol, i.e., numerous consumers become disconnected from the power grid.

For a simulation, we developed a model as depicted in Figure 2.9. The model contains the grid's response to a production-consumption imbalance with  $f_0 = 50 \text{ Hz}$  (nominal frequency), start time constant  $T_S$  and the network power  $S_N$ . Further, it contains two feedback loops: The first considers the self-regulation of load in case of frequency changes; the load typically changes 1%/Hz. The other feedback represents primary control, containing a saturation when reaching 200 mHz (at this point the full primary reserve is activated), a proportional element with a gain of 15,000 MW/s (full primary reserve of 3,000 MW should be activated at 200 mHz), a PT1-element representing turbine characteristics with  $T_N = 2 \text{ s}$  (fast gas turbines) and a maximum slew rate of 500 MW/s as specified by ENTSO-E policies. With  $T_S = 10$  and  $S_N = 150 \text{ GW}$ , the system's response to the reference incident (RI) of 3,000 MW corresponds with the design hypothesis of the



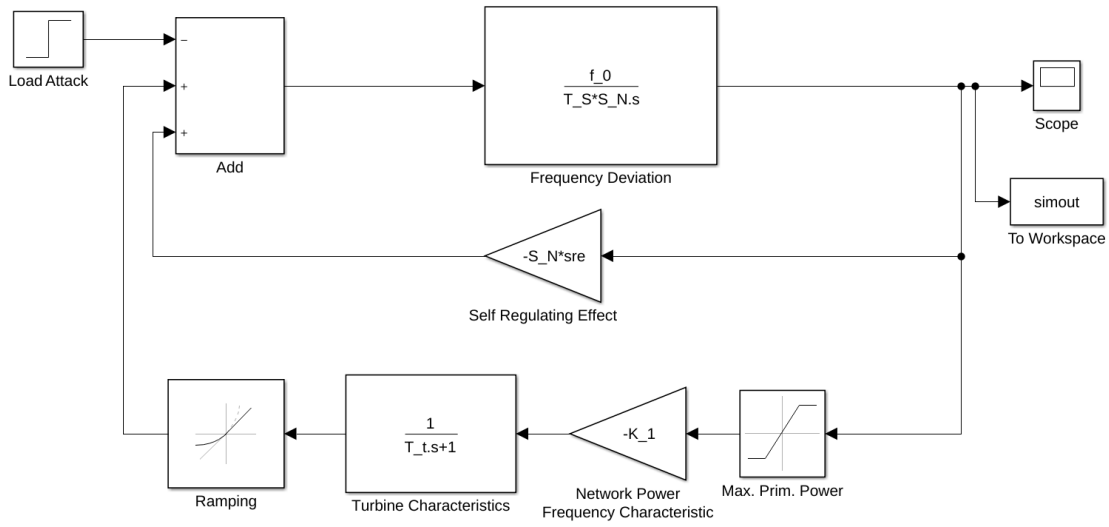


Figure 2.9: Model for static load attack (primary control)

policies [303] and emphasizes our model’s accordance with the continental synchronous grid. Secondary control is not included into this model as it would not be activated at such an early phase of imbalance.

In a first step, we investigated the impact of the power grid’s network power  $S_N$  on the amount of load that has to be modified by the adversary.  $S_N$  represents the amount of currently produced power and differs in the course of days, weeks and seasons. Generally, it is lower during night, summer and on national holidays, as consumers request less power than during daytime, winter and on work days. Values for network power  $P_N$  are taken from ENTSO-E statistics of the year 2016 [117]: The highest load was 583,711 MW on January 19th 2016, 5-6 a.m., the lowest load of 263,591 MW whereas on May 29th 2016, 6-7 p.m, occurred the median load of 2016 was 409,823 MW.

We measure the static load attack in multiples of a ENTSO-E’s *reference incident* of

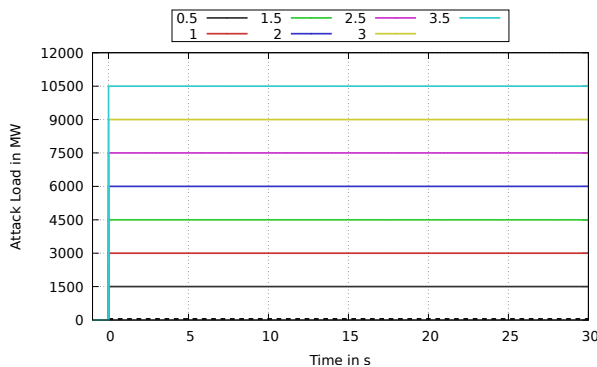


Figure 2.10: Jump function of the additional load for the static load attack in multiples of ENTSO-E’s reference incident (RI=3,000 MW)

## 2. BREAKING THE INDEPENDENCE PRIMITIVE

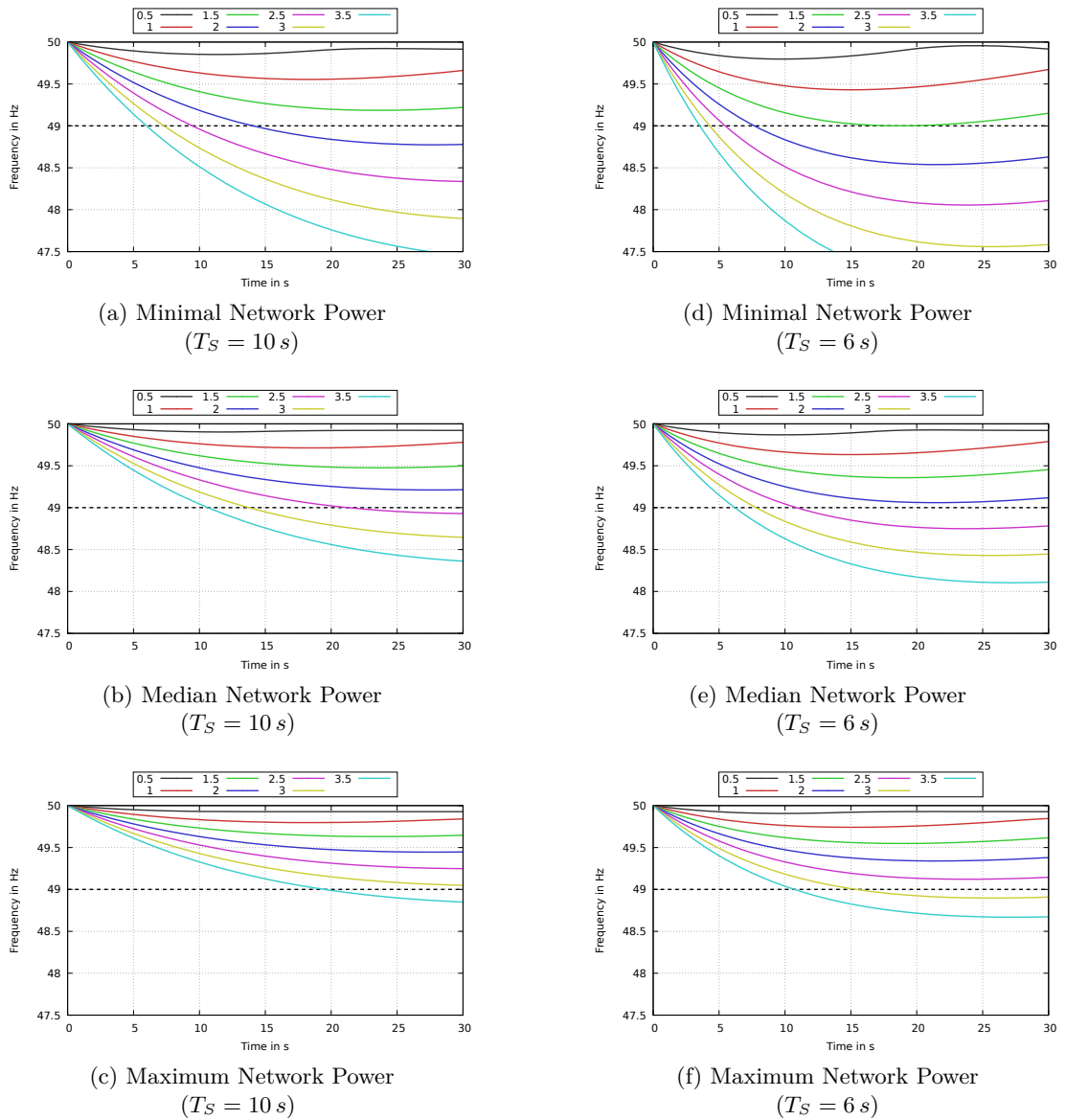


Figure 2.11: Impact of static load attack on frequency in a grid with high rotational inertia (a-c), i.e., predominantly fed by conventional power plants, and low rotational inertia (d-f), i.e., fed by a high share of renewables, at different levels of total network power. Static load attacks are in multiples of the ENTSO-E reference incident (3,000 MW).

3,000 MW. The impact of these attacks on grid frequency with a start time constant of  $T_S = 10\text{ s}$  is shown in Figures 2.11a-2.11c. Reaching the threshold of 49 Hz causes load shedding, and, thus, a successful attack. At minimal network power twice the reference incident, i.e., 6,000 MW is enough, whereas median network power requires 2.5

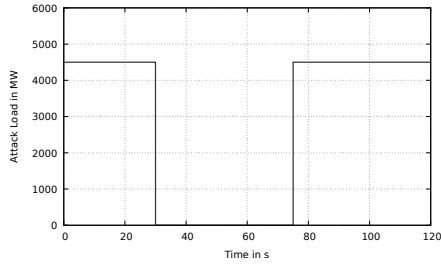


Figure 2.12: Dynamic load attack  
(1.5 reference incidents)

times the reference incident, i.e., 7,500 MW, and maximum network power 3.5 times, i.e., 10,500 MW. In conclusion, it is easier to reach the threshold for load shedding at times of an overall low power level in the network, i.e., at night, during summer and on national holidays.

Finally, the start time constant  $T_S$  is dependent on the type of power plants supplying the grid and is historically getting lower due to the increased use of renewables (wind turbines, PV)<sup>2</sup>.  $T_S$  might get as low as 6 s [299]. Figures 2.11d-2.11f highlight the consequences: more renewables make the frequency shifting faster, and reaching the threshold for load shedding becomes easier. Low start time constants are typically encountered during times of low power generation, e.g., on national holidays with lots of wind, as renewables sources are preferred for supply in Europe.

### Attack 2: Dynamic Load Attack

Dynamic attacks promise to be more successful than static ones, i.e., reach higher frequency shifts while modulating the same amount of load. In our case, the adversary drives all load to full power, waits until primary control is initiated and reaches its maximum; then, the adversary withdraws all power consumption. Since the primary

<sup>2</sup>Photovoltaics and many wind-turbines are connected to the grid by solid-state inverters. In consequence, they can not stabilize the grid's frequency by means of rotational inertia.

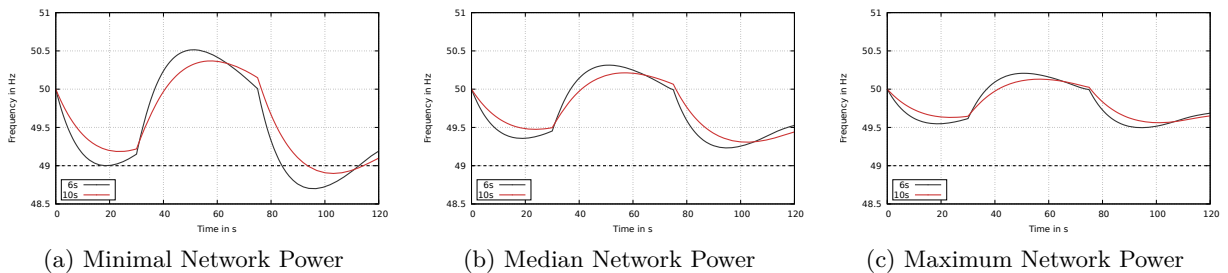


Figure 2.13: Dynamic load attack at different levels of total network power and rotational inertia

control's full activation takes 30 seconds, the attack load is modulated as depicted in Figure 2.12 (Our simulation relies on the model as shown in Figure 2.9).

The results of an attack via modulating 1.5 times the reference incident are shown in Figure 2.13: the absolute frequency shift at the second swing after 80 seconds is typically higher than at the first one; in addition, the frequency is becoming larger than the nominal value of 50 Hz for a period of roughly 30 seconds, i.e., frequency overshoots despite an adversary that is solely able to modulate additional load in a grid<sup>3</sup>. Again, the less network power, e.g., during summer and nights, and the smaller the start time constant (more renewables), the easier it is to reach the threshold of 49 Hz for load shedding; the higher the attack load, the higher the frequency shift.

### Attack 3: Inter-Zone Attacks

This attack relies on a synchronous grid containing multiple zones which are interconnected by transmission lines. In a first step, the adversary increases the load in one zone. Secondary control is eventually activated, and compensates for this additional consumption. As soon as this happens, the adversary reduces the load, while increasing it in the other zone, waiting for secondary control to compensate again. Finally, this leads to high amounts of transmission on the tie lines, which might eventually trip them.

For the simulation, we extended our model by another feedback loop representing secondary control, see Figure 2.16. Secondary control calculates the Area Control Error (ACE) as described in Section 2.2.2.  $P_{measured}$ , i.e., the transit to other areas, is fed into the model via input 2,  $P_{planned}$  is assumed to be zero. The ACE is then forwarded to a delay (which might be up to 5 seconds [304]), and eventually to a PID controller representing the secondary controller with anti-wind-up functionality ( $C_p = 0.17, T_N = 120$ ), a saturation when reaching the maximum amount of secondary control, again a PT1-element representing turbine characteristics  $t_N = 2$  s (fast gas turbine), and a ramping as power plants cannot increase/decrease with arbitrary dynamics. Then, we took two

<sup>3</sup>In the past, wind turbines were disconnected from the grid at a frequency of 50.5 Hz.

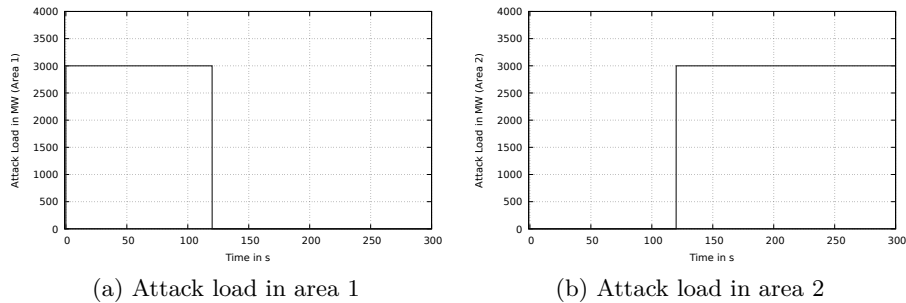
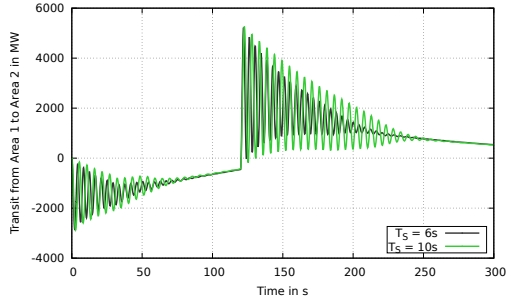
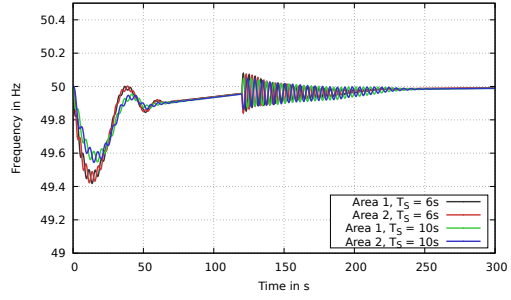


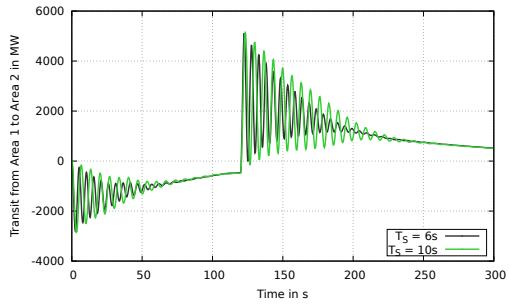
Figure 2.14: Inter-zone attack: Load added in one zone is removed from the other (reference incident)



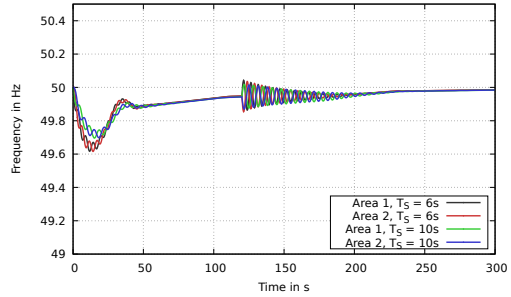
(a) Minimal Network Power



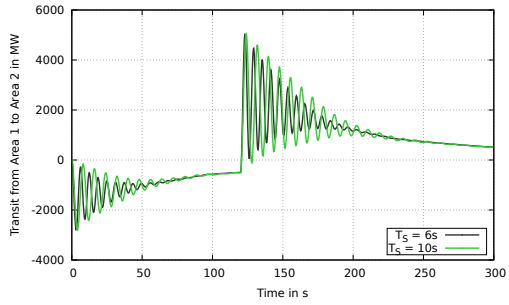
(d) Minimal Network Power



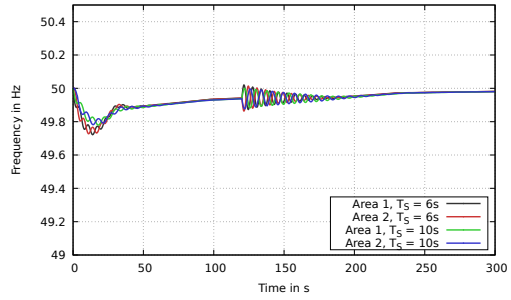
(b) Median Network Power



(e) Median Network Power



(c) Maximum Network Power



(f) Maximum Network Power

Figure 2.15: Transit power and frequency deviation in inter-zone attack at different levels of total network power and rotational inertia: a-c) tie line; d-f) frequency

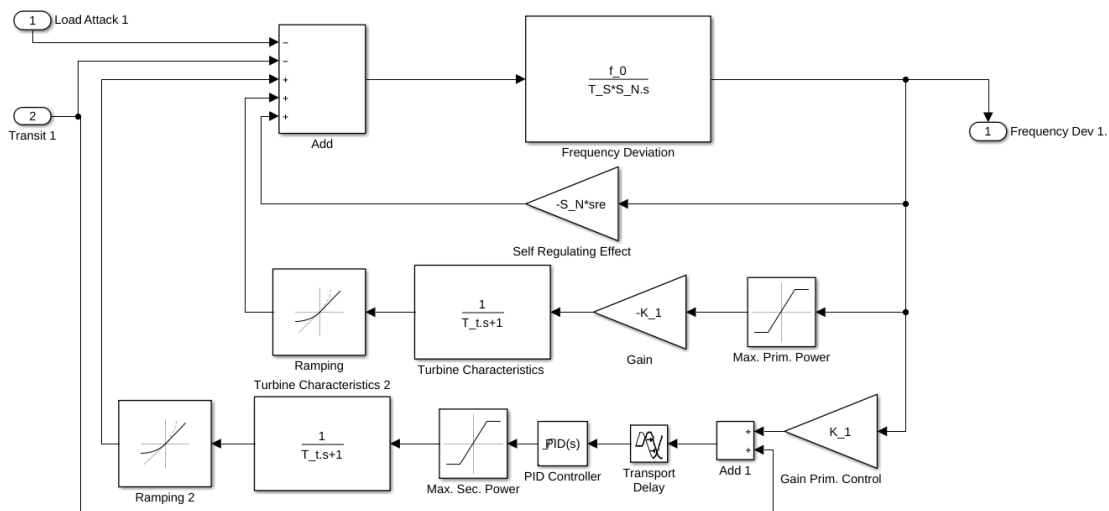


Figure 2.16: Model for control area including primary and secondary control

such areas and connected them to a 2-area model by subtracting one area's frequency from the other and feeding the result into the transfer function  $\frac{2\pi P_{Tie}}{s}$  to finally gain the power in transmission. Their frequencies are feed into the tie line's transfer function and further to both areas but with opposite sign.

Figure 2.14 shows the load that is modulated by the adversary in areas 1 and 2; Figure 2.15 shows the result in dependence of total network power load and rotational inertia. Figures 2.15a-2.15c show that the maximum amount of power in transit over the tie line is to a great extent independent from these parameters. They rather have an impact on the frequency deviation as shown in Figures 2.15d-2.15f, but inter-zone attacks aim to trip power lines. Thus, the adversary has to aim for a maximal power spike between area 1 and area 2 and fast changes, as line-tripping is done based on the total amount of load in transit as well as its time derivative [299].

## 2.6 Discussion

In Section 2.3.2 we outlined the botnet and in Sections 2.4 and 2.5 we measured and simulated the components and attacks. In Section 2.6 we put the pieces together and sketch different distributions of infections to estimate the botnet size needed for an successful attack.

Based on the hardware listed in Table 2.2 we created four prototypical desktop computer configurations as presented in Table 2.3 which reads as follows: We assumed that home computers have a CPU-class distribution of 5% Core2, 30% i3, 40% i5, 20% i7, and 5% i7E or equivalents, totaling 100%. GPU values read accordingly, again summing up to 100%. For servers we assumed a higher probability for real multi-processor systems, effectively summing up to more than 100%. Likewise, gaming PCs (and to some degree

Table 2.3: Prototypical computer hardware configurations with expected modulatable load

<i>Components</i>	$\Delta$ Watt	<i>Office</i>	<i>Home</i>	<i>Game</i>	<i>Server</i>
Core2	35	5%	5%	0%	0%
i3	64	40%	30%	5%	30%
i5	84	30%	40%	30%	80%
i7	86	15%	20%	40%	90%
i7E	140	10%	5%	25%	50%
GPU-Low	49	50%	50%	5%	0%
GPU-Mid	126.5	30%	30%	40%	0%
GPU-High	194	15%	15%	40%	0%
GPU-Top	249	5%	5%	15%	0%
TFT	80	125%	110%	150%	0%
Laser Printer	1,100	5%	30%	30%	0%
Expected $\Delta$ Load		<b>338.45</b>	<b>600.75</b>	<b>715.8</b>	<b>233.8</b>

Table 2.4: IoT scenarios

IoT	$\Delta$ Watt	<i>Mix 1</i>	<i>Mix 2</i>
AC	800	0%	4%
Thermostat	8,000	4%	8%
Oven	2,500	0%	1%
Refrigerator	400	0%	1%
Kettle	1,250	0%	1%
Expected $\Delta$ Load		<b>320</b>	<b>692.75</b>

Table 2.5: Infections needed

	<i>Distribution 1</i>	<i>Distribution 2</i>	<i>Distribution 3</i>
Office PC	40%	40%	30%
Home PC	30%	30%	40%
Gameing PC	15%	15%	20%
Server	15%	15%	10%
+ IoT-Mix (Table 2.4)	-	Mix 1	Mix 2
Avg. $\Delta$ Load p. Infection	458.045 W	778.045 W	1,201.125 W
Infections 3000 MW (1 RI)	6,549,575	3,855,819	<b>2,497,659</b>
Infections 4500 MW (1.5 RI)	<b>9,824,363</b>	5,783,728	3,746,488

others as well) have a higher probability of being connected to more than one screen [309]. The lowest row lists the expected controllable load per infected PC.

For IoT devices (Table 2.4) we created two different scenarios: A conservative one with just smart thermostats and another one with additional devices. The former reflects the fact that est. 20 Million devices [297] have been sold in the last 4 years in the U.S. to their 126 Mio. households [282]. We reduced the factor by 1/3 to account for to the distribution of electric heating systems [101] in the U.S.

Table 2.5 combines the different computer types from Table 2.3 as well as the IoT scenarios into three possible infection distributions, whereas the first — most conservative estimation — excludes IoT devices completely. The second distribution corresponds with the first, with the addition of thermostats and the third adds all classes of IoT devices.

The following row computes the expected controllable load on average per installation of a botnet client, based on the distribution of infected computers. The last two rows display the botnet size necessary for 1 and 1.5 reference incidents (3,000 MW or 4,500 MW respectively).

Depending on the mix of infected devices, a successful attack can be achieved with 2.5 to 9.8 million devices — depending on other conditions described in Section 2.5 are met, such as day of time and mix of energy sources. For attacks 1 and 2 the infections can be located anywhere within the synchronous grid.

While this can only be considered a rough estimate, it is well within reach of real-world botnets. More accurate estimates are difficult [46, 251], but go up to 50 Mio. infected computers at the peaks times of certain botnets [289]. These figures cover infections globally, but Europe’s estimated 17% share of Internet users in 2017 [211] and high technological level let these numbers appear feasible. Furthermore, we anticipate an increase of connected computers and Internet-enabled devices in the next years.

### 2.6.1 Limitations

The used simulation models are based on Ulbig et al. [299] and have to estimate some properties of the grid such as the mix of generator characteristics. A more precise simulation is possible with data from TSOs or ENTSO-E which include the exact mix of connected power plants and their scheduled (or actual) availability.

### 2.6.2 Future work

The simulation model for attack 2 uses resonance mainly caused by activation delay and generator characteristics of the primary control. However, additional grid-inherent resonances are known for the ENTSO-E synchronous area [115], [306, p.77], [185, p.3]. An attacker could piggyback on top of them and try to amplify them to gain more leverage. Grid-inherent resonant frequencies could also amplify the effects of attack 3.



We did not look at cascading effects which were almost always involved in large scale power outages [116,306,307]. These simulations are only possible with grid wide topology data including all tie lines characteristics.

In this paper we only targeted severe power disruptions of the grid e.g., by load shedding. However, an attacker could also just aim for economic damage invisible to the end-customer. Immediate costs arise by the additional deployment of reserves and increased unplanned international transfers. Long-term costs are associated with the permanent allocation of reserves as preparation for such attacks.

## 2.7 Related Work

Irregular behavior in power grids happens from time to time, mostly due to unexpected incidents and not as a consequence of malicious behavior. ENTSO-E investigates and publishes such incidents to advance the knowledge for proper incident response. Thereby, ENTSO-E reported on inter-area oscillations [115], the impact of solar eclipses on power production [257], a blackout in larger parts of central Europe caused by a cascade of tripping lines [306], and a similar one in Turkey [116]. The first action against a power grid known to have been malicious happened in the Ukraine in 2015. The adversaries used malware delivered via e-mails, stole credentials and finally got access to the power providers' SCADA systems [70]. The adversaries used attack vectors well-known in traditional IT, whereas our attacks strike the power grid – a cyber-physical system – in its physical part.

Numerous works considered false data injection attacks, i.e., an adversary compromising meters and sending wrong data to the provider, and their detection [196,197,202,334,337]. Mishra et al. [212] investigated rate alteration attacks, i.e., fabrication of price messages, in smart grids. Mohsenian et al. [214] introduced the notion of Internet-based load attacks on smart grids, for example by manipulating computational load, exploiting capabilities of demand-side management or (apparently) manipulating spot-market electricity prices, e.g., so that programmable smart meters start charging electric cars all at once. Furthermore, the remote kill switch found in some smart-meters to disconnect subscribers from the grid has been suggested for similar destabilizing attacks on power grids as in our paper [43,81]. However, as of 2017, meters with demand-side management are rolled out only to a limited extent. Smart meters that are rolled out at large-scale under various national and EU programs [167] often are metering-only and do not include a power control switch as they are more expensive and some nations completely opt out from such functionality [158].

Amini et al. [41,42] claim that dynamic load attacks are more successful with respect to their impact on grid frequency. Their model requires, however, huge power-shocks, effectively doubling the power consumption. We considered attacks with a load up to 3.5 times the continental synchronous grid's reference incident, i.e., 10,500 MW in total. This is lower than 4 % of the grid's total load, even at times of lower network power, thus making our attacks more practicable.

Xu et al. [331] aimed to increase loads in IaaS, PaaS and SaaS clouds to trip data centers' circuit breakers. The load increase is caused by the adversary renting cloud services or by using external web services to trigger computationally expensive operations. The authors sought to unplug a cloud provider's data center, but did not negatively impact the power grid itself, whereas our attacks aim to directly shut down the power grid or at least parts of it. Beyond that, our attack load might consist of any kind of controllable load and is not limited to cloud-based loads.

## 2.8 Conclusion

Power grids are among the largest human-made control structures and pre-date large communication networks like the Internet by decades. Their successful, i.e., synchronized, operation requires constant balance of power supply and demand; therefore, power providers maintain elaborated models to forecast demand in dependence of parameters like time of the day, season and weather conditions. However, these models rely on the assumption that fluctuations caused by single consumers are averaged out on a macro scale, i.e., for each consumer turning a light bulb off there is another one turning the light on. In our scenario, an adversary builds (or rents) a botnet of zombie computers and modulates their power consumption, e.g., by utilizing CPU, GPU, screen brightness, and laser printers in a coordinated way. Outperforming the grid's countervailing mechanisms in time, the grid is pushed into an unstable state triggering automated load shedding or tie line tripping due to under-frequency.

We developed three different attacks against the power grid and analyzed their feasibility. Therefore, we first investigated the dynamics and increase of different loads, in particular regarding PCs and IoT devices. We found CPUs, GPUs, and screens with a controllable load increase of 100 W and more; printers and IoT devices with even 1,000 W and more. In a second step, we simulated the impact of load attacks on grid stability, given that testing our attacks on a real power grid is infeasible for a variety of reasons.

Under favorable conditions, i.e., low total network power and a high share of inverter-connected renewables feeding power into the grid, 4,500 MW of additional load is sufficient to destabilize the system and trigger load shedding. In the European continental synchronous grid area, these conditions are typically prevalent at night or on public holidays with high wind power supply. According to our computations, an adversary requires a botnet of 2.5 to 9.8 million bots (Table 2.5). While this is not feasible in all cases, it might be worthwhile for entire nation attacks.

While terminology and details differ between synchronous grids worldwide, we want to stress that the general principles and conclusions are applicable to all AC power grids and our attacks work in every of these grids, though minor adaptations likely be necessary.

## Part II

# Mobile Phone Radio Networks



# Background and Mobile Phone Networks Security Problems

Over the last decades, numerous security and privacy issues in all three active mobile network generations have been revealed that threaten users as well as network providers. This chapter aims to identify, categorize and describe known vulnerabilities in the 3GPP implementations of 2G, 3G, and 4G networks. For a systematical approach, we categorizes known attacks by their aim, the attacker capabilities, the range, underlying causes, and root causes. By doing so, we identify ten causes and four root causes for attacks.

Additionally, we give a brief overview of the technical principles of a mobiles phone network and describe fake base station attacks in more detail.

This chapter is based on excerpts from our publications [90,91], unpublished work [263], and original content.

## 3.1 Overview and Introduction

Since the introduction of mobile phone communications, it transitioned from a luxury service into a commodity service and than into a crucial part or infrastructure for many economic sectors: logistics, payment services, vending machines, security services and many more. Additionally, commercial networks increasingly surpass dedicated administrative and governmental networks in coverage and data rates and thus carry an increasing amount of sensitive data. Over the past decades, mobile communication has become an integral part of our daily life. GSMA estimates over 5 billion subscribers in 2017 [141]. As a consequence, the reliability and security of mobile networks have become a substantial asset of our daily lives.

Over the last years, a large body of literature has revealed numerous security and privacy issues in mobile networks. There is a broad set of attacks [27, 48, 78, 181, 213, 275]

that affect the users' privacy and data secrecy, the mobile network operators' revenue, and the availability of the infrastructure. Besides the academic community, the non-academic community also substantially contributed to the comprehension of mobile network security. Unfortunately, attacks and countermeasures were mostly considered in an isolated manner and the research efforts have not been systematized or categorized into a big picture. However, these insights are necessary to develop generic countermeasures instead of separate fixes or mitigations. For example, messages being exchanged before the authentication and key agreement is the cause of multiple attacks [121, 131, 275]. Considering the attacks separately, one might not assume that this is a broader problem present in all three mobile generations.

This chapter will not only try to give a comprehensive overview of security issues in current mobile phone networks, it will also provide a condensed explanation on the foundations of modern mobile communication networks and explain some attacks (such as fake base stations) in more detail, as needed for the following chapter 4 and chapter 5. The scope of this chapter is on the technical side of mobile networks (Figure 3.1) including the air interface and the interfaces to other telecommunication networks such as the Internet, or other telecoms via SS7. It fills the gap between mobile applications [302] and mobile operating systems such as Android [35] on one side and generic Internet security surveys [300] and telephony fraud attacks [143, 265, 296] on the other side.

## 3.2 Mobile Network Background

In the following, we describe the technical background of mobile networks, including an overview of the currently active generations and a generic overview of the network architecture.

### 3.2.1 Generations

Over the years, the requirements for mobile networks have shifted from rather single-purpose networks (voice service) to multi-purpose networks (voice, video, data, texting). In the following, we introduce the currently active network generations.

- **GSM (2G, Global System for Mobile Communications)** is the first digital mobile communication system and was designed for voice transmissions. It uses circuit-switched scheduling in which fixed slots are allocated for transmissions over the air and on network components along the transmission path. This mimics the circuit-switched architecture of public telephone networks back then. The later-introduced General Packet Radio Service (GPRS) and Enhanced Data Rates for GSM Evolution (EDGE) is a packet-switched extension retrofitted into the fixed slot scheduling of GSM.
- **UMTS (3G, Universal Mobile Telecommunications System)**: In order to meet the increasing demand for data transmissions, the next generation was

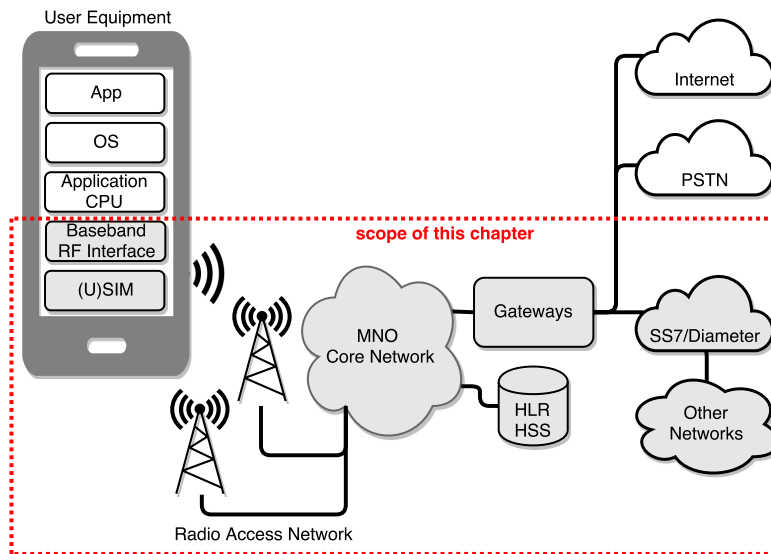


Figure 3.1: Generic mobile network architecture and the herein focused areas.

optimized for data transmission on the radio layer. Additionally, UMTS added new security features such as mutual authentication and new encryption algorithms. Although the network is packet-switched in its core, voice and SMS transmissions are still offered as distinct network services.

- **LTE (4G, Long Term Evolution)** uses a completely redesigned radio layer and a strict IP-based packet-switched architecture with guaranteed Quality of Service (QoS) classes. In contrast to its predecessors, voice and SMS transmissions are no longer network services, but offered as IP-based services (SIP, VoIP) on top of a general-purpose IP data network. However, fallback options exist for phones or operators which do not support Voice over LTE (VoLTE) [26, Sec. 8.2].

The irony of all those enhancements and improvements is that the old generations never went away. While a few operators began to phase out older standards, new handsets still support all standards for maximum versatility. Thus, security problems accumulated and sometimes new were introduced through interoperability and compatibility measures.

### 3.2.2 Network Architecture

Names and abbreviations for equivalent or highly-similar network components and concepts vary between the different network generations. Overall, we try to stay agnostic to the generations and access technologies wherever possible and will typically use the oldest abbreviation as generic term for a concept. If we need to specially differentiate between different terms, we denote them with a <sup>2G</sup> for GSM, <sup>3G</sup> for UMTS, and <sup>4G</sup> for LTE. For example: a Location Area (LA) and Location Area Updates (LAU) describe

the generic concept that later technologies named Tracking Area (TA<sup>4G</sup>) and Tracking Area Update (TAU<sup>4G</sup>).

Figure 3.1 shows a generic network architecture including the scope of this chapter. In general, the architecture consists of the components and concepts below.

### 3.2.3 User Equipment

The Mobile Station (MS)<sup>2G</sup> or User Equipment (UE)<sup>3G,4G</sup> (e.g., smartphone) is the device utilized to communicate with the network and consume its services. It consists of different components, such as the application processor that runs the mobile operating system, the graphical user interface, and all its locally installed applications. An additional baseband processor implements the mobile protocol stacks for multiple network generations and thereby establishes the communication with the network. The SIM<sup>2G</sup>/USIM<sup>3G,4G</sup> (Universal Subscriber Identity Module) directly identifies a customer and stores the authentication information as a pair of the permanent identity (IMSI, International Mobile Subscriber Identity) and the secret long-term symmetric key used for encryption and authentication. From outside, a user is identifiable and thus callable via a public phone number (called Mobile Station Integrated Services Digital Network Number (MSISDN)). It is the subscribers home provider that creates the mapping between the MSISDN and the IMSI. The first few digits of an IMSI contain the country code and the network operator code to help finding the the home operator when internationally roaming. Besides permanent identities – for privacy reasons, as many of these identifiers are transmitted in clear – temporary identities are dynamically allocated to the UE, such as a Temporary Mobile Subscriber Identity (TMSI) or Globally Unique Temporary ID (GUTI)<sup>4G</sup> that is used for paging and core network communication.

### 3.2.4 Radio Access Network

The Radio Access Network (RAN) transmits data between the UE and the core network that provides services to the user. Therefore, the mobile phone establishes a radio connection to the base station (BTS<sup>2G</sup>, nodeB<sup>3G</sup>, eNodeB<sup>4G</sup>) that acts as a network access point. For mobility management (Section 3.2.9), base stations are organized into cells which are in turn grouped for circuit-switched services into Location Areas (LAs)<sup>2G,3G</sup>, and for packet-switched services into Routing Areas (RAs)<sup>2G,3G</sup> and Tracking Areas (TAs)<sup>4G</sup>.

### 3.2.5 Core Network

The core network's task is to manage the connection mobility and to deliver the services, e.g., phone calls and Internet connection. For this mobility management, several core network elements are utilized. A central database, the Home Location Register (HLR)<sup>2G</sup> or Home Subscriber Server (HSS)<sup>3G,4G</sup>, stores the authentication, mapping, and other information about the users. Its security functionality is often referred as Authentication Center (AuC). Core network elements manage the mobility, connection, and security establishment. Signalling System #7 (SS7) is used within GSM and UMTS networks



for signaling purposes such as mobility management and call setup as well as externally for roaming. SS7 was developed in the mid-1970s for landline networks and was later extended for mobile telephony networks. Unfortunately, the protocol only provides limited security mechanisms. Today, SS7 is mostly used as an SS7-over-IP adaptation. LTE introduced new IP-based protocols for the core-network infrastructure, e.g., the SIP-based IP Multimedia Subsystem (IMS) handles voice, video, and text messages.

### 3.2.6 Inter Network

Many services require a connection to other communication networks such as the Public Switched Telephone Network (PSTN) or the Internet leading to the introduction of subsystems and gateways.

Mobile networks are connected to each other via SS7 or its successor, the Diameter protocol for global inter-network operator roaming, text messages, and call forwarding. The SS7 standard for Internet-based protocols is called SIGTRAN. Diameter inherited most of the SS7 semantics, but offers improved authentication and confidentiality through the use of IPsec and Transport Layer Security (TLS).

### 3.2.7 Radio Channel

The radio layer shares some common high-level design choices between GSM, UMTS, and LTE, whereas other characteristics like frequencies, modulation, or access technologies are highly individual. All generations incorporate three main types of logical channels into the physical radio channel: (i) Broadcast control channels carry information about the base station, its neighbors, and the network configuration. (ii) Paging channels are used to call out for specific UEs when the network wants to transmit data to them. (iii) Dedicated channels are used for traffic to and from each single device. These are the only channels that can be encrypted and integrity protected, if initiated by the network.

### 3.2.8 Pre-Authentication Traffic and Security Establishment

Unless initiated by the network, the traffic is unencrypted, not integrity protected and, thus, not authenticated. This means that only dedicated traffic to and from a specific device is secured. Thus paging, other broadcasts, most of the radio resource allocations, and low-level signaling traffic are always unprotected. All traffic that happens before the setup of an authenticated session is defined as *pre-authentication traffic*.

The authenticated session is established via an Authentication and Key Agreement (AKA) protocol which is a challenge-response protocol, that authenticates the partner(s) and derives a session key for the communication. While GSM only establishes user authentication, newer generations (UMTS, LTE) establish mutual authentication. The session keys are derived from a common long-term shared secret stored in the (U)SIM. The particular AKA used depend on the deployed SIM, the operator's AuC, and the access technology. In GSM, the example algorithm COMP128 became the de-facto

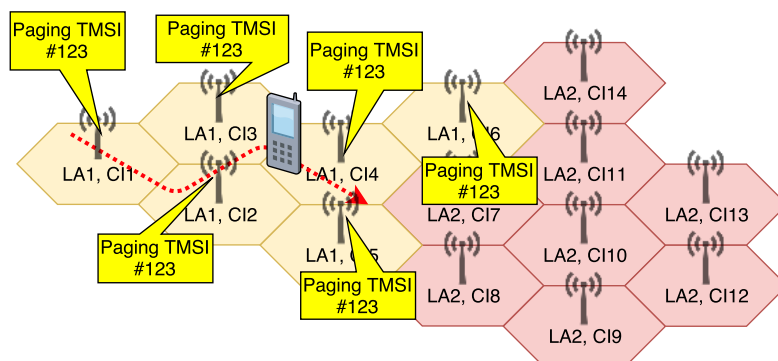


Figure 3.2: All cells of a Location Area are paging the same phones

standard [320, Ch. 16], albeit operators could issue SIMs with a custom algorithm. In later generations, key derivations are split between the UE and the Universal Subscriber Identity Module (USIM) where the publicly reviewed Milenage (and TUAK) algorithms are used. Operators are still able to customize AuC's and USIM's algorithms.

### 3.2.9 Mobility Management and Paging

When no active data transmission or phone call is ongoing, the phone goes into the idle state. In this state, the network only knows the coarse Location Area (LA) where the subscriber is located but not the exact cell. The phone listens to the paging channel as an incoming phone call, message, or data triggers a paging message of the subscriber in the whole Location Area (i.e., pages are the same on all cells of a Location Area). Upon receiving a paging message, the phone contacts the network and requests a dedicated (logical) channel for further communication. Only then, the network learns the cell-level location of the phone.

However, in the case the phone moves to another Location Area (circuit switched), it has to inform the network using a *Location Update Request*. Additionally, the phone sends *periodic* location updates at a low interval (typically every 24h) to reassure the network of its continued presence. Analogous semantics exist for Routing Areas and Tracking Areas<sup>4G</sup> in the packet-switched context. On LTE, the network can additionally provide an individual set of Tracking Areas for each UE, so that a group of subscribers – e.g., on a train – do not perform a Tracking Area Update all at once.

Additionally, each cell broadcasts a list of neighbor cells (e.g., their frequencies) to help the phone find these cells faster. The phone can save power by not having to scan all possible channels for new (stronger) cells, but only have to monitor the channels advertised by the network.

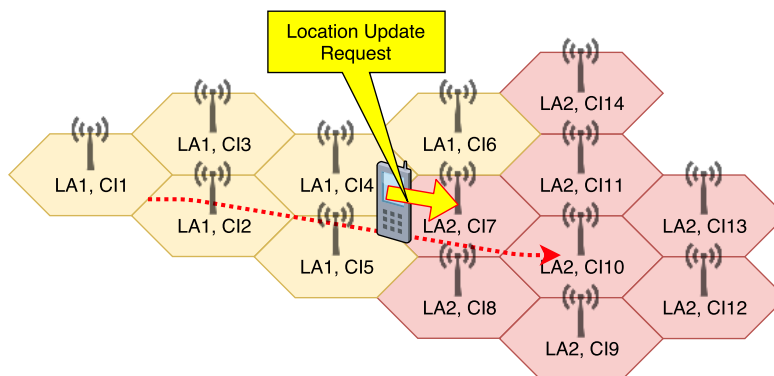


Figure 3.3: A UE can listen to any paging channel in the Location Area, but has to inform the HSS/HLR about entrance into a new Location Area.

### 3.2.10 Power Management and Conservation of Radio Resources

Above mentioned mobility management also serves the goal of power conservation on the battery powered UE. An UE will select the base station with the strongest signal and can then reduce its transmission power. Furthermore, reducing power also limits the range of transmissions. Thus, the operator can reuse the frequency with less interference at another base station.

Another method to reduce power consumption is limiting the paging channel to a certain time slot (GSM) or low-bitrate channel (LTE). Thus, the phone does not have to decode all radio traffic but only very limited set.

Similarly, the broadcast neighbor cell list allows the phone, to limit the monitored channels. Only if the connection to all of these neighbors is lost, the phone must perform a full scan, i.e., sweep to all of the frequency band to find suitable cells.

### 3.2.11 RAN Sessions and Data Tunnels

As most data services need stable addresses, tunnels are used between the UE and an IP endpoint. These tunnels hide the mobile network's mobility management and also allow to offer multiple connections to different IP networks, such as Internet access, VoLTE/VoIP, or private networks. Tunnels terminate at the packet gateways (Packet Data Network Gateway, P-GW<sup>4G</sup>). If necessary, Network Address Translation (NAT) middleboxes separate the mobile network from the Internet by translating the private IP address and port to a public IP address and port. Tunnels aim at guaranteeing certain QoS parameters, such as latency or bandwidth.

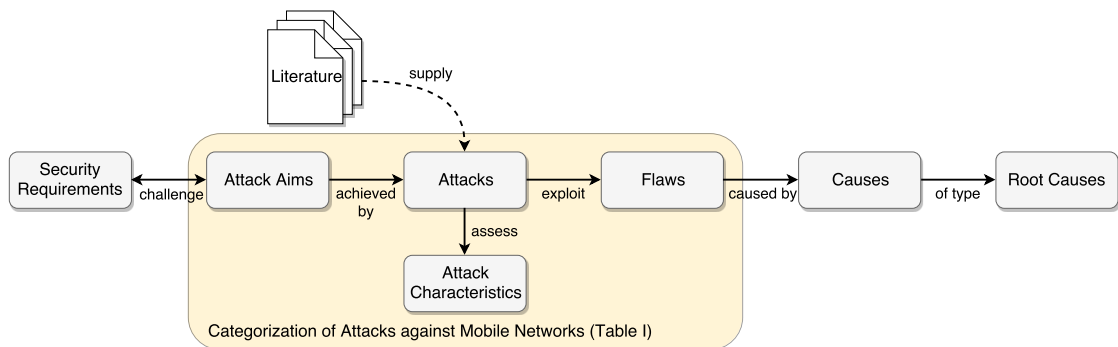


Figure 3.4: Systematization methodology for Vulnerabilities

### 3.3 Systematization of Vulnerabilities

In this section, we introduce our systematization methodology that we apply to categorize attacks.

#### 3.3.1 Methodology

We structure the systematization process according to the flow depicted in Figure 3.4, beginning with the selection of particular security requirements, continuing with the assessment of recent attacks. This process allows us to incorporate multiple aspects of previous work resulting in a high-level perspective on essential root causes of security issues.

**Security requirements** define concrete features to protect mobile networks and their users. Such requirements are challenged by **attack aims**, i. e., the major interests of an attacker (Table 3.1). Accordingly, **attacks** are instantiations of these attack aims and exploit existing vulnerabilities in the definitions and implementations of systems or system components. In order to assess the impact of attacks on the mobile network, we use a set of **attack characteristics** that give a precise definition of an adversary’s technical and organizational capabilities, e. g., the preliminaries for an attack. However, multiple flaws are often manifestations of a broader problem that we define as a **cause**. Such causes facilitate the differentiation of attacks into distinct classes, thus allowing to derive open research questions and challenges in relation to the current state of the art. All causes are grouped into four fundamental **root causes**, which form the logical structure of our systematization.

Below, we give a short example to illustrate the application of this method.

**Example** *Radio Measurement Reports* are used for the maintenance of radio access networks and can be requested by a base station without authentication. This flaw can be exploited via the *Radio Measurement Report Request Attack* that allows an attacker with *active radio capabilities* to pinpoint a victim [275]. Hence, this attacks

the user's privacy (*attack aim*) as it breaks the location confidentiality *requirement*. The attack can be executed in the non-authenticated state of UEs, allowing an active attacker to fake measurement report requests. Therefore, the *flaw* is that requests for Radio Measurement Reports are part of the unsecured pre-authentication traffic (*cause*). As this is a legitimate request according to the specification, the *root cause* lies in the specification.

### Coverage of Literature

We focus on academic research, i. e., scientifically peer-reviewed publications, as a foundation for the assessment. In addition, we use non-academic research (including publications, presentations, and demonstrations at hacker and industry venues) and white papers from industry (e.g., GSMA or 3GPP). Even though these publications are not peer-reviewed, they complement the academic body of the systematization with a comprehensive picture of mobile network security. In particular, the hacker community has contributed a lot to the understanding of mobile protocols [319] and has provided tools that academic researchers have built upon.

We require that the literature must present security- or privacy-related attacks and need to be unique to mobile networks and not focus solely on applications. We specifically exclude mobile operating systems security and common challenges of the public phone network. Preferentially, the literature should have a high impact, e. g., it affects many users, is operable from a large distance, or produces considerable damage.

### Structure

We base the structure of our systematization (Sections 3.5–3.8) on the root causes, and each section is further grouped into causes. Thus, attacks that are evoked by the same cause and root cause are logically grouped together. Regarding the structure of this document, we traverse the systematization process (Figure 3.4) backwards from root causes, to causes, to attacks.

#### 3.3.2 Security Requirements

Security requirements describe the demands that need to be met by the system in order to protect the interests of its stakeholders. For our systematization, we aim to establish generic and long-lasting security requirements spanning all three mobile network generations. However, the standardization bodies, e. g., 3rd Generation Partnership Project (3GPP), have issued diverging requirements over time, which is why, they do not allow us to provide a holistic view and might not fit modern security concepts. In order to define generic and long-lasting security requirements, we therefore base our work on the publication of Avizienis et al. [51] who define a taxonomy of dependable and secure computing and general security requirements that we underpin with some more concrete requirements published by the 3GPP [21, 23, 25].

#### 3.3.2.1 Confidentiality

Avizienis et al. [51] define confidentiality as the “*absence of unauthorized disclosure of information*”. This statement is substantiated by the 3GPP with the following requirement: “*the network shall provide several appropriate levels of user privacy including communication confidentiality, location privacy, and identity protection*” [23, p. 33]. In detail, this means:

- Communication confidentiality: “[...] *contents, origin, and destination of a particular communication shall be protected from disclosure to unauthorized parties*” [23].
- Identity protection: The network shall “*hide the identities of users from unauthorized third parties*” [23].
- Location privacy: The network shall “*hide the user location from unauthorized parties*” [23].

#### 3.3.2.2 Availability

Availability denotes the readiness and the continuity of correct services [51]. With the pervasiveness of mobile communications in our everyday lives, availability becomes a crucial factor for customers as well as part of critical infrastructures [120, 290].

#### 3.3.2.3 System Integrity

In contrast to data or transmission integrity, system integrity focuses on the hard- and software of the network components. Integrity is defined as the absence of unauthorized system alterations [51]. System integrity is an essential security requirement as it is crucial for the proper operation and trustworthiness of the system.

#### 3.3.2.4 Unauthorized Service Access and Correct Charging

The service should only be accessible to authorized parties [23]. This requirement includes correct recording and offsetting call data records and other chargeable items [22]. In other words, a proper authorization and charging system should only allow subscribed services to be consumed and it should charge the *right* user for the *correct* volume [242].

Subsequently, we will use these high-level security requirements to assign an attack aim to each identified attack that challenges one or more of those requirements.

#### 3.3.3 Attack Aims

Each attack has a clear primary aim that challenges one of the identified security requirements. An attacker might also pursue a secondary attack aim. For example, using side-channels, an attacker can obtain the shared key on the SIM card that undermines primarily the secrecy aim. However, the attacker might also clone the SIM card for free calls and thereby commit fraud attacks (secondary). We define five distinct attack aims:

- **Attacks on Privacy:** This aim covers all attacks that undermine the privacy of the user, including the *identity protection* and the *location privacy*.
- **Attacks on Secrecy:** This category includes attacks on *communication confidentiality*, e. g., the content of the transmission.
- **Denial of Service:** This attack aim contains all the objectives that impact the *availability* of services, or parts of them. Thus, *downgrade attacks*, such as disabling encryption or stepping back to less secure protocols belong here.
- **Attacks on Integrity:** This category comprises all the attempts which undermine the requirements for system integrity.
- **Fraud Attacks:** This aim covers attacks that aim towards directly or indirectly targeting financial benefits for the attacker or financial losses for others. *Direct under-billing* attacks dodge service charges at the expense of the operator, whereas *direct over-billing* produces financial loss to customers. *Indirect fraud* includes scams or spam via telephone.

### 3.3.4 Attacks

Attacks exploit system flaws under the defined attack aims. We use the following characteristics for an assessment of the attack impact (see Table 3.1). In general, as for Table 3.1, ● denotes a fully applicable attack for the characteristic, ◐ refers to limitations, and ○ characterizes attacks that are not applicable.

#### 3.3.4.1 Attacker Capabilities

For mobile radio attacks, an attacker often combines several capabilities to perform an attack (e. g., retrieving session keys over SS7 and passively monitoring and decoding traffic); thus, we describe the attacker model as a set of distinct capabilities (i. e., building blocks). We assume that the attacker is a-priori not in possession of any *private* information (secret keys) of the victim, but might be in possession of *public* identifiers such as the phone number (MSISDN).

- **Passive Radio:** An attacker with passive radio capabilities is able to capture radio transmissions, decode signals, and read raw messages. The recent developments of Software Defined Radios (SDRs) and re-purposed hardware render this type of attack quite affordable [134, 234].
- **Active Radio:** An attacker with this capability has full control over radio transmissions and is therefore able to put arbitrary messages on the radio channel. This enables an attacker to setup a own base station or a fully controllable phone stack using an SDR [134, 213, 225, 234, 252, 327].

- **User Traffic:** The attacker is able to control or initiate traffic on a commodity mobile phone. The phone performs normal radio emissions, but the attacker accesses the higher (user-land controlled) network layers (e. g., IP) or dedicated network services (e. g., SMS). In most cases, this ability does not require a rooted or jail-broken phone.
- **SS7/Diameter Interface:** An attacker with access to SS7/Diameter is able to send and receive Signalling System #7 or Diameter messages to and from other networks. Some network providers even sell these services [292].
- **Nondestructive Physical:** A nondestructive-physical attacker temporarily has physical access to the victim's device, but neither destroys nor modifies hardware or software. Thus, the attack leaves no visible or measurable trace. We exclude the destructive attacker, because these visible traces would raise doubts by the users.
- **PSTN Interface:** An attacker has access to voice or text services of the Public Switched Telephone Network (PSTN).
- **Internet Traffic:** An attacker with the ability to access the Internet in a way that can specifically contact the victim's phone. That can be achieved by knowing the phone's public IP address and the TCP/UDP port mapping on the operator's packet gateway. Other possibilities include identifiers of chat services, instant messaging apps, or cloud messaging services (such as Google Cloud Messaging (GCM) [137] or Apple's Push Notifications [47]), and the ability to transfer such messages.

#### 3.3.4.2 Limitations of Attacker Capabilities

For our systematization, we assume that the operator's authorized personnel is trusted and thus exclude such attacker capabilities from systematization. However, such attacks have occurred in the past and are a threat to the mobile user's data secrecy and privacy [248, 269]. For instance, in the 2005 Vodafone Greece incident [248], a staff technician was suspected to have planted a backdoor in mobile switches that allowed copying traffic on government phones. In the Gemalto SIM key material theft [269], secret key material for the SIM cards was transferred by the use of unprotected means. However, such attacker capabilities are beyond the scope of this paper, as the attackers had the permissions in the first place and deliberately misused them.

#### 3.3.4.3 Target

The target category depicts who is harmed by the attack, and if there is a relation to other categories, e. g., privacy attacks predominantly target the user. We focus on the primary goal and disregard secondary effects such as bad publicity due to data breaches.



Table 3.1: Categorization of mobile security attacks by their aim

Aim	Attack	Attacker Capabilities						Target	Technology			Range				Section		
		Radio Passive	Radio Active	User Traffic	SS7 Interface	PSTN Interface	Internet Traffic		Nondestructive Physical	User / Provider	2G	3G	4G	Phy	Cell		LA	Net
Service	Signaling DoS [59, 172, 186, 188]	●	●	○	○	○	○	○	U,P	●	●	●	○	●	○	○	○	3.5.5.1
	Attach Request Attack [338]	●	●	○	○	○	○	○	P	●	●	●	○	●	○	○	○	3.5.5.1
	GPS Receiver Denial of Service [53, 119]	○	●	○	○	○	○	○	U	○	○	○	○	●	○	○	○	3.5.1.1
	Continuous Wideband Jamming [52, 168, 194, 329]	○	●	○	○	○	○	○	U,P	●	●	●	○	●	○	○	○	3.8.1
	Protocol-Aware Selective Jamming [168, 194, 254, 329]	●	●	○	○	○	○	○	U,P	○	○	●	○	●	○	○	○	3.8.1
	IPv4/IPv6 Middleboxes Misconfiguration [155, 188]	○	○	●	○	○	○	○	U,P	○	○	●	○	○	○	●	○	3.7.1.1
	SMS Link Saturation [109, 293]	○	○	○	●	○	○	○	U,P	●	●	●	○	○	○	○	●	3.5.5.1
	Paging Response Race DoS [132]	●	●	○	○	○	○	○	P	●	?	?	○	●	○	○	○	3.5.1.1
	DDoS HLR: Activate Call Forwarding Request [294]	○	○	●	○	○	○	○	P	○	○	○	○	○	○	○	○	3.5.5.1
Insert/Delete Subscriber Data into the VLR/MSC [111]	○	○	○	●	○	○	○	U	●	●	●	○	○	○	○	●	3.5.4.1	
Secrecy	(U)SIM: COMP128v1 and MILENAGE Side-Channels [195, 253, 341]	○	○	○	○	○	○	●	U	●	●	●	●	○	○	○	○	3.6.2.1
	Baseband State Machine Exploits [207, 233, 264, 324]	○	●	○	○	○	○	○	U	●	●	●	○	●	○	○	○	3.6.1.1
	Encryption Downgrade [90, 91, 201, 228, 273, 281]	○	●	○	○	○	○	○	U	●	○	○	○	●	○	○	○	3.5.2.1
	SIM Key Extraction via COMP128v1 Cryptoanalysis [71, 73, 171]	○	○	○	○	○	○	●	U	○	○	○	○	○	○	○	○	3.5.3.1
	Weak Key due to Inter-Technology Handover [205]	●	●	○	○	○	○	○	U	○	○	?	○	○	○	○	○	3.5.3.1
	Inter eNodeB User Plane Key Desynchronization Attack [147]	●	○	○	○	○	○	○	U	○	○	●	○	○	○	○	○	3.5.3.1
	Key Reusage Across Cipher and Network Generations [55]	●	●	○	○	○	○	○	U	●	○	?	○	○	○	○	○	3.5.3.1
	MitM IMSI Catcher [90, 91, 201, 228, 273, 281]	●	●	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.2.1
	Passive Over-the-Air Decryption of A5/1 and A5/2 [55, 65, 72, 72, 133, 183, 229, 232]	○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.3.1
	Intercepting Calls with SS7/CAMEL [111]	○	○	○	●	○	○	○	U	●	●	?	○	○	○	○	●	3.5.4.1
	Session Key Retrieval via SS7 [111, 228]	○	○	○	○	○	○	○	U	●	○	?	○	○	○	○	○	3.5.4.1
	Privacy	AKA Protocol Linkability Attack [48, 67]	●	●	○	○	○	○	○	U	○	○	●	○	○	○	○	○
IMSI Paging Attack [48]		●	○	○	○	○	○	○	U	●	●	●	○	○	○	○	○	3.5.1.1
Location Leak by SIP Message [77]		○	○	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.7.1.1
Location/Tracking Area not Allowed (Downgrade) [27, 275, 340]		●	●	○	○	○	○	○	U	○	○	●	○	○	○	○	○	3.5.1.1
Measurement Reports Localization [121, 275]		●	●	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.5.1.1
OTA SIM Card Update Key Reconstruction [227]		○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.3.1
Unauthenticated IMEI Request [78, 91, 201, 228, 273, 281]		●	●	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.1.1
Unauthenticated IMSI Request (IMSI Catcher) [78, 90, 91, 201, 213, 228, 273, 281]		●	●	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.1.1
TMSI Deanonymization (Paging Attack) [154, 181, 275]		○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.1.1
Cell-Level Tracking with SS7/MAP [111]		○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.4.1
GPS Location with SS7/LCS [111]	○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.4.1	
Integrity	ASN.1 Heap Overflow [159]	○	○	○	○	○	○	○	U,P	●	○	○	○	○	○	○	○	3.6.1.1
	Binary Baseband Exploit [131, 310, 323]	○	○	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.6.1.1
	SMS Parsing [82, 216]	○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.6.1.1
	SIM Card Rooting [227]	○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.6.1.1
Fraud	Fake Base Station SMS Spam [192, 315]	○	○	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.5.2.1
	LTE IMS-based SMS Spoofing [77, 295]	○	○	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.7.1.1
	Misbilling: TCP Retransmission or DNS Tunneling [127, 127, 128]	○	○	○	○	○	○	○	P	○	○	○	○	○	○	○	○	3.7.2.1
	Underbilling using VoLTE Hidden Channels [177, 190]	○	○	○	○	○	○	○	P	○	○	○	○	○	○	○	○	3.7.2.1
	Uplink IP Header Spoofing/Cloak-and-Dagger Misbilling [242, 321]	○	○	○	○	○	○	○	U	○	○	○	○	○	○	○	○	3.7.1.1
	Unblock Stolen Devices [255]	○	○	○	○	○	○	○	U	●	○	○	○	○	○	○	○	3.5.4.1

● yes, applicable, needed for attack      ○ partially/supportive/optional  
 ○ no, not applicable, or does not apply      ? property unknown

#### 3.3.4.4 Technology

This category maps the applicability of an attack to the three major access technology generations and assesses if there has been a security development, e. g., if defenses have been introduced in later access technologies, or if new technologies open new attack vectors. The former does not necessarily prevent attacks, as multiple downgrade attacks are known.

For example, only GSM lacks mutual authentication (*cause*), hence it is prone to Man-in-the-Middle (MitM) attacks. However, UMTS and LTE are open to various downgrade attacks; therefore the problem will not be resolved until phones stop to (unconditionally) support GSM. Downgrade attacks that trick or force a specific party to fall back on older and less secure access technology must be kept in mind when discussing fixes or mitigations for new access technology generations. We filed downgrade attacks as part of Denial of Service (DoS) attacks, as they deserve a separate review.

#### 3.3.4.5 Range

The range of an attack is an indirect indicator of impact and cost. A higher range (e. g., a globally performable attack) increases its impact and versatility and might justify higher costs for an attacker. In contrast, an attack that requires more physical vicinity increases involvement of the attacker and reduces the set of victims. In Table 3.1, we classify the range by technical boundaries: Physical access (**Phy**), same radio cell including simulated ones (**Cell**), same location area (**LA**), same network (**Net**), and globally executable attacks (**Glo**).

#### 3.3.5 Flaw

For our systematization, we define a flaw as a specific and distinct vulnerability that is exploited by a particular attack. We coalesced attacks that exploit the same technical flaw or are otherwise very similar in their technical or operational principle. This leads to a 1:1 relationship between flaws and attacks.

#### 3.3.6 Cause

We group flaws that have similar technical or organizational reasons via a common cause. A cause is a broader technical reason summarizing multiple individual flaws and, if dealt with appropriately, would foil an entire class of attacks.

#### 3.3.7 Root cause

Root causes are the underlying reason for certain classes of attacks; they are defined on an abstract level and independent from technology. Each root cause summarizes particular flaws and vulnerabilities according to their structural or causal dependency (Table 3.2) and is completely disjoint from other root causes. We use this abstract structure as the foundation for our systematization.

Table 3.2: Root causes related to causes

Root Cause	Cause
Specification Issue	Unsecured Pre-Authentication Traffic
	Non-Existing Mutual Authentication
	Weak Cryptography
	Resource Usage Asymmetry
	Insecure Inter-Network Protocol
Implementation Issue	Insecure Implementation
	Leaky Implementation
Protocol Context Discrepancy	Cross-Layer Information Loss
	Accounting Policy Inconsistency
Wireless Channel	Channel Characteristics

### 3.4 Cause Overview

In the following, we introduce the abstract *root causes* and their individual *causes* as used in this systematization (see Table 3.2). Later, we present each cause in detail in relation to attacks based on the outcome of our process displayed in Figure 3.4. The results regarding attacks are condensed in Table 3.1 and as graph in Figure 3.5.

*Specification Issues* originate from incomplete, inaccurate, or faulty definitions of system behavior and comprise five individual causes: *Unsecured Pre-Authentication Traffic* allows to send messages to the phone or network prior to the key agreement and ciphering setup has taken place. *Non-Existing Mutual Authentication* relates to an attack vector exclusive to GSM networks, but is still an issue in recent technologies due to downgrade attacks. The use of *Weak Cryptography* significantly reduces the required effort for attacks on encrypted data, while *Insecure Inter-Network Protocols* undermine the users' privacy and confidentiality by poor protocol design choices. *Resource Usage Asymmetry* enables an attacker to perform cheap requests that result in intensive operations on the network side and hence can lead to DoS.

*Implementation Issues* are either caused by *Insecure Implementations* that open attack vectors in components of the mobile network, which can impair the system's integrity, or by *Leaky Implementations*, which means that sensitive information can be accessed through unintended side channels.

The root cause *Protocol Context Discrepancy* summarizes a class of security issues which use a protocol differently or in another environment than it was originally designed for. *Cross-Layer Information Loss* occurs at the interface of different layers in the network stack, e. g., when necessary, trustworthy security information is lost between network layers. *Accounting Policy Inconsistencies* result from different accounting schemes that can be played against each other, e. g., voice data is charged in minutes, whereas other

data is charged by volume.

The *Wireless Channel* and its characteristics is essential for the transmission of information in mobile communication and comes with several physical limitations that impact the security.

Following the structure of root causes, we discuss offensive and defensive characteristics of specific causes and derive open research questions and challenges for future mobile network technologies.

## 3.5 Root Cause: Specification Issue

Specifications ensure the interoperability between implementations by specifying protocols, state machines, and interfaces. However, there may exist issues in the specification that might lead to flaws that can be exploited by attackers. In the specification-related root cause, we collate all flaws that are based on specification issues. The reasons for these problems range from technical trade-offs to political motivation.

### 3.5.1 Cause: Unsecured Pre-Authentication Traffic

The signaling traffic prior to the security establishment with the AKA protocol is unprotected: it is neither encrypted nor integrity-protected, and thus unauthenticated<sup>1</sup>. This leads to implicit trust between the phone and the network. In this unauthenticated state, the phone fully obeys the network, even if the latter is not genuine. A malicious usage of messages in this unauthenticated state can serve for downgrade, track, or locate a specific user or handset.

#### 3.5.1.1 Attacks

One prominent example for attacks based on unsecured pre-authentication is the deployment of *fake base stations*. Fake base stations (also known as *rogue* or *fraudulent base stations*, *IMSI Catchers*, *cell-site simulators*, a *DRT-Box*, or by product names such as *Stingray*) are active devices simulating a genuine base station to the phone by broadcasting genuine network identifiers. These fake base stations exploit the fact that mobile phones cannot verify the authenticity of the network prior to the AKA protocol.

In the unauthenticated state, the base station (thus also the attacker) is allowed to ask for the permanent identity, such as the IMSI or IMEI, and can thus undermine the user's identity and location privacy [27, 67, 78, 201, 213, 228, 273]. Besides obvious requests such as the *identity request*, an attacker can also use more subtle ways to determine the vicinity of a victim, e. g., with the AKA linkability attack [48]. Additionally, an attacker can repeatedly page the victim's IMSI [48] and, thereby, determine if a user is in radio range. Moreover, an attacker can retrieve a more precise location by requesting measurement

---

<sup>1</sup>In LTE some uplink data is integrity protected but not encrypted.

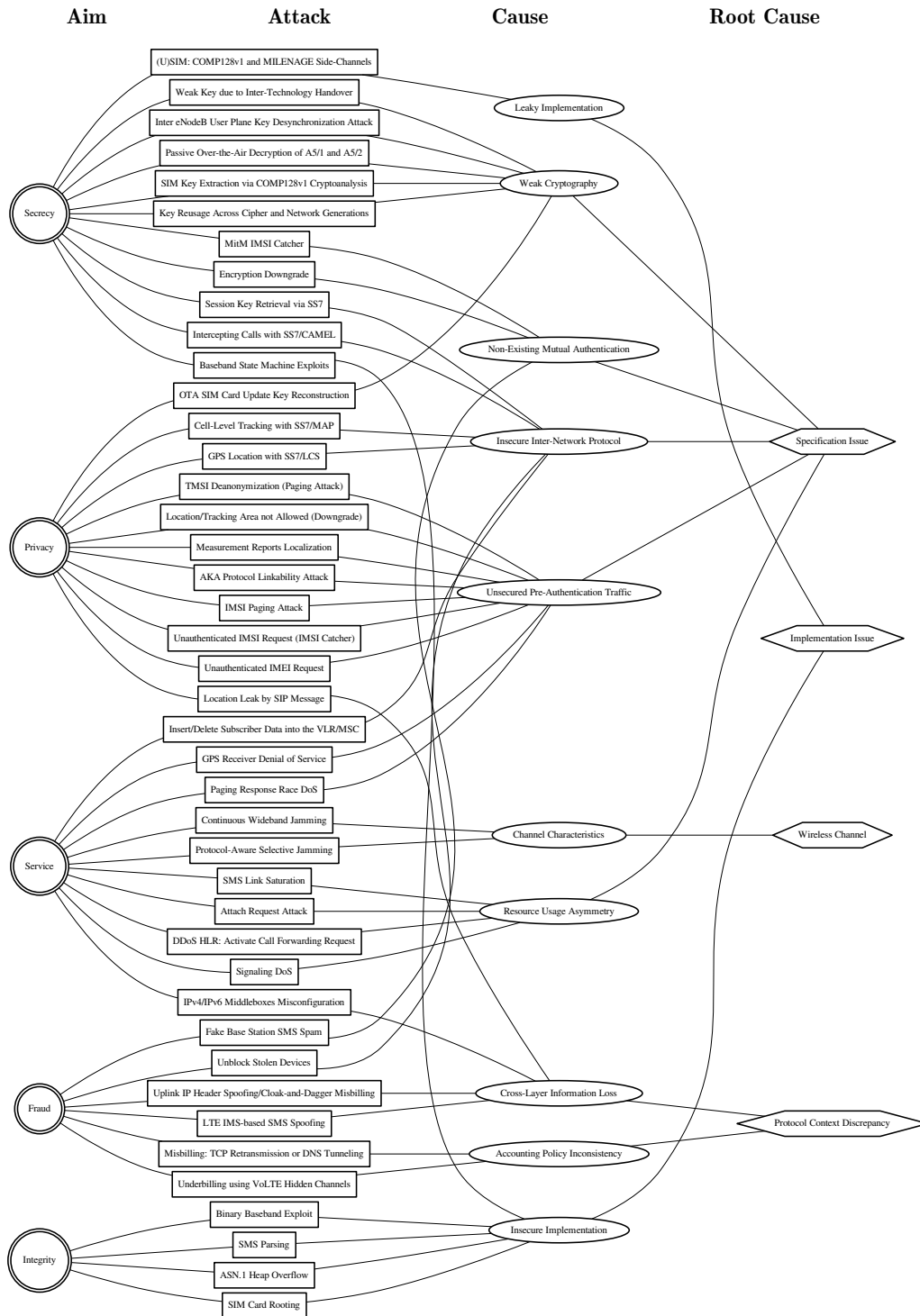


Figure 3.5: Visualization of Systematization including Attack Aims, Attacks, Causes, and Root Causes.

reports from the victim's handset [121,275] enabling an attacker to track a victim or to request the identity of people within radio range.

Furthermore, unsecured pre-authentication traffic allows *downgrade attacks* to a less secure access technology by denying service using the *tracking area update reject* or a combination of other messages [27,275,340]. This serves as a stepping stone for further attacks such as the GSM MitM attack (Section 3.5.2.1). A more detailed description on the function and operational principles of fake base station can be found in Section 3.9.

Additionally, a fake base station can disable the location service on some phones from the late 2000s by sending out the country code of Egypt. At this time, GPS receivers used to be forbidden in Egypt [325, p. 28], [53, 119]. Furthermore, Golde et al. [132] showed that unsecured pre-authentication uplink traffic in GSM can be misused for a DoS attack dropping calls in the entire location area by winning the race answering paging requests. This is a problem of the GSM state-machine specification, as it can not recover once it proceeds to the ciphering setup.

The missing protection of broadcast and paging messages also enables attacks that retrieve the temporary identity of a victim by triggering the paging process multiple times and statically analyzing the paged TMSIs [181,275]. An attacker can trigger the paging procedure in multiple ways: for example, with targeted Internet traffic, a short phone call and immediate hang-up, e.g., before the ring starts, or with a *Silent SMS* that is a text message which is silently discarded by the phone [24].

**Assessment** Attacks based on unauthenticated uplink traffic or on passively exploitable downlink traffic are vastly outnumbered by active radio attacks based on pre-authentication traffic. While potentially having a very severe impact, an active radio attacker is limited to his/her radio vicinity. Most of these attacks undermine the victim's data or location privacy. Many commercially available products exploit unsecured pre-authentication traffic [63,245], hence making it a high priority to be addressed.

#### 3.5.2 Cause: Non-Existing Mutual Authentication

The original specification of GSM does not include network authentication and, thus, allows a MitM attack. While the 3G AKA can be used in GSM if supported by all parties, no downgrade prevention exists [204].

Although the fact of non-existing mutual authentication originally exploits a specific vulnerability of GSM networks, they are still a relevant threat in today's networks as the weakest-link-principle applies. Downgrade attacks via unsecured pre-authentication traffic on UMTS or LTE (Section 3.5.1) still allow to exploit this GSM vulnerability on modern phones. The difference to pre-authentication traffic (Section 3.5.1) is the lack of mutual authentication. In that sense, the non-existing mutual authentication is an extension of the unsecured pre-authentication traffic issue and has similarities in attacks and defenses with the former cause.

### 3.5.2.1 Attacks

If the phone cannot verify the authenticity of the network, an unconditional trust of the phone to the network and, thus, to a potential attacker is established. In a network-centric architecture, where most decisions are made by the network, an attacker faking a base station gains excessive power over the handset.

Fake base stations often employ additional techniques to keep a victim in the fake cell, such as not supplying information on neighboring cells or manipulating cell reselection thresholds [91]. The phone behaves inconspicuously and is able to make phone calls as well as send text messages and data to the fake network. However, without any further exploit, the attacker can not gain the possession of the cryptographic keys. Still, the attacker can downgrade the communication to the null-cipher or an easily attackable cipher (see Section 3.5.3.1) for passing it to the real network. In this case, the phone remains reachable for the genuine network. Alternatively, calls, SMS, and data could be forwarded with additional modems or SIP, in which case the original caller-ID is lost, and the phone is not reachable from the outside. The impact of the attacks can be increased by an attacker with SS7 capabilities, e. g., she/he can directly inject the traffic into the phone network.

Similar to measurement reports on LTE, the GSM radio resource location service protocol enables the network to request GPS coordinates from the phone [19]. Developed for emergency services, most phones will answer the request even though it is not an emergency call [324]. Additionally, the non-existing mutual authentication has been a rich source for location-based SMS spam—mainly in China [192, 217]. Commercially available fake base stations with ready-to-use exploits are a reminder of the urgency with which this threat should be addressed.

### 3.5.3 Cause: Weak Cryptography

Cryptography provides the means to achieve data confidentiality. However, weak cryptography can lead to attacks revealing protected information. This can emerge from intentionally weakened algorithms or by evolving attack methods [240]. Cryptographic systems should be designed following Kerckhoffs' principle [173], which states that a system should only rely on the secrecy of the key, while everything else might be known to the attacker (or the public). In the following, we describe attacks that aim specifically at weak cryptography.

#### 3.5.3.1 Attacks

Cryptography is used for the encryption algorithms on the air interface, for the handover, and initial key derivation. In all these parts, we identify attacks due to the use of weak cryptography. The found attacks undermine the data confidentiality requirement, either by breaking the used session key or the shared key on the SIM card.

**Encryption Algorithms** Table 3.3 depicts the air interface encryption algorithms for all three generations, in particular the type of cipher, the effective key length, and if the cipher is attackable. All cipher suites in GSM except for A5/{3,4} are breakable within minutes on commodity hardware for different reasons. A5/1 is a 64-bit cipher based on three connected Linear Feedback Shift Registers (LFSR) with major cryptographic vulnerabilities that have led to passive decryption attacks [55, 65, 72, 133, 183, 232]. A5/2 was designed as a stripped-down export version of A5/1 with a shorter effective key length, and Goldberg et al. [130] showed how to break this cipher in near real-time. Rainbow table approaches are further eased by the predictable padding of messages [229]. Theoretical attacks exist against KASUMI but they are impractical in terms of space requirements, as they require  $2^{26}$  captured data frames [103] and do not directly translate into A5/3. A5/3 rainbow tables similar to A5/1 were proposed but never published [237]. GPRS ciphers GEA $x$  basically mirror the weaknesses and development of their A5/ $x$  counterparts [230]. In GSM, the cipher-capabilities of the network and the user device are not integrity-protected and are therefore vulnerable to manipulation. An attacker can change the encryption handshake to block A5/3 and force a downgrade to A5/1.

In contrast to GSM, all UMTS and LTE ciphers underwent public development and thus

Table 3.3: Cipher Overview

	Cipher	Type	Effective (nom.) key length	Attackable
2G	A5/0	Null Cipher	–	●
	A5/1 + Comp128v1/2	LFSR-based	54 (64) bits	●
	A5/1 + Comp128v3	LFSR-based	64 bits	●
	A5/2	LFSR-based	40 (64) bits	●
	A5/3	KASUMI	64 bits	◐
	A5/4	KASUMI	128 bits	○
	GEA1	LFSR-based	64 bits	●
	GEA2	LFSR-based	64 bits	●
	GEA3	KASUMI	64 bits	◐
	GEA4	KASUMI	128 bits	○
3G	UEA0	Null Cipher	–	●
	UEA1	KASUMI	128 bits	○
	UEA2	SNOW 3G	128 bits	○
4G	EEA0	Null Cipher	–	●
	EEA1	SNOW 3G	128 bits	○
	EEA2	AES	128 bits	○
	EEA3	ZUC	128 bits	○

○ not attackable      ● attacks with commodity hardware known  
 ◐ attacks known, but not practicable or not demonstrated



followed Kerckhoffs' principle. As a consequence, only one attack against the KASUMI based UEA1 algorithm was revealed, but still requires an unpractically large amount of captured data [104,164].

Additionally, each generation has a null cipher that offers no protection. Since the networks select the encryption algorithm, the user is unaware of sending data in clear text. A ciphering indicator should warn the user on the UE. However, just few vendors implement such a ciphering indicator [15,17,45,264].

**Interoperability of Access Technologies** Interesting problems arise due the usage of same key material within the same generation or due to the interconnection of network generations. Barkan et al. [55] describe that it is possible to downgrade to a less secure cipher for a short period of time or to reconstruct the key passively from over-the-air communication and later use it for all the other (secure) ciphers. In order to allow a GSM SIM to connect to UMTS, the key is extended to meet the UMTS key length [20,204,205]. Also, a USIM operating on GSM will simply use a shortened key. Thus, an attack on the much weaker A5/ $x$  series reveals parts of the key information for other access technologies during handover. Additionally, the LTE handover is vulnerable to the so-called “desynchronization” attack [147]. As shown in simulations, an attacker can desynchronize the used key with the core network and, thus, an old session key is reused.

**Key Derivation** Weak cryptography is also used for the initial key derivation algorithms. By reverse-engineering and breaking the COMP128v1 key derivation algorithm of GSM an attacker can reconstruct the shared secret key [71,73]. In combination with side-channels of some poorly protected implementations, COMP128v1 attacks can be brought down to nearly instant key recovery [171] (see Section 3.6.2). Another attack by Nohl et al. [227] on SIM cards can remotely reconstruct the SIM's software update key based on weak DES encryption or poorly implemented 3DES (proper implementations are safe). They leveraged the fact that the error messages concerning ciphering are sent encrypted with a known plaintext. As this attack is delivered via SMS, there are no proximity limitations to the attacker. A reconstructed OTA key enables the attacker to install new applications on the SIM card, subsequently accessing secrets stored in other applications (see Section 3.6.1).

**Assessment** All the attacks based on weak cryptography primarily undermine the data confidentiality aim of mobile networks. The attacker might also pursue a secondary aim. For example, the shared key obtained through SIM attacks can later be used to decode encrypted transmissions or write the keys on a freely programmable SIM card. Such cards can be used to impersonate a subscriber, redirect calls, change settings, or commit fee fraud. While attacks on the air interface can be executed by an attacker with passive radio capabilities, the attacks on the SIM card require physical access— thereby are thereby either limited to the radio transmission range or to the physical range. The attacks on the session keys are possible using affordable methods such as rainbow tables on an ordinary PC. GSM is especially prone to cryptography attacks. In contrast, newer

generations rely on secure algorithms following the Kerckhoffs' principle such that these attacks are not known for now.

#### 3.5.4 Cause: Insecure Inter-Network Protocols

Nowadays the telecommunication industry is deregulated and SS7 has been ported to an IP-based network. Both developments make SS7 easily accessible. Thus, an attacker with SS7 capabilities becomes more likely. However, for interconnectivity with the SS7 networks, SS7 messages are translated to Diameter. This makes Diameter also vulnerable to SS7 attacks, as this inter-working function does not provide authenticity. Even though Diameter was designed with security features based on protocols like TLS and IPsec, researchers found vulnerabilities in the dedicated Diameter protocol that do not rest upon the inter-working function of SS7.

##### 3.5.4.1 Attacks

The general idea of SS7 attacks is to request services on different layers of the home network or the serving network. As SS7 offers no authentication mechanisms, the network entities cannot decide if the request is legitimate. Thus, the entity replies properly, even though the request might not be legitimate.

An attacker can determine the user location on different levels of granularity—in the range of cells up to exact GPS coordinates [110, 111, 153]. Additionally, an attacker can map the temporary identity (TMSI) to the permanent identity (IMSI) of a victim by using the SS7 system. The permanent identity can then be mapped to the public telephone number. Both attacks are not compliant with the identity confidentiality aim. The misuse of SS7 can also lead to attacks that undermine the confidentiality of calls or of text messages [111]. This can be done by rerouting calls or by requesting the over-the-air encryption key. Besides this, the insecurity of SS7 can also be exploited for fraud attacks by unblocking a stolen device [255]. Additionally, an attacker can run a precise DoS attack against a distinct user by deleting subscriber data in the VLR [111]. Attacks that are possible due to the inter-working function between Diameter and SS7 are discussed by Holtmanns et al. [153] and Rao et al. [256]. Even Diameter has been found vulnerable and allows to intercept text messages [152].

**Assessment** The insecurity of SS7 leads to a wide range of attacks. Most of them aim to undermine the (location) privacy of the user. Even commercial services were built upon the insecurity of SS7 allowing to pinpoint and track a victim [111, 292, 314]. This shows that the SS7 vulnerabilities are actively used and are thereby a serious threat to users. Most of the attacks require SS7 capabilities of the attacker. However, some attacks can be accomplished by using passive radio capabilities, e.g., an attacker can decrypt the traffic as soon as the over-the-air encryption key is revealed.

### 3.5.5 Cause: Resource Usage Asymmetry

*Resource usage asymmetry* occurs when an simple action on one side triggers a computationally or resource-wise expensive reaction on the other side. This—for example—leads to signaling DoS attacks, during which an attacker misuses signaling/control messages to trigger an expensive action. Thus, the network allocates the resources within different components and may eventually run out of them after repeated or coordinated requests.

#### 3.5.5.1 Attacks

Unauthenticated messages like those used in the *attach procedure* can be utilized to overload the core network components [59, 338]. Additionally, they can impersonate legitimate subscribers. Similarly, Lee et al. [186] have presented signaling attacks for 3G networks and argue that low-volume but well-timed signaling attacks can have a major impact on the network components. By misusing multiple messages for establishment and release of radio connections, the authors caused a significant increase of message load in the network. Traynor et al. [294] evaluated network attacks targeting the HLR<sup>2G,3G</sup>. They found an effective method to tear down an HLR by frequently switching the call forwarding service on and off. They suggest that a mobile phone botnet can disable the service of an HLR.

Similarly, a mobile phone botnet could attack a 911 response center, which would result in an outage of emergency services [144]. While this is not exclusively related to mobile-phone networks, the elevated priority of emergency calls makes it a unique mobile network problem: The network will drop other connections in favor of emergency calls if necessary. Enck et al. [109] evaluated attacks considering the to open SMS functionality on the Internet. They analyze an attacker model that uses open SMS centers on the Internet to saturate the wireless link downstream from the base stations, obstructing the service in the whole cell.

**Assessment** All the attacks based on resource usage asymmetry focus on an exhaustive denial-of-service of the network. However, the impact of these attacks vary. While some attacks require active radio attacker capabilities, others already work with Internet capabilities. Besides intentional disturbance of the service, similar problems can occur due to misconfigured mobile apps or unexpected user behavior [112, 336].

## 3.6 Root Cause: Implementation Issue

Deviations of the implementation from the original specification can open attack vectors and, thus, can have a security impact on otherwise securely defined systems. Such deviations can be introduced on purpose, e. g., for compatibility trade-offs, or result from faulty implementations. In the following, we discuss the implications of insecure and leaky implementations.

### 3.6.1 Cause: Insecure Implementation

While insecure implementations can open attack vectors in deployed systems, current research mainly focuses on attacks on the baseband and SIM cards. By sending malicious data to vulnerable devices, an adversary can exploit implementation issues. In the following, we discuss how attacks undermine the system integrity, availability, secrecy, and privacy including potential countermeasures.

#### 3.6.1.1 Attacks

The lower layers of the protocol stack run on distinct baseband processors in the UE. Parser errors within the baseband processor can occur due to faulty implementations of parsing modules or libraries threatening the device's integrity. In 2016, a heap overflow in a widely used ASN.1 compiler was discovered [135, 159] affecting baseband implementations of multiple manufacturers. Weinmann [323] and Golde [131] demonstrated how to use baseband exploits to further target the application processor and its operating system.

Crashing-only flaws in the parsing and decoding stage of text messages [82, 216, 310] make the phone inoperable until the next reboot. Similar flaws on SMS parsing have been found on other processing levels [33].

Apart from attacks on the baseband, Nohl et al. [227] showed that the application isolation on the SIM card is so weak that processes can access foreign data including authentication credentials. Such applications can be remotely installed after reconstructing the over-the-air (OTA) update key (see Section 3.5.3.1).

Implementation flaws in the protocol state machines of the baseband result in the acceptance of a fake base station as a genuine network endangering data secrecy and privacy [207, 233, 264, 324].

**Assessment** On the one hand, we see that attacks can be launched globally and in a targeted manner that makes the impact of these flaws very high. On the other hand, the most dangerous ASN.1 heap overflow and the staged baseband-to-application-processor attacks required a fake base station with active radio capabilities and is thus locally bounded. The danger lies in the potential to take over the device at the lowest level.

### 3.6.2 Cause: Leaky Implementation

Implementations in software and hardware can leak information about internal states in surprising or non-obvious ways. Besides using a provable secure, an implementation might leak enough information to circumvent the strong security measures due to the implementation insufficiencies.

#### 3.6.2.1 Attacks

The SIM card stores the secret key for authentication and key derivation. Gaining access to this information breaks the security concept at its very core enabling decryption and

impersonation.

Rao et al. [253] and Zhou et al. [341] have built a key reconstruction attack upon the cryptanalysis of Comp128v1 on Global System for Mobile Communications (GSM) SIM cards with chosen plaintexts and by using electromagnetic field probes. In 2015, Liu et al. [195] found that the AES-based *MILENAGE* algorithm on USIM implementations is susceptible to power-based side-channel analysis and were thus able to extract the secret key.

**Assessment** The primary aim of such attacks is gaining access to the secret key and, thereby, undermining the confidentiality requirement. However, once the key is known to the attacker, he/she might fulfill secondary attack aims. It may enable him/her to decrypt the radio communication with passive radio capabilities or to impersonate a subscriber by cloning the SIM card.

Even though the aforementioned attacks reveal one of the most valuable secrets in mobile networks, the attacks require temporary physical access to the SIM card. Thus, forging SIM card clones is more likely to happen through an internal attacker or through the device owners themselves than through external attackers.

## 3.7 Root Cause: Protocol Context Discrepancy

This root cause is based on protocol context issues that are due to deploying a protocol that is not originally intended for the mobile network environment. Protocol properties are not harmful in a non-mobile network environment, but may be exploitable in a mobile environment if not adjusted properly.

### 3.7.1 Cause: Cross-Layer Information Loss

The layering of network stacks serves multiple important purposes such as implementation transparency (e. g., upper layers do not have to care about details of lower layers) and interoperability (e. g., upper layer applications can span or exchange data over multiple networks). However, such layering also means loss of information that might be needed at higher levels, e. g., at some point, IP addresses or connections need to be mapped to the subscriber identity.

#### 3.7.1.1 Attacks

The lack of a strong binding between radio-level authentication and IP-service authentication is the source for multiple vulnerabilities. The literature show that the implementation of such mapping is vulnerable and can be tricked with simple IP-based attacks, such as spoofing of IP addresses [242, 321]. IP address spoofing can be exploited for over- and under-billing attacks and to reverse the isolation of the internals to the Internet network. IPv4 and IPv6 NAT middleboxes pose a threat to the users as well as for the mobile network operator [155, 188]. Similarly to the NAT middleboxes, the Packet Data Network

Gateway (P-GW) routing configuration seems to be a problem in cases that allow direct communication between two phones [177, 190]. Another related problem is the lack of security checks within the SIP-protocol. Manipulated SIP headers can be used to fake the caller ID with UE-originated SMS messages [295].

**Assessment** All these attacks consider an attacker able to initiate user traffic and optional Internet traffic capabilities. Hence, all the attacks can be easily realized. The range of those attacks is network-wide, thus an attacker can be anywhere in the network and exploit the flaw. We see the trend that newer generations—especially LTE—are more prone to attacks that are based on cross-layer information loss. This happens because LTE aims to be a general-purpose network providing normal Internet connectivity, and the layering of stacks is more prominent in those networks.

#### 3.7.2 Cause: Accounting Policy Inconsistency

Mobile networks come with a variety of billing methods. Some services are charged by time and geographical distance, others by data volume. In earlier networks, the different billing methods were straightforward to distinguish as they were based on different network services. However, data networks—such as the Internet—were originally not in mind when earlier networks were built. Another problem are transmission artifacts that occur on lower layers without the knowledge or control of higher layers, such as data retransmissions because of bad connectivity or packet loss. For example, some providers charge for TCP retransmissions while others do not. In addition, some providers have special charging policies for extra services such as music streaming. These policy inconsistencies lead to hidden channels that can be exploited for billing attacks (fraud attacks).

##### 3.7.2.1 Attacks

Hidden channels for different protocols have been found, e. g., in the DNS protocol [241] or in TCP retransmissions [127, 128], both leading to under billing attacks. Additionally, TCP retransmission can also be exploited for over-billing attacks [127]. In this case, the attacker uses an existing connection to send unwanted TCP retransmissions to increase the victim's data usage. With the shift from the circuit voice to a packet-based voice switching, VoLTE introduced a new attack surface for under-billing attacks using the RTP and the SIP protocol [177, 190]. As voice is traditionally charged according to call duration, the voice-related channels can be misused as a hidden channel to transport data and thereby circumvent the accounting mechanism.

**Assessment** Most hidden channels are still exploited by an attacker with user traffic capabilities and optional Internet capabilities. Similar to the cross-layer information loss, these attacks are exploitable in the latest network generations and can be exploited everywhere in the network.

## 3.8 Root Cause: Wireless Channel

The wireless channel is essential for realizing mobility in mobile networks. However, this versatility makes the channel also easily accessible by unauthorized persons within the range of the radio transmission. Additionally, the wireless channel has limited resources. Over time more effective modulations and transmission schemes have been developed to improve the wireless transmission performance by reducing transmission redundancies. The easy access to the wireless channel makes mobile networks prone to jamming attacks for which an attacker disturbs the communication between two parties in a targeted manner. Jamming attacks are DoS attacks and require an active radio attacker. As a result, the wireless channel is prone to several attacks and exhibits fundamental limitations such that we define it as a root cause.

### 3.8.1 Attacks

Jamming attacks disturb the communication by increasing the noise on the wireless channel. Most prior research has concentrated on the evaluation of different constant jamming strategies and their effectiveness [52, 62, 168, 244, 329]. While constant jamming attacks jam the entire communication bandwidth over time, smart jamming attacks are protocol-aware and intentionally jam certain control information that affect the rest of the communication. In general, smart jamming attacks are more cost-efficient. Lichtman et al. [193, 194, 254] demonstrated that LTE is particularly vulnerable to smart jamming.

**Assessment** All jamming attacks require an active radio attacker who needs to be aware of the used frequencies and the bandwidth. For smart jamming attacks, the attacker requires knowledge of the protocol and needs to be synchronized with the cell to obtain the position of control information. Nevertheless, the hardware for such attacks is easily available [107, 169], in particular in the form of software defined radios such as USRPs [114]. While jamming attacks disturb the communication of all the victims, smart jamming attacks are more targeted. In all cases, the effective range of the attack is limited by the transmission power and location of the jammer. The motivation for jamming attacks is versatile. Besides simply obstructing the mobile service [44], jamming attacks can also serve as *downgrade attacks*.

## 3.9 Fake Base Stations Attacks in Detail

Fake Base Stations (also IMSI-Catchers, Rouge Cells, Cell Site Simulators, or Stingray) combine multiple of the above vulnerabilities and are also a tool to facilitate many more attacks. This includes almost all attacks from Table 3.1 with the "Radio Active" capability.

## 3.10 Capabilities of Fake Base Stations

In general, IMSI Catchers come in two variants: (i) a tracking or identifying IMSI Catcher and (ii) capturing or Man-in-the-Middle IMSI Catchers (see below). The first read out specific data from a phone or launch a specific attack before releasing the phone back into the genuine network. This is useful for enumerating phones in the vicinity or check for a specific device in radio range. The latter holds the phone captured in its fake cell and can relay traffic to the outside world.

While IMSI Catchers originally exploit a specific vulnerability in GSM networks, they are still a relevant threat in UMTS and LTE networks, for several reasons: First, the weakest-link principle applies. As long as users can be deliberately downgraded to a less secure system, the weakest link sets the limit. Additionally, it has been recently shown that IMSI Catchers are possible on 3G and 4G in either a tracking-only setup or for full traffic interception in combination with backbone attacks (SS7, Diameter). These protocols are often used for interconnection and roaming of phone calls, but also of cryptographic material such as keys. In the roaming case the remote network has to be able to fulfill the same cryptographic operations as the home network. Engel [111] also presented sole backbone attacks, but they are out of this paper's scope.

### 3.10.1 Access Technology

#### 3.10.1.1 2G/GSM

The original IMSI Catcher was build for GSM. Originally used only for identifying users (tracking), later devices allowed full man-in-the-middle attacks. GSM networks are specifically easy to impersonate, as the standard does not demand encryption nor support mutual authentication.

#### 3.10.1.2 3G/UMTS

Recent datasheets [124, 245] show (limited) UMTS and LTE capabilities of commercial available IMSI Catchers. For man-in-the-middle attacks they often downgrade users to 2G and capture them there. Osipov and Zaitsev [233] presented a de-facto 3G IMSI Catcher by using a reverse-engineered femtocell. They also discovered that contrary to the standard, many phones accept unauthenticated SMS messages or time synchronization.

#### 3.10.1.3 4G/LTE

Similar to UMTS, tracking IMSI Catchers are possible and phones tend to ignore integrity for some message types [207, 275] or allow authenticated commands in pre-authentication traffic (e.g., IMEI retrieval) [68, 264].



## **3.10.2 Radio Capabilities**

### **3.10.2.1 Frequency range**

All access technologies (above) are able to operate on multiple frequency bands, e.g., standard GSM is typically operated in the 850, 900, 1800 and/or 1900 Mhz range, depending on regulatory in a country. In an attempt to succeed all other access technologies LTE defines 44 different bands with flexible channel widths.

### **3.10.2.2 Number of channels to simulate**

The more channels an IMSI Catcher is able to simulate the faster it can acquire mobile phones in the vicinity and/or serve multiple frequency ranges or access technologies simultaneously.

### **3.10.2.3 Antenna**

A directed antenna can focus on a particular region, maybe even a single user. An omni-directional antenna is good for catching all users in vicinity. A directed antenna focuses the same radiation power to a much smaller segment, and therefore typically can achieve longer range.

### **3.10.2.4 Power**

The power additionally determines the range of the device.

## **3.10.3 Catching Capability**

### **3.10.3.1 Tracking or Identification Mode (Catch and Release)**

In this mode, the IMSI Catcher is luring phones into its fake cell, reading out IMSI and IMEI and pushing them back into the real network. For a target with known IMSI or IMEI this method can be used to check his/her presence in vicinity (omni-directional antenna) or position (directional antenna). When used with a directional antenna, this can also be used to (visually) correlate a person to his/her IMSI and IMEI (see Section 5.4).

### **3.10.3.2 Capturing or MITM mode (Catch and Hold)**

In this case the MS/UE is held in the cell and not pushed back into the real network. There exist several methods to decrypt, relay, and/or modify the traffic (see Section 5.5).

### **3.10.3.3 Passive Monitoring**

This mode can be used e.g., after a target has been identified. Since the attacker does not have control over the phone it can switch to different cells and Location/Tracking Areas anytime. It has to follow the target across different frequencies and cells.

### 3.10.4 Cryptographic Capabilities

On GSM an attacker can choose between several methods. The easiest one, is to downgrade the client side and the network side to A5/0 (i.e. no encryption). However, many networks started prohibiting clients using A5/0. This can be problematic if legacy clients do not support any encryption. The GSM export-grade cypher A5/2 has been broken by Goldberg et al. in 1999 [130] and phased out by GSMA (GSM Association) by 2006 [140]. Barkan et al. presented a realtime ciphertext-only attack on A5/2 [55] in 2008. However, the GSM standard cipher A5/1 is also not secure; a number on publications [55, 106, 145] showed severe weaknesses and later 2 TB rainbow tables for decryption within seconds became freely available [183]. Thus, we must assume [280], that reasonable new IMSI Catcher are able to decrypt A5/1 and A5/2. Recently, many operators implemented A5/3 – a backport of the KATSUMI based UMTS cipher – for which no practical attacks are known. However, only newer handsets support this mode (cf. Figure 5.3, and are easily downgrade-able by a fake cell (Section 3.10.5 below).

For UMTS and LTE encryption no practical cryptanalytic attacks are known, and mutual authentication is needed for (most) transactions. However, vulnerabilities in the SS7/Diameter exchange between providers allow the recovery of sessions keys [111, 228] and therefore either decrypting traffic or impersonating a network.

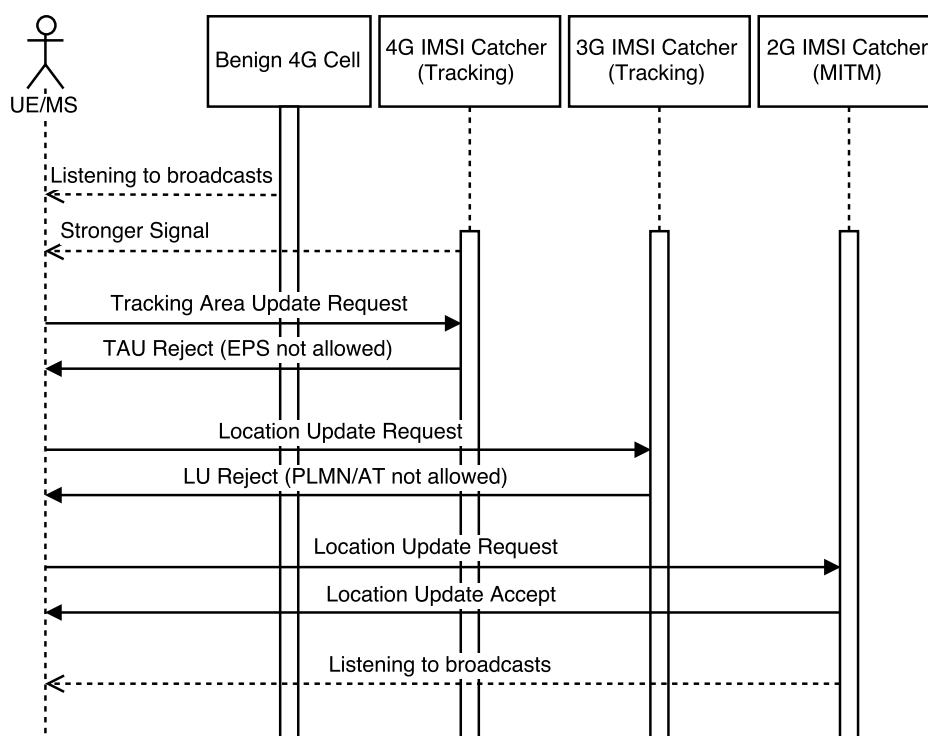


Figure 3.6: Downgrade attack from 4G to 2G using *Access Technology not allowed* messages (simplified)

### 3.10.5 Access Technology Downgrade Capability

For UMTS and LTE a downgrade to a less secure access technology (i.e., GSM) is an option. By moving the victim around different access technologies, an attacker can cherry pick attacks from different network generations.

#### 3.10.5.1 Jamming

A simple but brutal way is to jam the frequency band. In an attempt to restore connection to the network, the phone will try other (potentially less secure) access technology: e.g., jamming the UMTS band will encourage phones to connect via GSM. Longer jamming sessions will show up in the operator's network quality metrics and allow radio technicians to pin-point the source. Therefore, this method is most suitable for short term operations or confined spaces. In general, an attacker might strive for more subtle and less detectable ways. For details see 3.8.

#### 3.10.5.2 Spoofing No-authorization for a Specific Access Technology

A BTS, NodeB and eNodeB has the ability to deny access to a specific cell, location/tracking area or access technology for a number of reasons (e.g., no resources left, no subscription for a specific service, no authorization, etc.). Depending on the error code from the network, the phone will not retry and revert to other methods (e.g., another access technology) [30, 31, 139]. An error code for a permanent error will be cached by the MS/UE until next reboot or a lock-out timer runs out. 3GPP defined rules on how to allow a network operator to expel a mobile from one access technology e.g., for LTE [31, 275, c.f. reject cause #7] or 3G [139]. Therefore, a chain of tracking IMSI Catchers denying access and forcing a cell re-selection with another access technology can downgrade a client step by step (Figure 3.6). Once arrived at 2G/GSM without mutual authentication the attacker can capture the phone and hold it in the fake cell, by spoofing an empty or unusable neighbor cell list

These Location/Tracking Update Reject messages are intentionally not covered by the mutual authentication in UMTS and LTE, as a (foreign) network must be able to reject a user that has no subscription or no roaming agreement with the home network.

#### 3.10.5.3 Spoofing Radio Resource Allocation

In a similar – but less complex – approach to above, the fake 4G eNodeB can issue a radio resource control (RRC) message with the TAU-reject message asking the UE to switch channels [340]. As the RRC message contains a GSM channel (ARFCN) the phone will also switch access technology (and security model). The attacker is already waiting with their GSM IMSI Catcher there, giving them full MiTM capabilities. As above, the problem is the missing integrity and authentication on many signaling messages (Section 3.5.1).

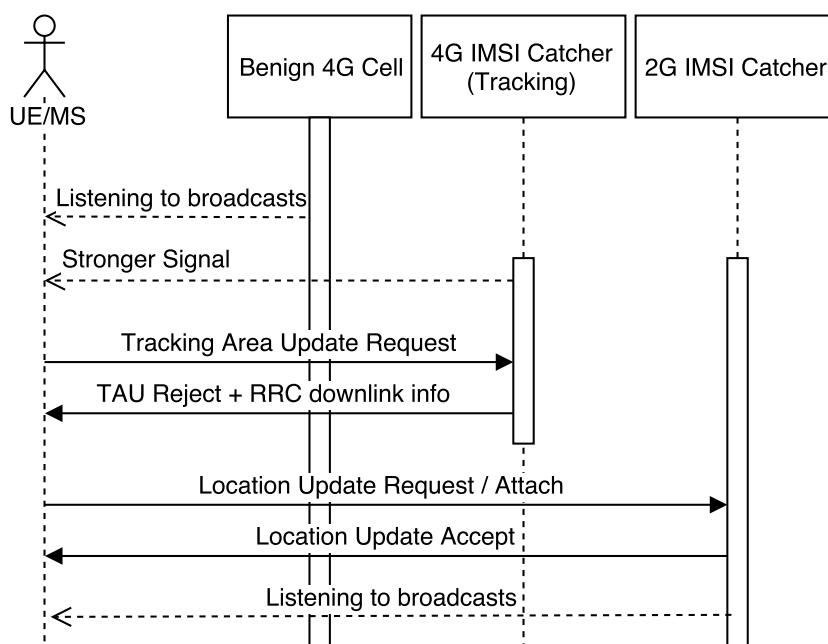


Figure 3.7: Downgrade attack from 4G to 2G using *Radio Resource Control* messages to reassign an GSM ARFCN to the UE

### 3.10.6 Traffic Forwarding and Provider Side Link

There are manifold possibilities to pipe traffic back to the network. Data (Internet) traffic can easily be NATed to any IP address. Mobile applications are built in mind that the Internet uplink changes frequently (e.g., switching WiFi networks, switching from WiFi to cellular service and vice versa) and therefore recover from an IP address change seemingly. However, Voice and SMS react very differently and need special treatment. Table 3.4 gives an overview.

#### 3.10.6.1 Downgrade Provider Uplink to no Encryption

In this mode the IMSI catcher signals upstream to the network and downstream to the captured device that the respective other side does not support encryption (so called A5/0 mode). It can then relay the unencrypted traffic in both directions. However, many networks began to phase out A5/0, even when this actually can brake some legacy devices.

#### 3.10.7 Forward Encrypted Traffic, Downgrade to Weak Encryption if Necessary

On GSM, all cipher suites except A5/3 onward (Table 3.3) are breakable, e.g., using rainbow tables. The cipher-capabilities of the network and the user device are exchanged in clear. Therefore, an attacker can easily downgrade the encryption setup to block A5/3

Table 3.4: Overview on forwarding techniques and its effects

Uplink Method	Access Technology	Voice		SMS		
		Outgoing CallerID	Incoming calls	Sending CallerID	Recv.	
A5/0 – no encryption	GSM	correct	Yes	correct	Yes	(1)
SS7/SIP uplink w/o CallerID restrictions	GSM	correct	No	depends	No	(2)
SS7 uplink with roaming capabilities	GSM, UMTS, LTE	correct	Yes	correct	Yes	(3)
Cracking Kc of target, simulating original device to Network	GSM	correct	Yes	correct	Yes	(4)
Using SS7 to get Session Key, pass messages unaltered	GSM, UMTS, LTE	correct	Yes	correct	Yes	
Other SIM card	GSM	wrong / hidden	No	wrong	No	
Consumer PSTN access (e.g. SIP)	GSM	wrong / hidden	No	wrong	No	

- (1) Some Networks started prohibiting A5/0 (2) Easy to get  
(4) incl. Passing messages 1:1 and cracking them afterwards (3) Hard to get

onward and force a downgrade to A5/1. (A5/2 is unlikely, as it has been officially phased out by GSMA [140].) Encrypted traffic is therefore simply relayed and can be decrypted separately or in real-time. The advantage over the passive monitoring attack is that the attacker remains in control over the mobile and can prohibit it from changing to another cell.

### 3.10.7.1 SS7/SIP Voice Uplink Without Caller-ID Restrictions

Many wholesale and trunk telephony providers offer SIP or SS7 protocol access without Caller-ID restrictions. In voice telecommunication networks the Caller-ID is set by the border provider and implicitly trusted by all others. Although harder to find, some of these providers allow SMS without Caller-ID restrictions as well. Although this works well for outgoing calls, receiving calls is not possible.

### 3.10.7.2 Relay with SS7 Key Retrieval

Direct SS7 interconnect access for the attacker allows retrieval of session keys, just as a roaming partner network would. Nohl [228] and Engel [111] demonstrated this and blamed the outdated security concept of the interconnect interfaces. They have been

designed in a time with a small number of international telecommunication companies which all trusted each other.

With the session keys, the attacker can decrypt and modify the data stream in both directions and the victim can also receive calls.

#### 3.10.7.3 Voice Forwarding via other SIM Card

This method is used by many mobile or portable IMSI Catchers such as in cars or helicopters. For voice the Caller-ID is typically suppressed. However, Caller-ID suppression is not supported for SMS. Most attackers might therefore choose to drop SMS completely. Incoming calls under the victims phone number can not be received.

#### 3.10.7.4 Voice Forwarding via PSTN

Other public switched telephone network (PSTN) access methods are imaginable, but come with the same restrictions as above.

#### 3.10.8 Victim-side Link

Just as described in Section 3.10.6, the client side (down) link can also be organized in different in a multiple ways. The artifacts introduced here are the main detection criteria for client side IMSI Catcher detection.

Just as on the provider side, the encryption can be downgraded to A5/0 or a weak cipher. Neither Android nor iOS will warn the user about this. Acquisition of session keys via SS7 allow capturing IMSI Catchers on 3G and 4G as well.

While such decrypting capabilities would allow for an complete passive attack, it has some disadvantages. In contrast to a complete passive attack, the attacker remains in control of the phone and can hold it hostage in its own cell by providing the phone with no (valid) neighbor cells. Otherwise, the attacker is at constant risk of loosing the phone to another cell and has to follow it (quickly).

### 3.11 Mobile Communications Acronyms

<b>3GPP</b> 3rd Generation Partnership Project	<b>EPC</b> Evolved Packet Core
<b>AKA</b> Authentication and Key Agreement	<b>ETSI</b> European Telecommunications Standards Institute
<b>AuC</b> Authentication Center	<b>GCM</b> Google Cloud Messaging
<b>DNS</b> Domain Name System	<b>GPRS</b> General Packet Radio Service
<b>DoS</b> Denial of Service	<b>GSM</b> Global System for Mobile Communications
<b>DPI</b> Deep Packet Inspection	<b>GUTI</b> Globaly Unique Temporary ID
<b>EDGE</b> Enhanced Data Rates for GSM Evolution	<b>HLR</b> Home Location Register
<b>eNodeB</b> Evolved NodeB	<b>HSS</b> Home Subscriber Server

<b>IMEI</b> International Mobile Station Equipment Identity	<b>QoS</b> Quality of Service
<b>IMSI</b> International Mobile Subscriber Identity	<b>RA</b> Routing Area
<b>IMS</b> IP Multimedia Subsystem	<b>RAN</b> Radio Access Network
<b>LA</b> Location Area	<b>RTP</b> Real-Time Transport Protocol
<b>LTE</b> Long Term Evolution	<b>SDR</b> Software Defined Radio
<b>MAC</b> Message Authentication Code	<b>SIP</b> Session Initiation Protocol
<b>MitM</b> Man-in-the-Middle	<b>SS7</b> Signalling System #7
<b>MS</b> Mobile Station	<b>TA</b> Tracking Area
<b>MSISDN</b> Mobile Station Integrated Services Digital Network Number	<b>TMSI</b> Temporary Mobile Subscriber Identity
<b>NAT</b> Network Address Translation	<b>TLS</b> Transport Layer Security
<b>OTA</b> Over-the-Air	<b>UE</b> User Equipment
<b>P-GW</b> Packet Data Network Gateway	<b>UMTS</b> Universal Mobile Telecommunications System
<b>PSTN</b> Public Switched Telephone Network	<b>USIM</b> Universal Subscriber Identity Module
<b>PKI</b> Public Key Infrastructure	<b>VoLTE</b> Voice over LTE





# Client-side Mobile Phone Network Attack Detection

*IMSI Catchers* are used in mobile networks to identify and eavesdrop on phones (see description in Section 3.9). In recent years, the number of vendors increased and prices dropped. Thus, the device became available to much larger audiences. Self-made devices based on open source software are available for about US\$ 1,500.

In this chapter, we identify and describe multiple methods of detecting artifacts in the mobile network produced by such devices. We present two independent novel implementations of an *IMSI Catcher Catcher* (ICC) to detect this threat to everyone's privacy. The first one employs a network of stationary (sICC) measurement units installed in a geographical area and continuously scanning all frequency bands for cell announcements and fingerprinting the cell network parameters. These rooftop-mounted devices can cover large areas. The second implementation is an application for standard consumer grade mobile phones (mICC), without the need to *root* or *jailbreak* them. Its core principle is based on geographical network topology correlation, facilitating the ubiquitous built-in GPS receiver in today's phones and a network cell capabilities fingerprinting technique. The latter works in the vicinity of the phone by first learning the cell landscape and then matching it against the learned data. We implemented and evaluated both solutions for digital self-defense and deployed several of the stationary units for a long-term field-test. Finally, we describe how to detect recently published large-scale denial of service attacks.

This chapter is an extended version of our Best Student Award winning ACSAC publication [91] with source code available [85].

The main contributions are structured as follows. A survey of network level artifacts caused by an *IMSI Catcher* are described in Section 4.4. In Section 4.5 we present a concept of a usable and customer grade warning system. Therefore, we determination

which detection methods are available and implementable with what consumer grade hardware in Section 4.6. We present our implementation and the evaluation of these methods in Section 4.7. Finally, we describe and evaluate the detectability of large scale denial of service attacks such as [132] in Section 4.9 before we summarize our findings in Section 4.11.

## 4.1 Preface

IMSI Catchers are MITM (man in the middle) devices for cellular networks [166]. Originally developed to steal IMSI (International Mobile Subscriber Identity) numbers from nearby phones (hence the name), later versions offered call- and message interception. Today, IMSI Catchers are used to track handsets, deliver geo-target spam [217], send operator messages that reconfigure the phone (e.g., installing a permanent MITM by setting a new APN, http-proxy, or attack the management interface [279]), directly attack SIM cards with encrypted SMS [227] that are filtered by most operators by now, and can potentially intercept mobile two-factor authentication schemes (mTAN). Pell and Soghoian [240] argue, that we are currently on the brink of age where almost everyone could eavesdrop phone calls, similar to the 1990s where cheap analog scanners were used to listen to mobile phones in the US and Europe.

In brief, these devices exploit the phone's behavior to prefer the strongest cell phone tower signal in vicinity to maximize the signal quality and minimize its own power consumption. Additionally, on GSM networks (2G), only the phone (via the SIM, Subscriber Identification Module) needs to authenticate to the network but not vice versa and therefore can easily be deluded to disable content data encryption. This enables an attacker to answer a phone's requests as if the phone was communicating with a legitimate cell phone network.

In contrast, the Universal Mobile Telecommunication System (UMTS, 3G) requires mutual two-way authentication, but can be circumvented using the GSM compatibility layer present in most networks [206], or mobiles can be forced to downgrade to a 2G connection by other means. Additionally, network operators use GSM as a fallback network where UMTS is not available. This makes GSM security still relevant and important in today's mobile network world.

## 4.2 History and Motivation

The first IMSI Catchers date back as early as 1993 [260] and were big, heavy, and expensive. Only a few manufacturers existed and the economic barrier limited the device's use mostly to governmental agencies. However, in recent years, a number of smaller and cheaper as well as self-built projects appeared making cellular network snooping attacks feasible to much larger audiences.

Chris Paget built an IMSI Catcher for about US\$1,500 [78] and presented it at DEFCON 2010. His setup consists of a *Software Defined Radio* [114] and free open source software such as GNU Radio, OpenBTS, and Asterisk. Several other (academic) projects also built such devices [247, 322] based on similar setups. Appropriate patches and configuration guides are publicly available.

In 2010, Nohl and Manaut [226, 231] presented practical snooping attacks on GSM's main cipher suite using custom firmware on modified mobile phones. However, such a solution can only monitor a very small number of frequencies at once and is likely to lose the intercepted phone on handovers to other cells. Therefore, a professional attacker will still use IMSI Catcher-like functionality to lock the radio channel.

As IMSI Catchers perform an active radio attack, we put forward multiple passive ways to detect such an attack, both stationary and mobile. We facilitated ordinary mobile phones or easily acquirable hardware. This allows for easy deployment of the described techniques for end users or interested hobbyists. We therefore intentionally chose to exclude expensive protocol analyzers or complex self-built solutions.

## 4.3 Background

For a high-level description of cellular mobile phone networks, please consult Section 3.2.2.

In GSM, a cell is uniquely identified by the mobile country code (MCC), network code (MNC), location area code (LAC) and the cell ID (CI). The neighbor list typically includes additional per cell attributes like the frequency (ARFCN) and channel quality metrics. Given that UMTS networks are organized differently, LAC and CI are replaced by PSC (primary scrambling code) and CPI (Cell Parameter ID). For the sake of simplicity, we will call any tuple that uniquely identifies a network cell a *Global Cell ID* or *Cell ID* for short.

IMSI Catchers blend into the mobile network operator's infrastructure impersonating a valid cell tower and therefore attracting nearby phones to register to it. For above (Section 3.2.10) discussed reasons, a phone prefers cells with higher signal levels. Thus allowing an attacker to hijack phones from the genuine network.

Two main operating modes can be distinguished.

**Identification Mode, Catch-and-Release** As a phone is lured into the fake cell, the worldwide unique identifiers such as IMSI and IMEI are retrieved and the phone is sent back to its original network via denying its original *Location Update Request* with an *Location Update Reject*-Message. This procedure typically takes less than two seconds, whereas attracting the phone can take minutes. No other information besides the identification numbers is retrieved.

A law enforcement agency can then apply for a warrant<sup>1</sup> and access the call- and meta information of a subject via the mobile network operator. This considerably saves the agency working hours, as no one has to operate the IMSI Catcher over the whole period of observation and follow the subject in its every move.

Other attackers can use this mode for user tracking purposes or to lookup the exact phone model based on the IMEI to better tailor future attacks.

**Camping Mode, Tracking Mode, Catch-and-Hold** The phone is held in the cell of the IMSI Catcher and content data is collected. Traffic is forwarded to the genuine network so that the victim stays unaware of the situation.

IMSI Catcher users that do not have time for for a warrant or can't acquire a warrant (e.g., because they operate outside the law) use this method. It will also gain importance as A5/3 and A5/4 are introduced into GSM networks, making passive snooping attacks on the broken A5/1 and A5/2 ciphers useless. In UMTS networks, phones are additionally downgraded to GSM and its less secure ciphers.

## 4.4 IMSI Catcher Artifacts and Detectability

An IMSI Catcher has many detail problems to overcome; the respective solutions will typically introduce irregularities in the network layer that leave hints for an educated observer. Due to the secret nature of the operation of these devices, not much information is available. Nevertheless, we generated the list below based on the material available and our own research. Some of the traits can be mitigated but most are of structural nature. However, not every IMSI Catcher will produce all of the artifacts described below.

### 4.4.1 Choosing a Frequency

To increase signal quality, avoid radio interference, and thus trigger the mobile provider's own radio quality monitoring system, an attacker has to use an unused frequency (i.e. ARFCN, Absolute Radio Frequency Channel Number) for its IMSI Catcher. A relatively safe choice for a frequency are unallocated radio channels (e.g., *guard channels* between different operators or reserved channels for testing). However, it is less likely to lure a mobile phone onto this channel, as the phone (MS) will preferably only look on the advertised neighbor frequencies. Another method is to use an advertised frequency that is actually not being used or is not receivable in the specific geographical area under attack.

**Detectability:** Off-band frequency usage can be detected using a current frequency band plan as assigned by the local authorities. Radio regulatory bodies and frequency plans are available for almost all countries.

---

<sup>1</sup>May vary depending on the legislative system; in the U.S. also called *pen trap*

### 4.4.2 Choosing a Cell ID

Typically, an attacker will introduce a new cell ID (preferable including a new LAC) previously unused in the specific geographical region for two reasons: First, to not provoke an accidental protocol mismatch when the MS should receive the corresponding genuine BS by accident. Secondly, to provoke a *Location Update Request*<sup>2</sup> from the phone to be able to lure it in the fake cell.

**Detectability:** Our data shows, that cell IDs are very static. Many mobile operating systems use them together with Cell ID databases to coarsely estimate the phone's location where either GPS is unavailable, rough estimations are detailed enough, or to aid the GPS receiver during initialization. Using such a database and correlating its information with the real geographic location could reveal unusual cell IDs and frequency usage in a specific area.

### 4.4.3 Base Station Capabilities Fingerprinting

Each beacon signal of a base station is accompanied by a list of supported features (e.g., packet radio services such as GPRS or EDGE). If the attacker does not copy the capabilities of the original network precisely, the simulated cell will not provide all services like the original network. For example GPRS and EDGE are services that need very complex emulation layers as they use a different modulation but share time slots with the rest of GSM. We do not expect many IMSI Catchers to support these protocols.

**Detectability:** A MS should denote such capabilities in the above Cell ID database (or a local one) and use them to find suspicious base stations not matching their previously known capabilities. Cell capabilities change very rarely, and if so, the network operator usually upgrades to new systems (e.g., GPRS to EDGE, HSDPA to HSUPA), but not vice versa.

### 4.4.4 Network Parameter Fingerprinting

Another information conveyed by the beacon signals to the mobile station are basic network parameters about the organization of the mobile network such as time slot organization, threshold values and timeout values. While they can differ from base station to base station, our research has shown that most of them tend to be uniform across a given network operator but vary between different operators. A IMSI Catcher operator might not always copy all of these parameters as they are not operationally important for an attack. Detection possible as described above (Section 4.4.3).

---

<sup>2</sup>A low *T3212 Periodic Location Update Timer* is another technique, but the smallest possible value is 6 minutes.

#### 4.4.5 Forcing a MS to Register

Despite providing the better signal and simply waiting for a victim to voluntarily switch cells, an attacker can actively step in. An easy way to force a victim's device to disconnect from the original network and register to a new (possible) fraudulent base station (as provided by the IMSI Catcher) is an RF jammer. After a fruitless scan of the advertised neighbor frequencies the phone eventually falls back to a full scan, therefore giving the IMSI Catcher the opportunity to attract the phone.

Several companies [34, 124] offer systems for targeted jamming of a specific phone.

**Detectability:** Jamming can be detected by a MS by watching channel noise levels (e.g., from the neighbor list).

#### 4.4.6 Handling UMTS Clients

One possible way is to downgrade an UMTS capable MS to the less secure GSM network by rendering UMTS channels useless with an RF jammer (as above). Meyer and Wetzel [206] presented another way: a MITM attack for UMTS networks which facilitates its GSM compatibility layer. This layer is present in most deployed UMTS networks, as they use GSM for backward compatibility and to increase the coverage. Additionally, some companies [34, 124] claim, their equipment can transfer single targets from UMTS to GSM.

**Detectability:** Jamming can be detected as described above. A cell database can be used to spot unclaimed GSM usage where UMTS should be typically available.

#### 4.4.7 Encryption

Older IMSI Catchers are likely to disable encryption (set cipher mode A5/0) in order to ease monitoring. However, current state-of-the-art attacks on GSM A5/1 and A5/2 cipher allow for a timely decryption and key recovery. Weaknesses found in the A5/2 cipher [106] have led to its abolition by the GSM Association in 2006 [140]. However, the stronger variant A5/1 is also prone to precomputation attacks using rainbow tables. These are publicly available [183] and allow computers with a 2 TB hard disc and 2 GB RAM to recover the key in about two minutes. While this makes completely passive eavesdropping on phone calls possible, phones can easily *get lost* by handing over to another cell (see next section). Furthermore, the newly introduced and currently rolled out [274] A5/3 and A5/4 ciphers (backported from UMTS) will force attackers back to active interception with IMSI Catchers to downgrade the encryption used. Known attacks on A5/3 are not yet feasible [64, 103, 182].

**Detectability:** The absence of a cipher alone is not a sufficient indicator: encryption might be unavailable in foreign roaming networks. However, once a phone had an

Table 4.1: IMSI Catcher detection matrix

IMSI Catcher Artifact	Detection Method	Android API	iOS API <sup>‡</sup>	Telit [288]
Unusual Cell ID	Cell database	serving cell & neighbors <sup>†</sup>	serving cell	yes
Unusual cell location		yes	yes	no
Unusual frequency usage		no	no	yes, ARFCN
Short living cells		yes	limited	yes
Unusual cell capabilities		serving cell & neighbors <sup>†</sup>	indirect	scan, neighbor
Guard channel usage	Band plan	no	no	yes
Network parameters	Network fingerprinting	no	no	limited (GPRS)
RF jamming	Watching noise levels	limited	no	yes
Disabled cipher	Read cipher indicator	expected in future [17]	no	no
Neighbor list manipulation	Cell DB & sanity check	limited <sup>†</sup>	no	limited
Receive gain	sanity check	no	no	no
Missing caller ID, SMS	Periodic test calls	yes	yes	yes

<sup>†</sup> Neighbor cells available via standard API, but not implemented in all phones.

<sup>‡</sup> Only via iOS private API. See Section 4.6.2 on reasons why iOS is not considered in this work.

encrypted session with a particular network and particular SIM card, it should assume that a sudden absence of any encryption is an alarming signal.

#### 4.4.8 Cell Imprisonment

Once an attacker *caught* a phone, she/he will try to lock it in so it does not switch to another active cell. Therefore, it will either transmit an empty neighbor list to the phone or a list with solely unavailable neighbors. The base station can also manipulate the *receive gain* value [78]. This value is added to the actually measured signal levels by the MS to prefer a specific cell over another (hysteresis).

**Detectability:** A mobile station monitoring its neighbor list (e.g., together with a geographical database) is able to find such suspicious modifications.

#### 4.4.9 Traffic Forwarding

The attacker needs to forward the calls, data and SMS to the public telephone system. There are multiple ways to achieve this. The simplest solution is to use another SIM card and a MS to relay calls into the mobile network. However, from the networks point of view these calls will be made under another identity. The attacker will most likely disable caller ID presentation to not immediately alarm the recipient. In this setup, the IMSI Catcher will not be able to handle any incoming calls for the surveyed station or any SMS.

Another setup could route these calls directly into a SS7 phone exchange network. Telecom operators usually trust their wholesale- and exchange partners with provider grade connections to set legitimate caller IDs. An attacker with access to such an interface could also spoof caller ID for outgoing phone calls and text messages. However, it is

unlikely that the attacker can also manipulate the routing of incoming calls. See Section 3.10.6 and Table 3.4 for more details.

A third setup option (a full MITM attack) could facilitate a more advanced GSM frame relaying setup where data is handed over to the original network as if it were sent by the victim's phone.

**Detectability:** The first setup is detectable by making test calls and independently checking the caller ID (e.g., using an automated system).

#### 4.4.10 Usage Pattern

IMSI Catcher in *identification mode* are operated for rather short periods of time to locate and verify an unknown phone such as prepaid phones or phones in a particular area. For tracking purposes and for eavesdropping the fake cell is active for the whole duration of the surveillance. Both operating times are considerably lower, than the average lifetime of a genuine cell.

**Detectability:** Cells that suddenly appear (with good signal quality) for a short period of time and cease to exist afterwards.

### 4.5 Catching an IMSI Catcher

Simple, cheap, and easily deployable *IMSI Catcher Catchers* (ICC) either need to run directly on a user's mobile phone or on affordable hardware (e.g., stationary device). While both concepts can be used to document IMSI Catcher use in a specific area, the former is also able to warn its user directly. In this section we describe both concepts, before we present our implementation in Section 4.6.

As Table 4.1 summarizes, the main detection method consists of a cell ID database. Commercial as well as free database projects exist. Most of them provide an online interface to their data. However, they neither guarantee to be complete nor correct, partly due to their crowdsourcing nature. Also, they lack additional attributes needed for fingerprinting cell capabilities. Therefore, a *IMSI Catcher Catcher* (regardless if it is a mobile app or a dedicated stationary device) needs to be able to collect and maintain its own database regardless of any external databases (even when it is initially fed from another source). Furthermore, a mobile app can not assume online access is possible while being under attack.

Both types constantly collect all the data available about nearby cells. The mobile solution facilitates the almost ubiquitously built-in GPS receiver available in smart phones to correlate the data with its location. Therefore, from the phone's perspective the network topography is revealed similarly to *explorable maps* known from computer games, where the user only sees the areas of the map which he visited before (*Fog of*



*War*). Visiting an already known area allows comparison of the current results with the stored data.

Additional tests include monitoring the noise levels of channels (RF jammer detection), network- and cell capabilities (e.g., cipher and GPRS availability), and sanity checks of network parameters (e.g., empty neighbor list might indicate a cell imprisoned phone). A caller ID test is implementable using an automated query system. However, regular calls to that system might result in non-negligible costs and have to be cryptographically authenticated.

The mobile app user (mICC) interface can be simplified to a user friendly four stage indicator:

**Green** No indicators of an IMSI Catcher attack found. Previously collected data matches the current network topography and all other tests completed negative.

**Yellow** Some indicators or tests show anomalies. However, these hints are not sufficient to postulate an IMSI Catcher attack. The user should avoid critical details in calls.

**Red** Indicators strongly suggest an IMSI Catcher attack or some other major network anomaly.

**Grey** Not enough data available (e.g., the user is in a previously unknown area).

An application with more intrusive access to the baseband might limit the phone's use to trusted cells only.

In contrast, a dedicated stationary IMSI Catcher Catcher (sICC) placed at a favorable position with a good antenna might receive a far greater radio cell neighborhood and allow to monitor a greater area non-stop (Figure 4.8). This is of great advantage when searching for a potentially transient event like the rather rare and short usage of an IMSI Catcher. Multiple devices can form a sensor network monitoring e.g., a whole city. As they don't move around, a GPS receiver is unnecessary. Most tests compare the collected data with the stations own history.

## 4.6 Implementation

Implementation poses some additional challenges: Only very limited baseband information is available to high level applications. In mobile operating systems, low level access is prohibited. System- and root applications can have access but are then limited to a very specific phone model (or chipset). This requires a *rooted* or *jail broke* phone. Additionally, only information is available that the chipset manufacturer has chosen to be disclosed. This also applies to commercial or industrial GSM/UMTS modules.

Among other baseband information, the neighbor cell list is an infamous example. Device support varies vastly, even for products of the same manufacturer. There is no identifiable

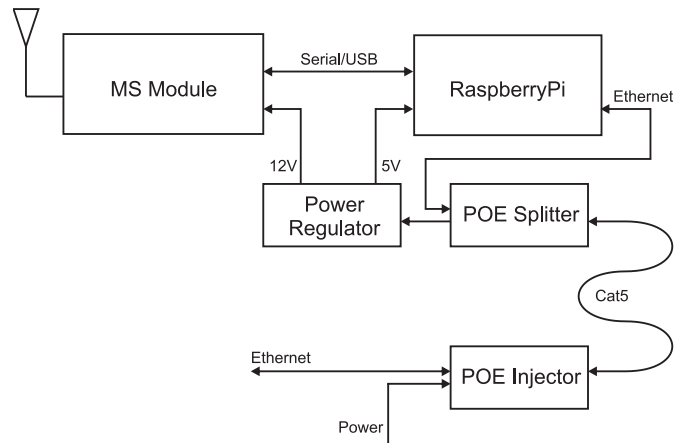


Figure 4.1: Dedicated device:  
block diagram

pattern between low-end and high-end or older and newer products. Baseband information used to be called *engineering-*, *field test-*, or *network monitor* functionality for a long time. However, a few years ago, access to information such as the *serving cell* or *neighbor cell list* became popular for (coarse) locating devices in combination with a geolocation cell ID database, where GPS is not available, a loose estimation is detailed enough, or to simply aid the GPS during initialization. Therefore, recent smart phone operating systems provide a direct or indirect API interface to this information - even when it is unreliable in some cases.

When available, the next challenge is just around the corner: A MS is not required to keep a list longer than six nearby cells. Thus, the neighbor list provides only a very limited geographical view into the nearby network structure of the currently selected operator, despite some potentially more receivable cells. This is especially true in very dense networks such as in urban centers.

To extend the view and collect more data than the neighbor list length, a MS could be switched to use just a specific network band, such as 900 or 1800 Mhz GSM band or the 2100 Mhz UMTS band (many older phones and some data modules allow for this). Collecting disjunctive neighbor cell information for all bands separately extends the view on the network. Additionally, a device with a foreign SIM might be able to register at multiple (roaming) networks to investigate each one separately. However, both techniques interfere with the normal operation of a hand set. A mobile device constantly performing these kinds of investigations is not able to provide services for the end user in commonly expected quality. It would require a dedicated device for such measurements.

#### 4.6.1 GSM Modems and Modules

For the dedicated stationary type of the IMSI Catcher Catcher (sICC) we tested several USB modems from ZTE, Nokia, and Huawei as well as MiniPCI modems from Qualcomm, none of which supported neighbor cell listing. Nokia and Huawei seem to support it on

older devices, but dropped support on more recent ones.

Additionally, we started to test industrial modems such as devices from Telit. Among others, the Telit GT864 allow network registration and neighbor list scanning even without an inserted SIM card, allowing to scan each network in a region on each frequency band separately (see above). This provides a much greater view on the network structure than a simple mobile phone can provide.

On top of it, many Telit modems implement a cell beacon monitoring mode [288] that can be easily facilitated into a frequency band sweep cell beacon scan. Thus, allowing a complete view over the receivable network cells by frequency including their ID, some capabilities, signal, and noise levels. The latter also allows a simple jamming detection.

**Our Implementation** Our dedicated stationary setup (Figure 4.2) consists of a Telit GT864 [287] and a Raspberry Pi embedded Linux computer. Internet up-link (to collect the captured data) is either provided by an Ethernet network, power LAN, via WIFI (USB-Dongle), or an UMTS modem. Data is collected locally in an sqlite3 database and periodically uploaded to a central server. The whole setup including mounting material costs less than €200. As the device is able to perform full frequency scans for all providers without the limitation of length-limited neighbor cell lists, we placed these devices on rooftops to extend their range.

As of August 2014, the network consists of four devices, the first one went online in July 2013. Our sICC is able to sweep through the whole 900 and 1800 Mhz GSM and EGSM bands within five to seven minutes. Besides the Cell ID, its main and auxiliary ARFCNs,

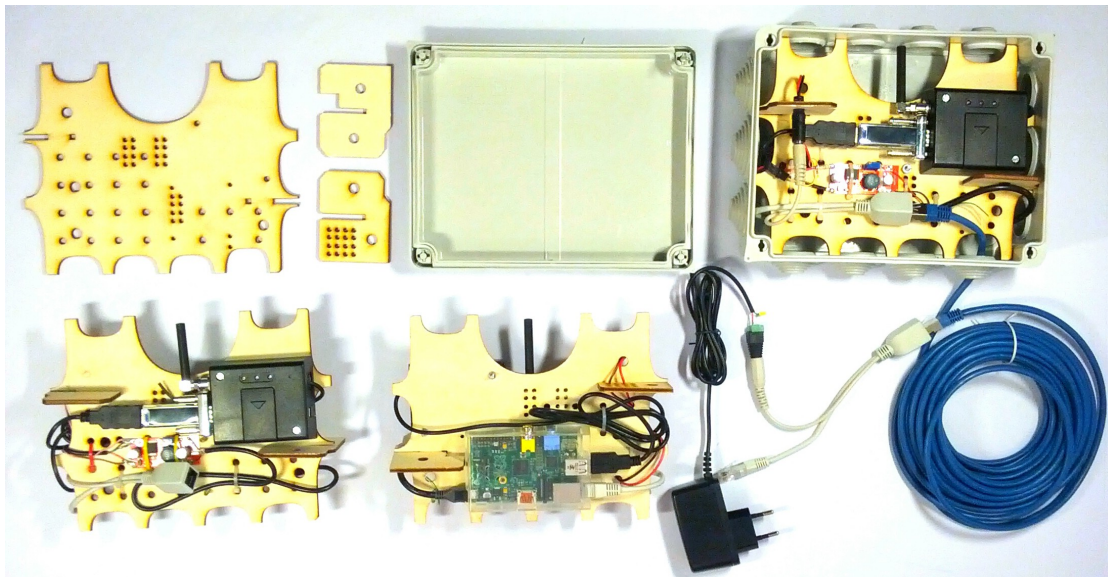


Figure 4.2: Construction of the dedicated stationary unit, using a laser-cut carrier (front and back)

it also records its receive levels and bit error rates as well as several GPRS configuration parameters (t3168 and t3192 timeouts, routing area codes, GPRS paging modes, etc).

### 4.6.2 iOS

iOS neither exposes high-level nor low-level baseband information (e.g., cell info) to applications through the official and public API. Methods such as `_CTServerConnectionCellMonitorGetCellInfo()` are available through a *private API*, whose documentation has leaked to the web. A field test App is available since iOS 5.1 by dialing `*3001#12345#*`. While the OS does not prevent the private API usage, it has been reported to be an immediate exclusion reason from the Apple App store. Applications using this API are only available to phones with a developer license or jail-broken phones and are therefore not of great use for a broader public.

Without a chance for widespread usage, we excluded iOS phones from further consideration.

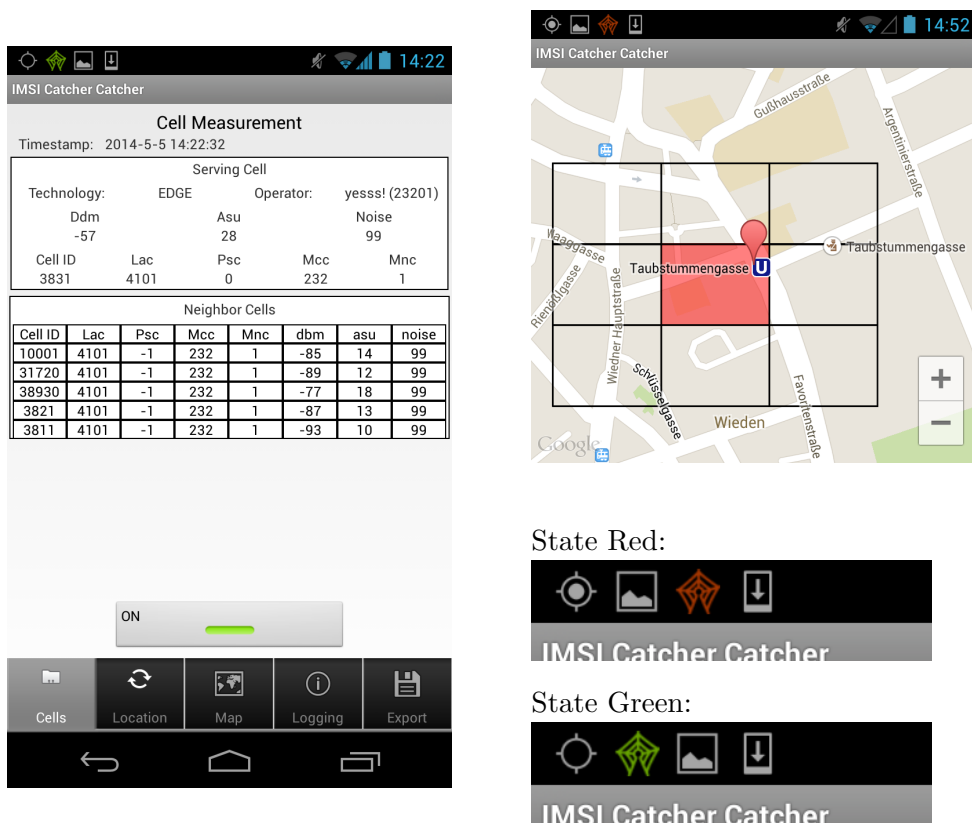


Figure 4.3: Screenshots of the mICC

### 4.6.3 Android OS

Android is a little more generous in providing access to baseband information. The `TelephonyManager` defines access to the important, but not all values on the wish list for the IMSI Catcher Catcher. Some values, such as the *cipher indication*, have been requested years ago and only recently got assigned for implementation [17].

The neighbor cell list problem described above also continues in the Android universe: The API defines the `TelephonyManager.getNeighboringCellInfo()` method. However, not even the long-time lead device *Google Galaxy Nexus* supports this method. Other devices only return meaningful values for this call for GSM type of networks, but not UMTS. It is not always clear if the underlying chipset does not provide this information or if the high level API lacks implementation by the phone manufacturer. A survey by the authors of the G-NetTrack application [126] reveals that this functionality is supported by less than half of the tested devices. Most devices report data only for the current serving cell. Recent devices have higher chances of implementing this method, most notably the *Google LG Nexus 4* and *Google LG Nexus 5*.

In contrast, Samsung Galaxy S2 and S3 expose many parameters unavailable through the standard API (such as the cipher mode [17]) via a Service Mode Application [118]. Some HTC devices offer similar hidden *Field Test Applications* [258]. These applications run under elevated privileges and often directly communicate with the baseband chipset via an operating system level device. Copying their interface will limit the application use to a rooted phone of a very specific model.

The absence of a neighbor list feature does not make a mobile IMSI Catcher Catcher (mICC) application impossible, but much less effective. This especially affects the speed of the network structure learning phase and some sanity checks on the network structure (e.g., cell lock-in by not having any neighbors). Another value offered by the API but not implemented in all phones is the noise level.

**Our Implementation** In favor of keeping our implementation [85] root permission free, we intentionally renounce the use of low-level information. While this provides less details it enlarges the potential user base.

A background service collects GPS position and cell related data (serving cell, neighbor cell, supported packet data modes). Measurements are triggered by the `PhoneStateListener.onCellInfoChanged()` - Callback and a regular 10-second timer (whichever comes first). This way, brief redirection to and from a cell (Section 4.3, Identification Mode) can be detected. For the sake of simplicity we group measurements in rectangular geographical tiles of about  $150 \times 100$  meters and store them in an `sqlite3` database. Some tiles might be in the learning phase while others are used for evaluation at the same time. We consider a tile fit for evaluation if the user collected cell data in this cell and all of its 8-connected tile neighborhood. Otherwise, nearby cells might easily create false alarms. A cell is considered valid for a given tile, if it was received as serving- or neighbor cell in one of the 9 tiles.

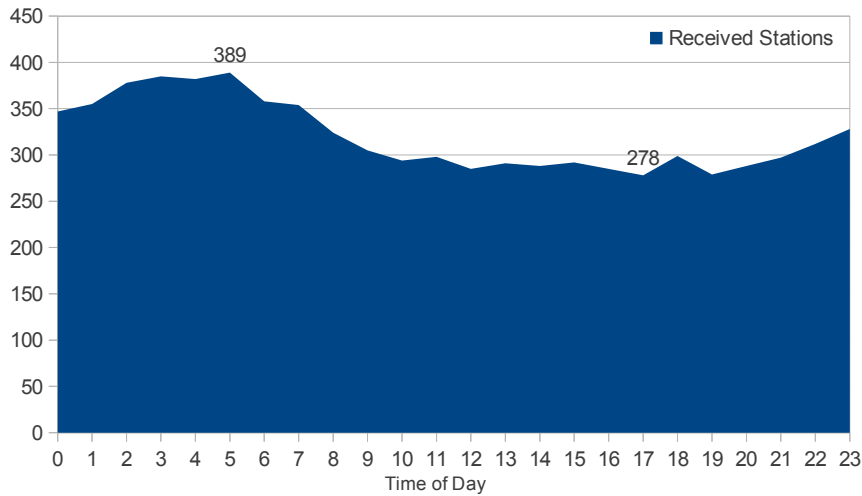


Figure 4.4: Maximum number of unique distinct cells received throughout the day (sICC)

The app also runs in the background and displays the current evaluation result in the notification bar, so that it is visible in the system dialer and phone application.

## 4.7 Results and Discussion

The evaluations goal is to answer two main questions: (1) Are the two IMSI Catcher Catcher able to detect the presence of an IMSI Catcher? (2) Are IMSI Catchers used in our vicinity?

We evaluated both systems with lab tests as well as field tests. For our lab tests we used an USRP1 based IMSI Catcher running OpenBTS 2.6 in identification mode. Therefore, we patched OpenBTS to download the IMEI and IMSI of any phone and then reject the Location Update request - pushing the phone back into the genuine network based on [247]. Because of their very brief interaction with the phone, such IMSI Catchers are particularly hard to detect. Experiments were concluded in an controlled environment to not interfere with outside phones.

### 4.7.1 Stationary IMSI Catcher Catcher

In the lab experiment, the sICC was able to detect the new fake cell based on its cell id, parameters and capabilities.

For the field test, our first sICC was installed on a rooftop in Viennese city center in July 2013. Three additional stations have been installed in the first months of 2014. We collected over 40 million datasets. The range of some installations is remarkable: Under rare conditions (*Inversions*) we receive single stations up to 90 km away. Radio

conditions vary among the day and so does the number of received cells (Figure 4.4). A map based Google’s geolocation database is shown in Figure 4.8. This external database is only used for visualization purposes and is not required for detection.

Regarding fingerprinting of cell parameters, we found many useful parameters<sup>3</sup>. In our test set of Austrian A1, T-Mobile, Orange/H3G, and Slovak O2 Telefonica network they all have the same value on all cells within a network, but distinct values between operators. Other values<sup>4</sup> displayed two distinct values within the Orange/H3G network.

CellIDs are very stable regarding their used ARFCN. However, on very received cells, one ARFCN can seem to have alternating different CellIDs. This can happen in situations, where the receiver sits in between two distant cells that are both using the same channel.

As Figure 4.5 shows, most cells remained static throughout the entire collection time. We attribute the bulk of very short-living cells to the following two effects: First, exceptional but transient weather and RF conditions that allowed the reception of cells very far away - often from foreign networks (Slovakia, Hungary, Czech Republic). We attribute this to *tropospheric scattering* and *ducting* caused by inversions [138, p.44] (see Section 4.10.1). These cell receptions are typically in the GSM 900 band and recorded as having very low signal levels and high bit error rates.

Second, we noticed a bigger cell reorganization at one of the operators (*A1 Telekom Austria AG*) in the night from November 16<sup>th</sup> 2013. During a period of several hours, many cells appeared for only a brief period of time. We have not yet received any explanation from the operator. Also in November 2013, Orange/H3G received previously unassigned frequencies in the GSM 1800 band.

We found two additional irregularities in our collected data: (1) Some cells seemed to operate outside the official assigned frequency ranges. A request at the Austrian Regulatory Authority for Broadcasting and Telecommunication (RTR) revealed an error on their side in the published frequency band plan. This was later corrected [50]. (2) We received a cell with a valid looking Austrian MNC, LAC, and CI, but an unassigned network country code (NCC). We speculate that this could be either a transmission error or a base station in maintenance or test mode.

Under certain conditions it can make sense for an IMSI Catcher to emulate a foreign network to catch a roaming handset. However, in our case we are receiving different stations during nighttime over a span of multiple months. We therefore do not think these symptoms fits an IMSI Catcher and attribute them to natural effects (Section 4.10.1).

#### 4.7.2 Mobile IMSI Catcher Catcher

For the prototype app we required at least 30 measurements and two re-entries into each map tile, before it finished the learning state. Additionally, the whole 8-neighborhood of

<sup>3</sup>PBCCH existence, SPGC, PAT, t3168, drmax, ctrlAck, alpha and pcMeasCh

<sup>4</sup>NMO and bsCVmax

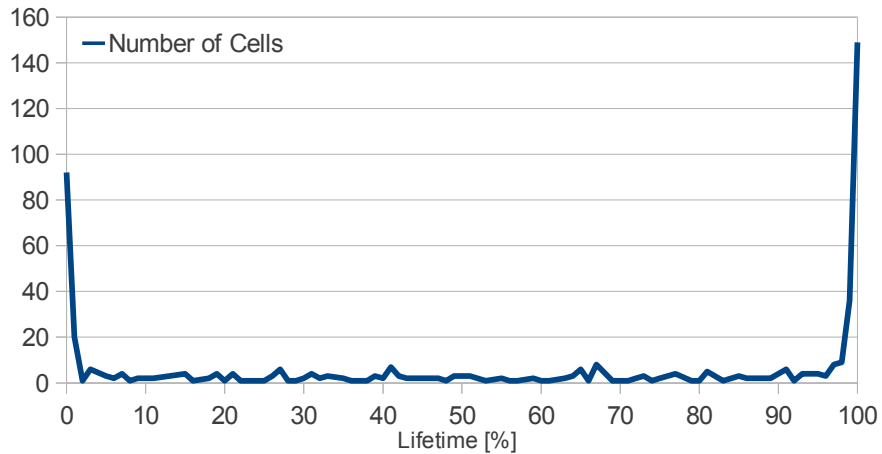


Figure 4.5: Cell ID lifetime throughout the experiment

the current tile must finish learning before it is considered for evaluation. The map view of our app supports the user in coloring tiles based on needed data. An always visible color coded icon in the notification bar indicates the warning level (Figure 4.3).

In our lab experiment, we were able to detect new and short living cells reliably, even when the *Location Update* was immediately rejected by our IMSI Catcher. Subtle differences exist in the implementation of the baseband to Android API interface. Some models report the new CellID and LAC for the ongoing but not completed cell change. Others only update the CellID immediately, while the LAC remains unchanged until the new base station accepts the *Location Update* request (e.g., Nexus 4).

For our biggest field test we chose a notoriously violent event in Vienna: a politically disputed ball taking place in the city center, and its counter-demonstrations. We

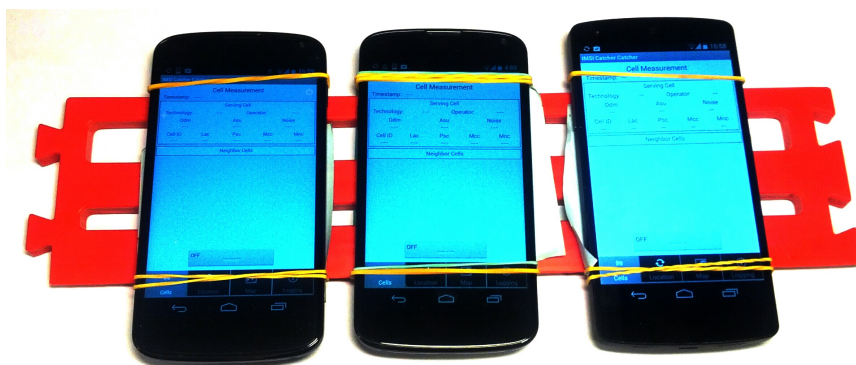


Figure 4.6: Field test for all three GSM networks



anticipated that the authorities could use an IMSI Catcher to identify rowdies as suggested by media reports. We assembled a battery of three phones (Figure 4.6) for all three disjunctive GSM networks in Austria. We visited the demonstration route the day before and then attended the demonstration undercover. However, we could not find any indicators of an IMSI Catcher.

### 4.7.3 Limitations

Our geographical network topology correlation approach and the cell database in general assumes a rather static mobile network structure, as every change will be flagged as suspicious. In fact, network structure is very steady and this is actively utilized by mobile operating systems for coarse self-localization and commercial suppliers of geographical cell databases.

There are corner cases where the mobile IMSI Catcher needs refinement. One such case are tunnels and underground trains. In Vienna, the public metro enjoys an almost flawless GSM and UMTS coverage. However, without GPS reception these underground cells often get associated with the place of entrance into the underground structure, as the phone's GPS receiver needs some time to detect its failure.

Another problem are holes in the tile map. If a tile is entirely located within an inaccessible area (e.g., a large private property), the 8-connection neighborhood rule forces all nine cells to never advance from the learning state into the evaluation state. This could be mitigated by a hole filling algorithm (e.g., an interpolation). Additionally, setting appropriate warning thresholds needs extensive real world testing.

## 4.8 Related Work

The osmocomBB Project [234] offers some IMSI Catcher indicators in their custom baseband firmware including cell fingerprinting and cipher indication. However, the project's target hardware platform are Texas Instruments' *Calypso* chipset based phones such as the (outdated) Motorola C123 or V171. This series of handsets went out of production in the mid-2000s. Considering the fast production cycles and the non-disclosure policies in the mobile phone industry it is unlikely that such open source projects will develop similar custom firmware for recent phones any time soon.

Another tool by Hummel and Neumann [157] works on a PC using an USB connection to a phone with an *Intel/Infineon X-Gold* baseband processor (Samsung Galaxy S2 and S3, but not S4).

Vallina-Rodriguez et al. [308] also faced the problem of acquiring internal baseband values and decided to require root privileges.

Unlike previous works, our approach works by recording the geographical topography of a mobile network and is therefore able to detect structural changes that an active IMSI Catcher will cause. It facilitates the almost ubiquitously built-in Global Positioning

System (GPS) receiver in smart phones. By using only standard API without any special permissions it ensures compatibility with as many phones as possible and is fit for public use. Some similar approaches for Android are also employed by the AIMSICD-Project [273]. Other projects surfaced after our initial paper [91], such as SR Labs' Snoopsnitch [281] or commercial projects such as by GSMK [142].

## 4.9 Exposing Large Scale Denial of Service Attacks

In future (see also Section 4.10) we like to read GSM, UMTS, and LTE broadcast channels directly. It also allow for detecting other types of attacks, such as the *Let me answer that for you* type of denial of service attack by Golde et al. [132]. In general this attack exploits a race condition, in which a fraudulent array of phones with a custom firmware answer a paging request before the genuine phone does. The following cipher handshake will almost certainly fail, leaving the GSM state machine no other option than to drop the call. As paging is broadcast over the whole Location Area (LA) this potentially affect a huge number of subscribers even when deployed only in one spot. A single LA can cover large portions of a multi-million inhabitants city [132, Fig. 8].

Based on paging statistic of over 470,000 paging requests of all three Austrian GSM networks we simulated how the distribution of paging broadcasts re-transmits will change in a network under attack based on the retry policies of the individual networks. A certain number of mobile stations does not answer on the first paging request (e.g., caused by a dead spot or interference) and has to be paged again. Some networks switch over from TMSI to paging by IMSI as a last resort. For our statistics we have to focus on TMSI paging, as there is no easy way to de-anonymize a large number of mobile stations at once. The distortions should be negligible for our purpose. We further conservatively assumed all paging requests within a 10 second window to belong to the original request. Only in very few cases (e.g., receiving many SMS messages in a brief period of time) this will not hold true.

Each paging request has a certain probability to not be answered by the target station on the first try and is therefore repeated. Based on the individual retry policy of each network, this produce a specific distribution on how many paging requests are tried a second, third, forth,... time. In our simulation we assumed a much less skillful attacker than in [132] with only 80% success rate and another one with 95% success rate. In both cases, the distribution of paging retries is severely distorted. Interestingly enough, some networks (i.e. T-Mobile and Orange) almost always page in pairs with just a few hundred milliseconds in between, in which case we grouped these requests.

Figure 4.7 displays how the retry-statistics is distorted from the normal empirical data (green) by applying a DOS attack with 80% respectively 95% success rate. Watching this relation can reveal such an attack against a whole Location Area, however it will not detect attacks against single phones (once the TMSI - IMSI pseudonmization is broken).

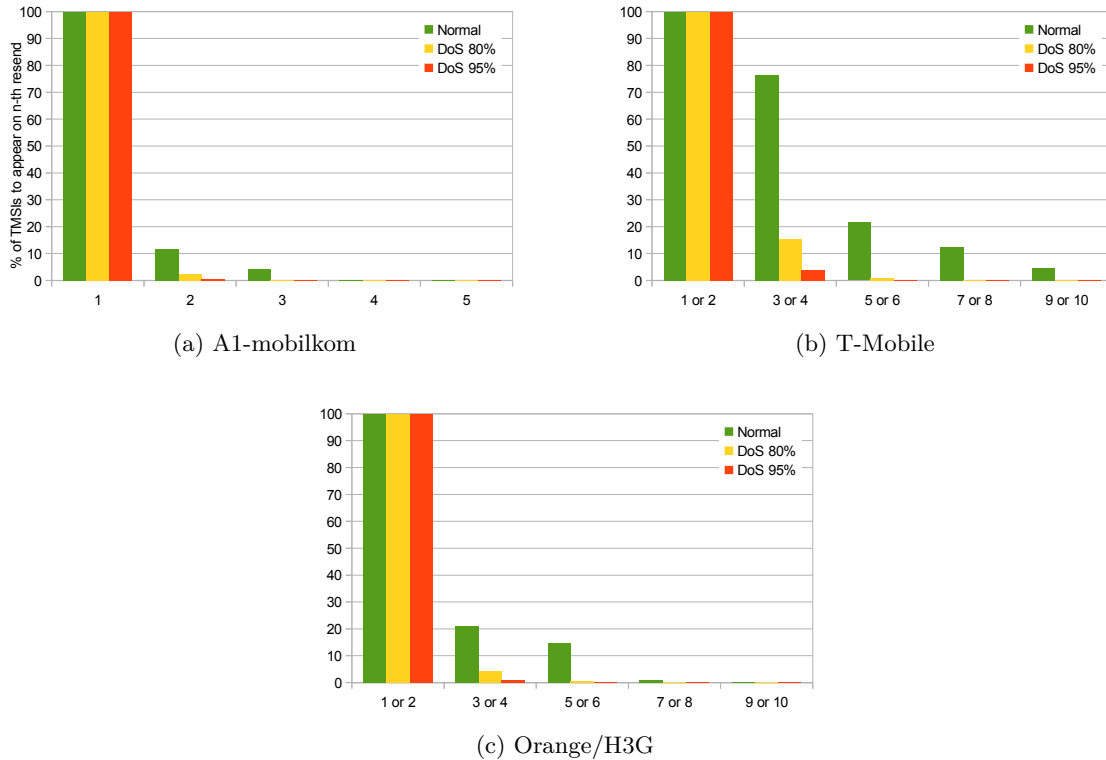


Figure 4.7: Number of TMSIs to (re)appear in the  $n$ -th paging resend within a 10 second window.

## 4.10 Future Work

We are currently experimenting with a new RTL2832U based stationary IMSI Catcher Catcher prototype. The RTL2832U [235] is used in many DVB-T/DAB television and radio receiver USB sticks in the US\$25 range. The chipset offers a way to bypass the DVB decoder and directly download 8-bit I/Q-samples with typically 2.8 MS/s turning it into a *Software Defined Radio* (SDR). Different tuner types exist, where the Elonics E4000 is the only one covering all major mobile phone bands by ranging up to 2200 Mhz. However, their extreme low price is to blame for the bad quality of many secondary components used. The oscillator accuracy can be as low as 50 ppm, leading to huge frequency offsets and shifts during operation. 30 kHz up or down is not a big deal, when receiving a multi-Mhz broad DVB-T signal. However, on a 200 kHz GSM signal they are very disruptive and need extra compensation.

Directly decoding the broadcast and control channels (i.e. BCCH and CCCH) gives much more insight and material for fingerprinting base stations (e.g., more details about

the organization of logical channels, broadcast traffic)<sup>5</sup>.

#### 4.10.1 Inversions and Tropospheric Ducting

Based on laser ceilometer [108] data from the Austrian central institution for meteorology and geodynamics (ZAMG) we have found a slight correlation ( $\phi = 0.21$ ) on reception of selected far off cells and border layers between 1000 and 2200 meters. This suggests that a better weather model might help us to understand the occasional excessive range of our stations. Eventually, this will allow us to clean up received data as these effects can produce similar short term reception patterns to briefly operated IMSI Catchers.

### 4.11 Conclusion

*IMSI Catchers* – as man-in-the-middle eavesdropping devices for mobile networks – became cheap and relatively easily available. Even in UMTS 3G networks, GSM 2G security is still important, as these networks are closely linked together, and therefore the *weakest link* principle applies.

Our goal was to survey, implement, and evaluate *IMSI Catcher Catchers* (i.e. devices that detect *IMSI Catchers*). We therefore identified structural artifacts thanks to which IMSI Catchers can be detected. Some of these can be mitigated, but not evaded completely.

Our first implementation is based on a network of stationary measurement devices with cheap and easily acquirable hardware. Data is collected in a central database for long time observations and then analyzed. We collected over 40 million datasets in 10 months. The second one is based on the Android platform and uses only publicly available APIs. Thus, ensuring its operability in future versions and on as many devices as possible. Furthermore, it neither requires special permissions nor rooting (or jail-breaking) of the phone. Because of its simple color-based warning system it is suitable for daily use.

Both solutions are not dependent on any external databases, as they collect all needed information by themselves. With an OpenBTS based IMSI Catcher, we validated the described methods. Both of our IMSI Catcher Catchers were able to detect the attack reliably, even in *identification mode* where the phone is captured for less than two seconds. In the future, we like to extend our tests to commercial available products. Our long term observation of real mobile networks with our fixed measurement devices was inconclusive at the time of writing.

Our results indicate that the detection of this kind of attack became feasible with standard hardware. Additionally, we described how to detect additional attacks on mobile networks, such as an recently published DOS attack.

Both implementations [85] have been released under an open source license.

---

<sup>5</sup>This data is mostly privacy neutral, as it contains public system information about the network and pseudonymized paging requests.

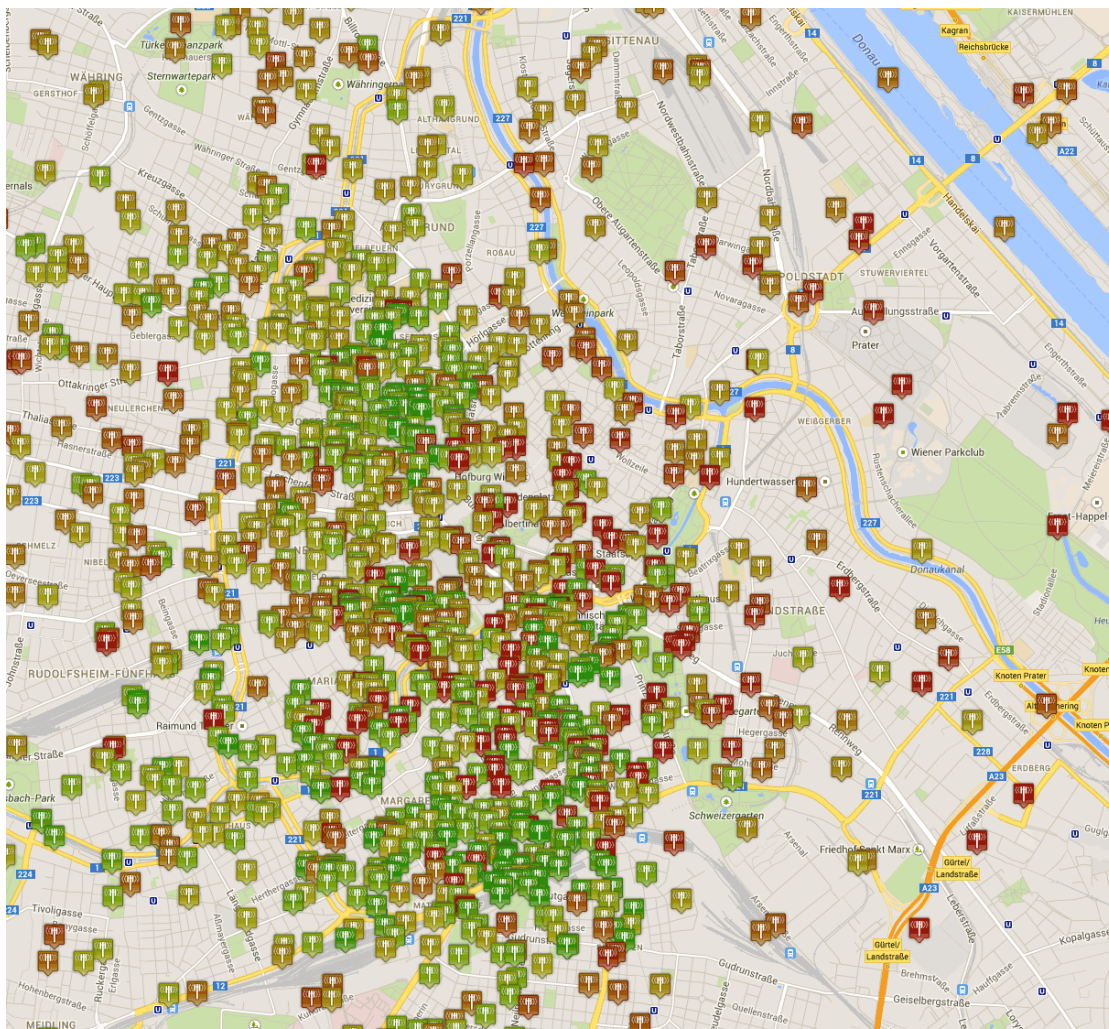


Figure 4.8: sICC: Color coded by signal strength of received cells. (Google Maps)

#### 4. CLIENT-SIDE MOBILE PHONE NETWORK ATTACK DETECTION

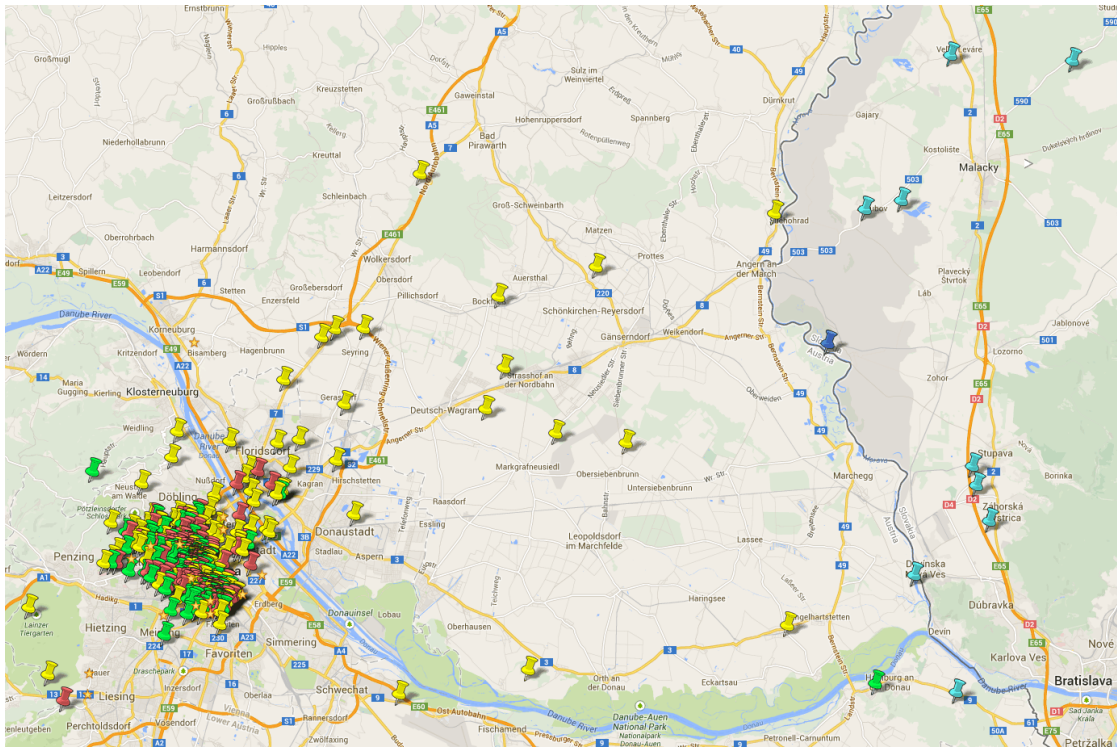


Figure 4.9: Stationary IMSI Catcher Catcher, Demonstrating the tropospheric ducting: Range up to 80 km. Color coded by mobile operator. (Google Maps)

# Operator-side Fake Base Station Detection

Fake base stations or IMSI Catchers are also of interest to the network operator. They introduce service outages and radio interference, and overall affect the customers trust into the network. In this chapter we are the first to present and discuss multiple detection capabilities from the network operator's point of view, and evaluate them on a real-world cellular network in cooperation with an European mobile network operator with over four million subscribers. One of the major challenges from the operator's point of view is that cellular networks were specifically designed to reduce global signaling traffic and to manage as many transactions regionally as possible. Hence, contrary to popular belief, network operators by default do not have a global view of their network. Our proposed solution can be readily added to existing network monitoring infrastructures and includes among other things plausibility checks of location update trails, monitoring of device-specific round trip times and an offline detection scheme to detect cipher downgrade attacks, as commonly used by commercial IMSI Catchers.

## 5.1 Context

IMSI Catchers are MITM (Man-in-The-Middle) devices for cellular networks [166]. Originally developed to steal IMSI (International Mobile Subscriber Identity) numbers from nearby phones, later versions offered call- and message interception. Today, IMSI Catchers are used to (i) track handsets, (ii) deliver geo-target spam [217], (iii) send operator messages that reconfigure the phone (e.g., installing a permanent MITM by setting a new APN, http-proxy, or attack the management interface [279]), (iv) directly attack SIM cards with encrypted SMS [227] that are filtered by most operators by now, and (v) also can potentially intercept mobile two-factor authentication schemes (mTAN). IMSI Catchers have become affordable, and can be build for less then USD 1,500 [78].

Pell and Soghoian [240] argue that we are currently on the brink of age where almost everyone is able to eavesdrop phone calls, similar to the 1990ies when cheap analog scanners were used to listen to mobile phones in the US and Europe.

In brief, these devices exploit the phone's behavior of preferring the strongest cell phone tower signal in the vicinity to maximize the signal quality and minimize its own power consumption. Additionally, on GSM networks (2G), only the phone (via the SIM - Subscriber Identification Module) needs to authenticate to the network, but not vice versa and can therefore be easily deluded to disable content data encryption. This enables an attacker to answer a phone's requests as if the phone was communicating with a legitimate cell phone network.

In contrast, the Universal Mobile Telecommunication System (UMTS, 3G) and Long Term Evolution (LTE, 4G) require mutual two-way authentication, but are still not completely immune to IMSI Catchers. Tracking and identifying IMSI Catchers are build on the weakness that a network has to be able to identify its subscriber before it can authenticate him/her. Additionally, unauthenticated commands can be used to downgrade a phone into using 3G or the less secure 2G (GSM) only, eventually giving way to a full Man-in-the-Middle attack. Additionally, some phones execute unauthenticated commands, even though the standard demands prior authentication [233].

This issue gains additional momentum as commercial networks increasingly surpass dedicated administrative and governmental networks in coverage and data rates and thus carry more and more increasingly sensitive data. Additionally, today, many economic sectors critically depend on a reliable and secure mobile communication infrastructure (e.g., logistics).

While most previous work focused on the detection of rouge base stations on the consumer side, this chapter takes the approach from the network operator's perspective and discusses novel detection capabilities from an academic as well as practical point of view.

The cooperation with T-Mobile Austria – a mobile phone network operator with over four million subscribers – enabled us to test theories, identify detection artifacts and generate statistics out of core network data. We focused on passive detection methods, readily available data in today's monitoring solutions and the identification of changes that promise better detectability and scalability.

## 5.2 Background

Previous work [91, 200, 228, 273, 281] focused on the subscriber (customer) side (see also Chapter 4); this work shifts perspectives and addresses the detection of such attacks from the operator side. The particular challenge lies in the structure of digital mobile networks: They were drafted in a time of low bandwidth connections, when signaling traffic occupied a significant amount of the network infrastructure. Therefore, these networks were designed in a highly hierarchical and geographically distributed fashion with as much signaling traffic as possible being handled locally or regionally, thus, offloading the



backbone. This poses unique challenges when acquiring and correlating the necessary data in order to detect anomalies in the network. Additionally, the legacy of having a GSM network being upgraded to UMTS and later again upgraded to LTE implies that the structure and the used data formats are not as clean and neat as one would expect from a freshly built LTE network with additional 2G and 3G radio front-ends.

Compared to the time when 2G networks were designed, today the ratio between user data and signaling data has completely changed. With LTE, users are offered 100 MBit or more.

The lowered backbone bandwidth costs and the (now) relatively low volume of signaling data allows mobile phone operators to en-bloc collect and monitor more data parameters than before. Many cellular network operators routinely collect data on different network levels and elements (e.g., from switches, servers, and via network probes) to detect, track and debug malfunctions and optimize their network. The strength of such Network Intelligence systems is to correlate transactions over different levels and protocols in the network structure, extract important values, and build an extensive index of the latter. This is done for several million signaling packets per minute. The limitation is that these indices are primarily built to search for traffic based on simple identifiers such as a specific customer, network element, protocol, or transaction type. Our goal is to use this monitoring systems to find far more complex symptom patterns that are typically produced by IMSI Catchers.

### 5.2.1 Working principles of a mobile phone network

For a brief introduction in mobile phone networks please consult Sections 3.2.2 ff., especially Section 3.2.9 on mobility management. All statements below on Location Areas<sup>2G,3G</sup> and Location Update Request (LUR) also apply to Tracking Areas<sup>4G</sup> and Tracking Area Updates (TAU) but the latter have been left out for better readability.

In idle state, the networks keeps track of the user not on cell level, but on the level of Location Areas (LA, a group of cells). A phone will issue a Location Update Request (LUR) when moving from one to another. Additionally, TMSIs (Section 3.2.3) are meant to be reassigned on Location Area changes, and some networks even reassign them on every interaction (e.g., call, text message) between the phone (MS, UE) and the network.

On a Location/Tracking Area Update message the phone will (usually) transmit its current TMSI and the old Location Area Identity (LAI, consisting of the *Mobile Country Code* MCC, *Mobile Network Code* MNC, and the *Location Area Code* LAC on GSM and UMTS) or *Tracking Area Identity* (TAI, comprising MCC, MNC, and the *Tracking Area Code* TAC). The *Mobile Switching Center* (MSC) for a Location Area can now fetch all the data about the subscriber from the old Location Area and inform the central user database (*Home Location Register* HLR or *Home Subscriber Server* HSS ) about where to reach that subscriber from now on.

Additionally, Location Area Update Messages are the Swiss army knife of the *Mobility Management* (MM) in mobile networks: A phone freshly turned on will first try to make

a *Location Update Request* (LUR) using its last known (cached) values. If its TMSI hasn't expired and is valid in this Location Area, the network will accept the phone. Otherwise it will trigger a re-authentication<sup>1</sup>. Therefore, even a phone arriving on a plane from another continent will first try to perform an LUR providing the LAI/TAI data from another network. This is intended, as it allows for national roaming and seamless handover of active calls across an international border.

Additionally, a ME/UE will perform periodic Location/Tracking updates, even when not moved in an interval configured by the network (e.g., 24 hours) to assure the network of its continued presence.

Periodically during operation and at shutdown, parts of the baseband state are stored on the SIM card and the phone itself. For example, instead of performing a full frequency scan for all receivable base stations at power on, the phone will first try the frequency range where it received signals from its mobile phone network before. Also, it will retry its old TMSI in an attempt to speed up the procedure. (After all, if the phone has not been offline for too long, it still could be valid.)

### 5.3 Detection System Design and Data Sources

For the development of our detection methods, we tested the interaction of 22 phones between an IMSI Catcher based on an USRP [114] and a mobile phone network. After that, we were able to retrieve log and PCAP files from the mobile phone network's monitoring system for analysis. Based on that we developed detection strategies and implemented them. We tested them on real monitoring data and counter checked them with statistics from the real network.

Based on our NDA and the secrecy of telecommunications laws we had to work on site and where not allowed to take any actual data outside of the building. Additionally, the limitations of the current monitoring systems only allowed us to retrieve data based on simple queries and a specific buffer size. For example, we could either retrieve data for a specific IMSI (e.g. our test SIM card) or a specific cell for longer periods of time, or a specific transaction type nationwide but only for a short time period (e.g. minutes), but not both.

The problem lies in the scattered transactions in mobile phone networks that forbid a natural global view on the status of a network. Thus, state-of-the-art mobile network monitoring put probes next to the MSCs which preselect and extract key values out of the signaling traffic. This signaling traffic is heavily depended on the access technology. A database cluster collects this data and makes it available based on simple queries on the extracted features. This system has to deal with high loads: e.g. just the Location Updates for 2G and 3G peak at roughly 150,000 transaction per minute during daytime, whereas the 3G transaction are more complex and consist of more packets than on 2G.

---

<sup>1</sup>A network operator can trigger re-authentication and TMSI reassignment also at other times.

The number of returned transactions on a query is limited by a (rather small) return buffer. However, data can be retrieved and reassembled to complete transactions which include everything from the initial mobile request, its way through the network instances up to the database access at the HLR and back to the mobile. This data can be exported to text and PCAP files for further analysis. Basically, any data extraction has to be reimplemented for each access technology. Even if the high level behavior (e.g., Location Updates) are quite similar, the signaling traffic is completely different on a technical level.

This setup sets limits in the ability to analyze data for complex anomalies such as finding network areas with higher than usual non-adjacent neighbor location updates (see Section 5.5.3). Therefore, we tested our programs and made our statistics on data sets consisting of several thousands up to 47,000 transactions, based on the type of transaction. With small changes in the monitoring system (e.g. extraction and indexing of additional values by the probes) our solutions below can work on much larger data sets or on real-time data (e.g. they can request a much more focused selection of packets, and don't have to filter them themselves).

## 5.4 Tracking IMSI Catcher

A tracking (or identifying) IMSI Catcher does not hold a mobile device in the fake cell, but drops it back into the real network immediately. For an attacker it is advantageous to simulate a new Cell-ID as well as a new LAC as this will always trigger an active communication (Location/Tracking Update) from the attracted mobile device.

Simulation of a new Cell without a LAC leaves the attacker without knowledge which phones are currently listening to the broadcast channel. He/she could only page previously known subscribers (based on IMSI) to verify their existence. Additionally, it will disturb the availability of the attracted phones for the complete operating time of the IMSI Catcher.

Unless for very specific operations, for the above mentioned reasons, an attacker will most likely choose a fake Location/Tracking Area Code (LAC) (or one that is unused in the geographical area) so that every mobile phone attaching to this cell initiates a Location/Tracking Update procedure. This informs the attacker of every phone entering the cell, gives him/her the ability to download identification data and then reject the Location/Tracking Update. Depending on the error cause used, the phone might return later (temporary error), or put the LAC or MNC on a blacklist (permanent error). An attacker wishing to enumerate all phones again simply chooses another LAC. This procedure disturbs each phone for less than a second per scan and has no major implications on availability.

Figure 5.1 (upper part) presents the message flow. Known IMSI Catchers download the IMSI and IMEI since both are easily retrievable. The IMEI is also commonly downloaded

## 5. OPERATOR-SIDE FAKE BASE STATION DETECTION

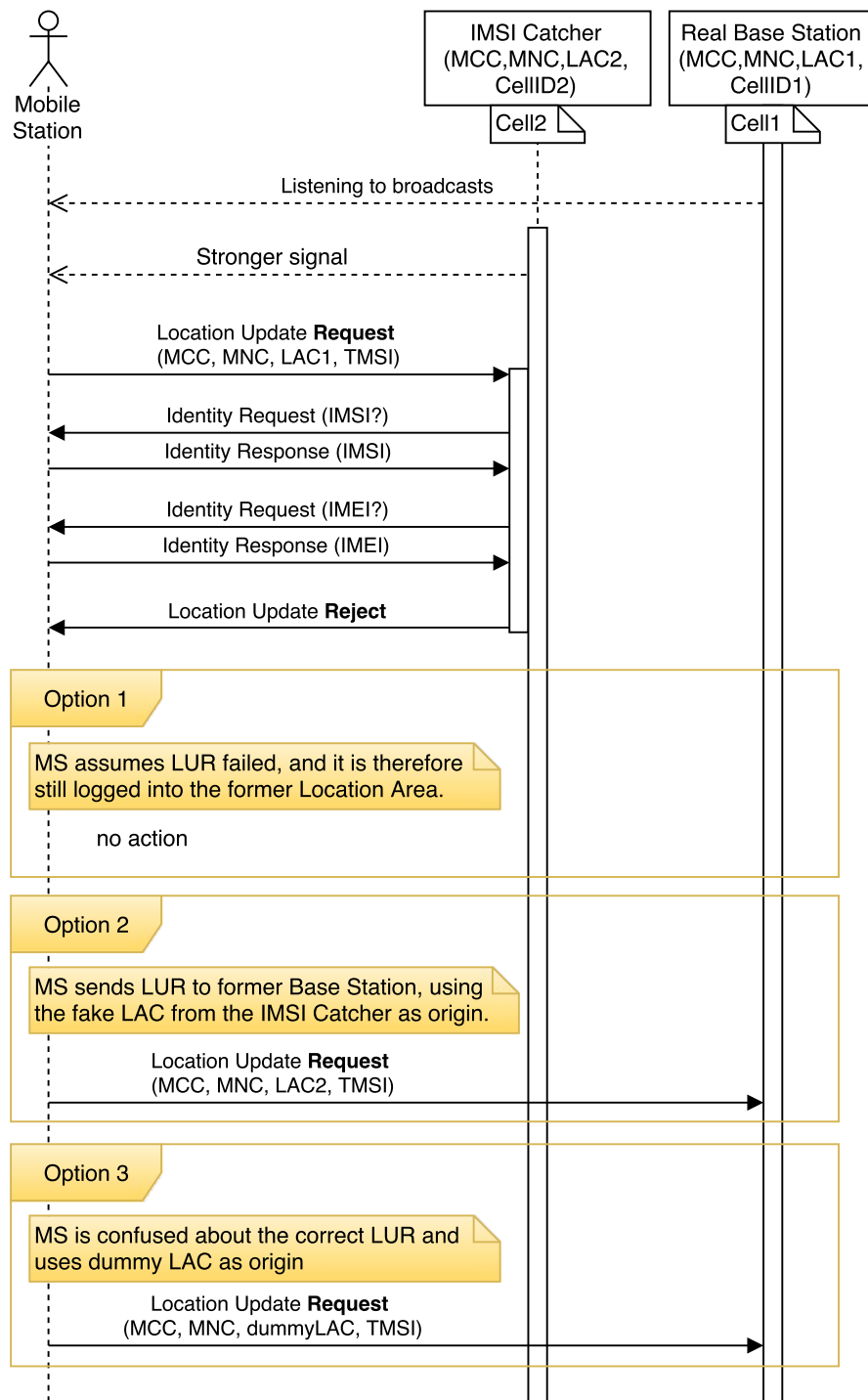


Figure 5.1: A tracking IMSI Catcher identifies a phone and drops it back into the real network.

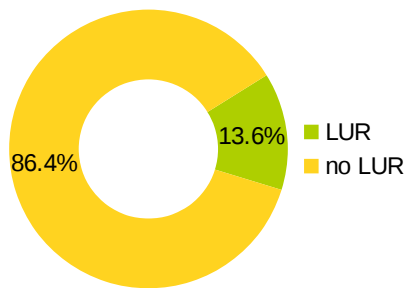


Figure 5.2: Phone models that produce a new LUR after a Location Update Reject (n=22 test phones)

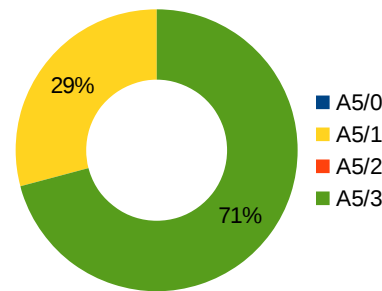


Figure 5.3: Cipher usage on 2G nationwide (n=7402 call setups)

by genuine networks in order to apply the correct protocol (workaround) policy based on the phone model.

#### 5.4.1 Detecting phones when reattaching to the original network

From the operator's point of view, a phone leaving the network for a fake cell is invisible. If there should be a page request in the mean time, the phone will not receive it. However, since the phone is away for only a short period of time, it will likely receive a retransmit of that page request.

Once the phone receives a *Location Update Reject* message, it has three options (cf. Figure 5.1):

- Assume that it is still known by the network at its old location. Therefore, no new message is needed.
- A new Location Update Request is sent to the network using the IMSI Catcher's Location Area Code as origin (see also Section 5.5).
- A new Location Update Request is sent using a dummy Location Area Code, since the last LAC value isn't valid.

We tested 22 different phone models<sup>2</sup> for their behavior after they dropped back into the genuine network in 2G (Figure 5.2). 86% produced no Location Update (Option 1) and 14% generated<sup>3</sup> a Location Update Request with a dummy origin-LAC 0xFFFFE (65534). The special values 0 and 0xFFFFE are reserved when no valid LAC is available by the MS/UE [4, 113]. Additionally, on GSM many phones also use 0x8001 (32769).

<sup>2</sup>Nokia Lumia 920.1, E71, 6310, 6150, 3210, 3710A-1, LG Nexus 4, Nexus 5, Apple iPhone 4, iPhone 6, Nexus One, Motorola Moto G2, Moto G XT1032, Samsung Galaxy Nexus, Galaxy S3, Galaxy Xcover2, Galaxy S5, Sony Xperia Z2-SCR10, BG Aquaris E4.5 Ubuntu Phone, Kyocera Torque KS-701, Sony Ericsson ST171

<sup>3</sup>All Nokia models introduced before 2000.

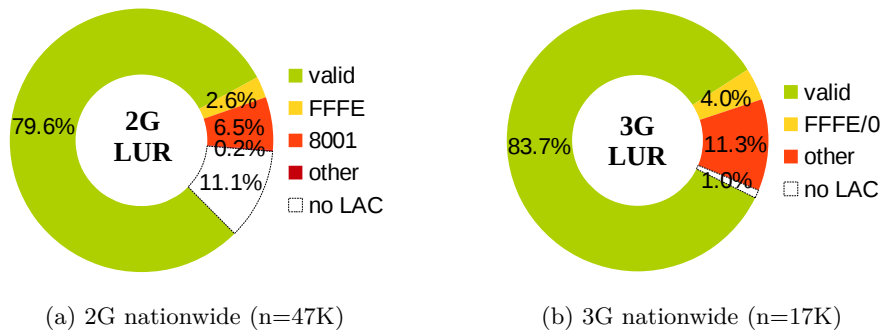


Figure 5.4: Origin LAC provided at Location Update Requests. *Valid* means that the LAC is within the local LAC plan. *0*, *0x8001*, and *0xFFFE* are literal (dummy) values. *Other* are LACs from outside the network (e.g. international or national roaming, accepted and rejected). *No LAC* describes the requests that do not provide a valid LAC or that provide dummy Values for MNC and MCC as well (such as 0x00 or 0xFF)



Figure 5.5: Some of the tested phones

However, these dummy LACs are no direct indicator for an IMSI Catcher even for this minority of phones, as they are used quite regularly. In a dataset containing all nationwide 2G Location Update Requests within one minute (daytime) we found 9.1% of all transactions using a dummy LAC and 11.1% using no LAC at all (see Figure 5.4a) without any geographical pattern. The numbers for 3G (Figure 5.4b) are smaller: 4% of Location Update Requests contained a dummy LAC (0xFFFE or 0x0000) from the same network. 1% contained also dummy values for the Mobile Country Code (MCC) and Mobile Network Code (MNC).

64% of our test phones generated a *GPRS Attach*<sup>4</sup> request within the next two minutes,

<sup>4</sup>Technically, this is an Location Update Request with *Origin LAC* set to the current LAC and an optional GRPS header with the Attach-Bit set.

if and only if it had a data connection before and did not have an additional WiFi connection. This is due to the fact that our test setup did not indicate GPRS support for the fake cell. Such a GPRS Attach request is nothing extraordinary and happens regularly (42% of all Location Updates on a real network contain such a header) for example if a phone drops out of WiFi and needs an Internet connection.

18% of this *GPRS Attach* messages had the *No Valid TMSI available* flag set. However, on a real network 4.5% of LUR messages have this flag set.

## 5.5 Capturing IMSI Catcher

An IMSI Catcher of this type holds the mobile in the cell and can therefore man-in-the-middle any transaction, and has control over the mobile phone by means of any network management commands (Figure 5.6).

### 5.5.1 Detection of cipher downgrades

A man-in-the-middle IMSI Catcher has to forward the traffic to the network. An easy way, is to tap into the cipher negotiation sequence and change the set of supported ciphers. The easiest choice for attackers is A5/0 (no encryption) and A5/2 (the weakened export-variant of A5/1), as described in Section 3.10.4. However, many networks (incl. T-Mobile Austria) banned these ciphers for years.

Instead, they started to support the A5/3 cipher [274]. On GSM this is the only cipher without (publicly) available rainbow tables or other decryption methods.

However, many MS still do not support this mode. On our network, in September 2015, 29% used A5/1 and 71% A5/3 (Figure 5.3, n=7402). Other cipher modes were prohibited in this network.

An operator-run database of {IMEI, highest-used-cipher}-tuples provides the basis to detect cipher downgrades. This database is updated on first contact with the network and whenever a device uses a higher ranked<sup>5</sup> encryption than the one stored. As long as there is no SS7/Diameter standard on exchanging this form of information, every operator has to run their own database (or include it into the HLR/HSS). Once the highest available cipher of a device is established, the network should not accept a lower one, or at least generate a warning. Thus, making a downgrade attack visible to the operator except when the user is attacked on the very first contact with a new network. Except for a firmware bug, there is no reason why a device should stop supporting higher cipher levels.

### 5.5.2 Detection of relayed traffic

The most compatible and least interfering way for a capturing IMSI Catcher to operate is to relay all traffic. If it is encrypted with A5/1 or A5/2 the decryption can be done

<sup>5</sup>Encryption strength: A5/0 < A5/2 < A5/1 < A5/3

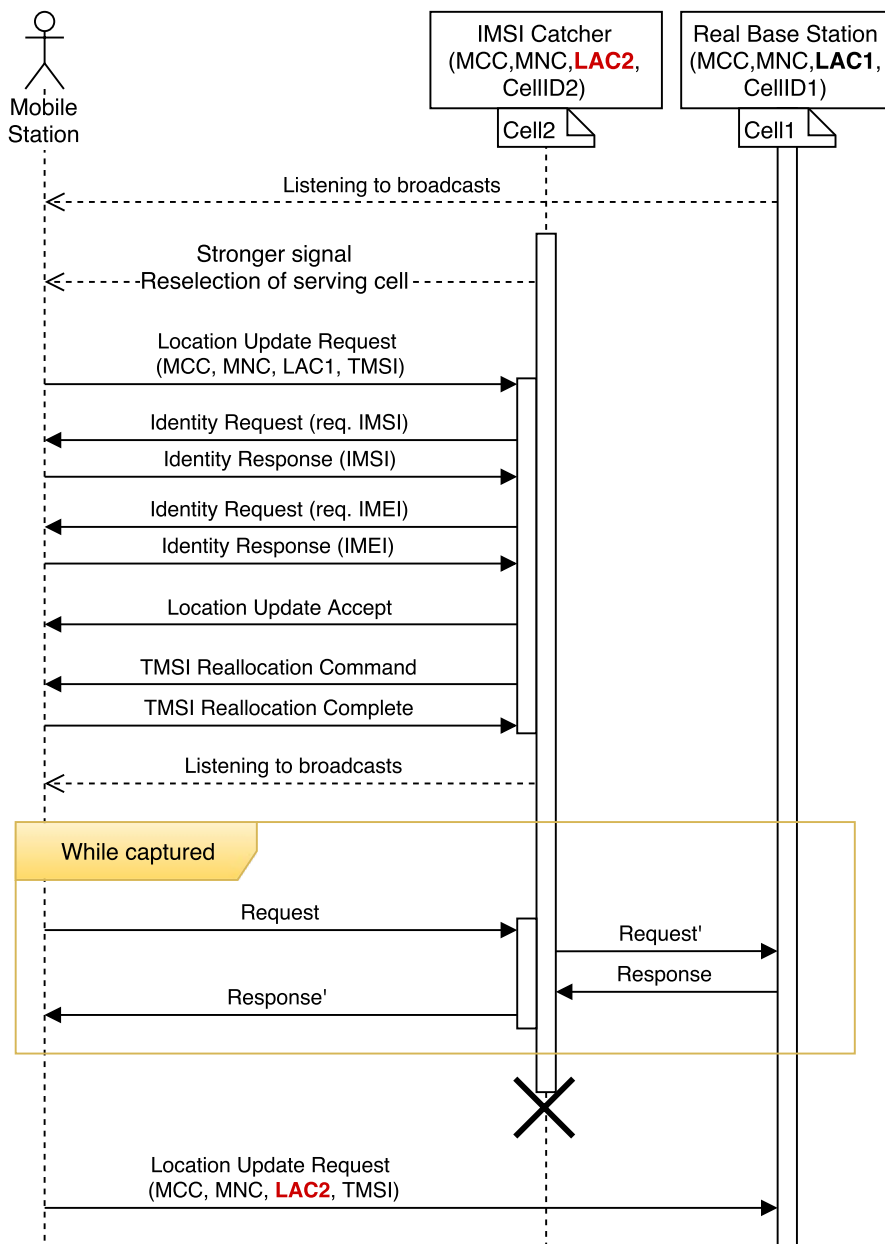


Figure 5.6: A man-in-the-middle IMSI Catcher identifies a phone and withholding it from the real network. During fall-back into the real network, the captures phone gives away the LAC of the IMSI Catcher.



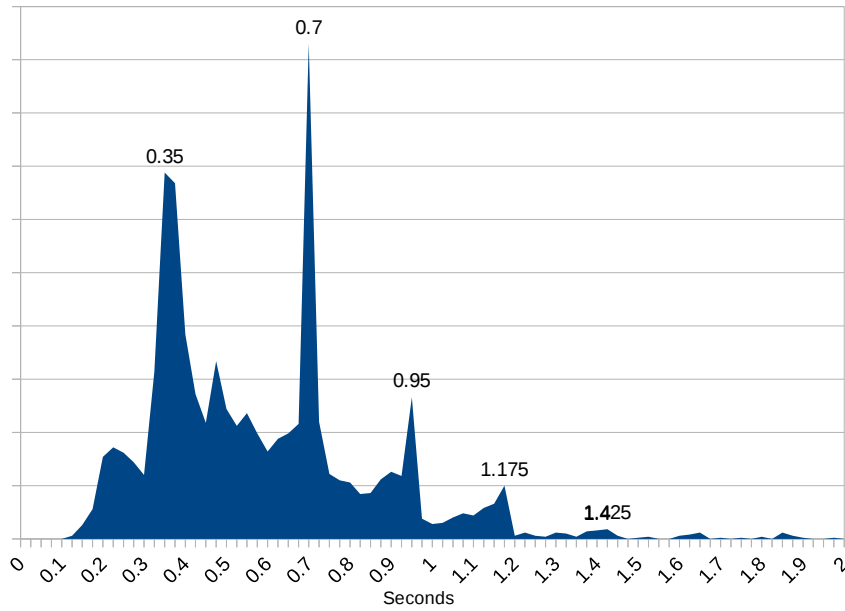


Figure 5.7: Authorization round trip time: Distribution of time between *Authentication Request* and *Authentication Response* on a real network.

separately, otherwise it has to be downgraded. Based on enough traces, the session key  $K_c$  can be reconstructed [156, 183]. In conjunction with another vulnerabilities (e.g., weak COMP128), also the secret authentication key  $K_i$  can be read and the SIM card cloned [73]. Once  $K_c$  is known, this allows an IMSI Catcher to decrypt A5/3 as well, since the  $K_c$  is used for all ciphers. For SIM cards with only a 64 bit key, the  $K_c$  is doubled  $K = \{K_c || K_c\}$  to 128 bit and therefore allows decryption of UMTS as well<sup>6</sup>.

We tested if the analysis of the round-trip times can be a good measure to uncover traffic relay. Therefore, we analyzed authorization round trips in the wild of 4165 random transactions within one minute, nationwide. The histogram in Figure 5.7 shows a high deviation ( $\bar{x} = 0.586$  sec,  $\delta = 0.334$ ) of response times with a notable retransmission interval of about 0.25 seconds. We estimate that a well-designed traffic-forwarding IMSI catcher could relay the traffic in 100 ms or less, thus being far from statistically significant in single instances.

Further analysis presented vast differences between manufacturers as well as handset types. Based on the *Type Allocation Code* (TAC)<sup>7</sup> we run independent nationwide collections. Figure 5.8 shows 12 diverse popular handset types and highlights three different iPhones to illustrate their different behavior (based on an average of 3,400 transactions per phone

<sup>6</sup>The attacker has to brute-force the 48-bit sequence number, though.

<sup>7</sup>TAC are the first 8 digits of an IMEI that encode the manufacturer and phone model. Popular models might end up with multiple assigned TACs. This is somewhat similar to the assigned OUI prefix in Ethernet MAC addresses: they encode the manufacturer.

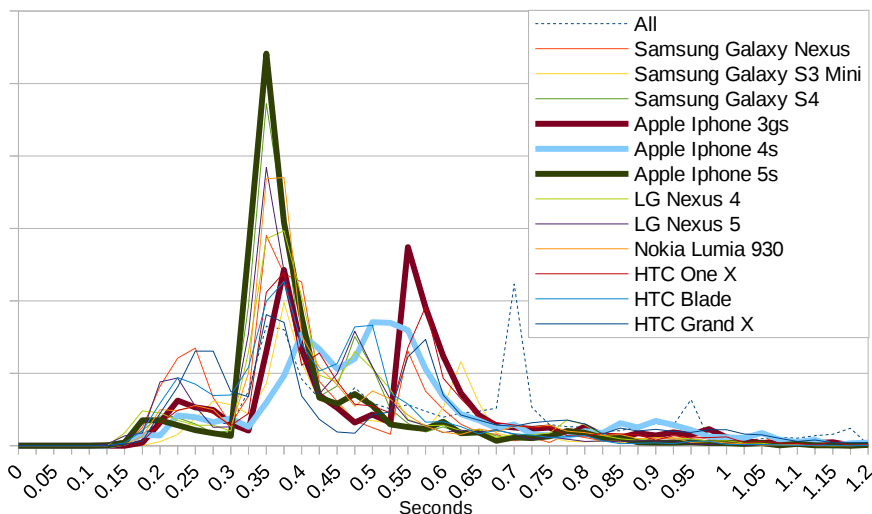


Figure 5.8: Normalized distribution of authorization round trip time broken up by phone models. Three Apple phones highlighted to show the distinct differences in their authorization response time. ( $n \approx 3400$  for each phone type)

type). Since these values have a much smaller standard deviation (e.g.,  $\sigma_{\text{GalaxyS4}} = 0.198$ ,  $\sigma_{\text{IPhone3gs}} = 0.200$ ,  $\sigma_{\text{IPhone4s}} = 0.206$ ), they are a better basis to detect relay delays (i.e. average authorization round trip time increases on multiple occasions for a single user). Additionally, a provider side detection can correlate such changes geographically (i.e. average authorization round trip time increases in a geographical area).

### 5.5.3 Detection of unknown, unusual or implausible origin-LAI/TAI in Location Update Requests

Eventually, every IMSI Catcher victim falls back into the genuine network (Figure 5.6). During this step, the LAC of the attacker is leaked back into the real network<sup>8</sup>. As stated above, it is favorable for an attacker to choose an unused LAC as this forces every victim to actively contact the fake base station on entrance and therefore inform the attacker about its capture. This LAC is either completely unknown in the genuine network or far away.

We investigated the possibility of creating shadow instances that follow every location area update and reject implausible location changes. While the current monitoring infrastructure does not allow to monitor all location updates nationwide for all mobile phones (Section 5.6.3), we scaled down and implemented a prototype that is able to follow individual UE/MS through different access technologies based on PCAP files from the core network. The two main investigated properties are (i) the correctness and

<sup>8</sup>See Section 5.6.3 and 5.6.4 for further discussion and possible mitigations.

completeness of location update trails and (ii) the geographical plausibility of location updates (i.e. only adjacent locations).

The *correctness and completeness of location update trails* means that location trails form an uninterrupted chain. A gap would be a strong hint for a visited LAC to not be under the control of the operator. The *geographical plausibility* checks if updates only occur between geographically neighboring locations. This neighbor property does not have to be derived geographically, but can be established statistically (i.e. recording frequent location updates between Location Areas). Unless operators agreed on national roaming, the phone stays on the home network, so no operator collaboration is necessary.

In the following evaluation we discovered a number of corner cases that complicate the interpretation of the results.

### 5.5.3.1 Power on at a new location

UE/MS not always correctly detach from a network when turned off (e.g. battery loss, temporary reception loss during power off). At the next power on, the UE/MS will use the previous LAC as origin for a location update. Imagine this plausible case as depicted in Figure 5.9: A flight passenger turns off the phone at takeoff in one city, but the *IMSI deattach* message was not produced or did not arrive at the network. After landing, the passenger turns the phone back on during the train ride from the airport to the city. In most cases, the phone will send a location update to the network as if it just passed the border between the two location areas. This even happens after intercontinental flights. Airport cells could be whitelisted to some extent, but they will not catch all cases (such as in the example above).

Because such (tunneled) location update are indistinguishable from a direct location changes, they are not immediately a red flag.

Additionally, road and railway tunnels also offer geographical shortcuts, but – unlike plane routes – the ends of the tunnel only connect two points and will be statistically assigned as neighbors, since a large number of passengers traverse without turning off their phones.

### 5.5.3.2 Old baseband state restoration

Phones regularly and at certain events save parts of the baseband state information to non-volatile memory. For faster boot times, the phone can facilitate this information (e.g. already knows the frequency range of the preferred operator and does not has to scan the whole frequency range). This includes the last known LAC.

One of our test phones had a defective power button which lead to random reboots. In the traces we discovered that the phone sometimes used obsolete LAC information as origin (i.e. reused a LAC as origin a second time, because another location change was not recorded properly before reboot).

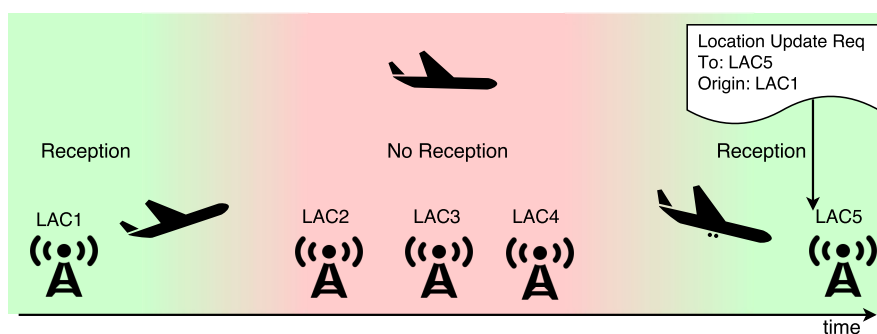


Figure 5.9: Location update tunneling effect: Because a detach message is not guaranteed, location/tracking area updates happen between non-adjacent cells.

#### 5.5.4 Detection of a access technology downgrade

As described in Section 3.10.5.2 and Figure 3.6, access technology downgrades are easy to perform and included in today's commercially available IMSI Catchers [124]. A phone camping on 2G even though 3G or 4G should be available in the area is not a strong indicator. In some cases, structural properties can lead to better reception of certain frequency ranges (e.g., 2G on lower frequencies is usually better receivable underground). On the other hand, a MS/UE can be set intentionally to use 2G only for power conservation. A provider could install an application on the SIM to monitor the access technology and location updates; however, this is out of scope for this work.

## 5.6 Discussion

We identified strong and weak indicators based on the statistics of certain features in real-world data. Strong indicators have low potential for false positives.

A **per device (IMEI) database of the highest-used cipher** can reliably detect cipher downgrades or deactivation of ciphering. Additionally, we have shown that mobile phones **leak the (fake) LAC of the capturing IMSI Catcher to the real network**. This case can trivially be detected based on analysis of Location Update Requests. If the attacker misuses a genuine LAC, it can still be detected by a **consistence check of the Location Update trail**. Based on certain corner cases, the latter has the potential for false positives (LUR tunnel effect, restoration of old baseband states) and therefore needs to be backed up by additional geographical, temporal and subscriber based correlation.

Another method is the **transmission delay introduced by an MITM attack**. We tested this technique based on the authorization round trip times. In general, the deviation is quite large, but can be narrowed if the device type is considered as well. Every device has a very specific distribution of round trip times. However, for a statistically significant result (e.g. for a device under attack), multiple measurements have to be collected.

---

From the provider point of view, the hardest attack to detect is that of a tracking-only IMSI Catcher. Except for a few very old phones, this particular attack does not produce any messages in the core network. It has still to be explored if certain frequency-monitoring functions on BTS, NodeBs, and eNodeBs can be repurposed to detect such rouge base stations.

### 5.6.1 Ethical Considerations

As described in the research set up (Section 5.3) we have used real data only under very strict conditions to comply with ethical and legislative requirements. We have only worked on signaling data and never had access to user data or personal subscriber information.

### 5.6.2 Comparison with client detection methods

Operator detection of IMSI Catchers does not supersede client detection (c.f. Section 5.7). It complements it and gives the operator the opportunity to monitor such attacks in its network regardless of precautions by individual subscribers. However, since the detection schemes can only find phones that are either under the control of an attacker - or just switched back to the genuine network - the operator can only warn the user in question post-attack.

On the other hand, client based techniques give the user the ability to detect a current attack against his/her very device. On tracking IMSI Catchers this technique provides better detection rates.

### 5.6.3 Limitations

The current implementation of our detection methods is based on the old somewhat limited monitoring system deployed in the network. It can filter some pre-extracted of each packet and transaction against a query containing a limited set of operators and literal values (i.e. filter by a specific cell, IMSI, IMEI, protocol type, etc). It can not compare between cells or apply more complex filters. Additionally, the return buffer size is limited to 10K-30K results, depending on the search mode. This limits our current implementations to single users (or single cells) at a time. This is the reason we could not run a nation wide search so far.

### 5.6.4 Future Work

Our results show that detection from the operator side is possible and tested its usefulness within the limitations of the current monitoring system. We suggest that parameters such as ciphering and origin LAC in Location Area Updates should be extracted directly at the probes and made available. This pre-selection step will eliminate current limitations. For example, it will allow to search for inconsistencies in used ciphers, based on the IMEI

(or TAC). Additionally, a new monitoring system based on Apache Hadoop is currently in development that is expected to remove most limitations of the current system.

With the large number of dummy LACs used by phones, one can wonder if an attacker could use dummy LACs such as 0xFFFE for masking their existence. Another way, to mask the fake LAC of an IMSI Catcher is, to announce a neighbor frequency occupied by a second IMSI Catcher with a reasonable LAC. While doubling the hardware costs for an attacker, this might whitewash the *Origin LAC* field used in Section 5.5.3. Both ideas need further testing with end devices to confirm or deny their practical feasibility. As discussed before (Section 5.5.4), a SIM card application can monitor and report certain network parameters back to the network (e.g., keep a local copy of a CellID/LAC trail) and detect both cases. However, over time, many different cards from different vendors have been acquired so developing and maintaining such an application poses a financial burden and an operational risk.

Furthermore, we plan to refine the timing models used in Section 5.5.2 to become more accurate and create better models for timing delays introduced by traffic relaying.

## 5.7 Related Work

So far, IMSI Catcher detection has almost exclusively been tackled from the clients' point of view. Malet and Nohl first developed a solution for OsmocomBB phones, and later on for rooted Android phones with a very specific Qualcomm chipset [200, 281]. Other applications replicated similar client side detection without the need for a rooted phone [91, 273].

Van den Broek et al. proposed a pseudo-random IMSI that will not allow others than the home operator to distinguish particular users [311]. However, this will introduce a higher overhead in the roaming case and needs to be extended to cover cases where IMSI Catchers use additional identification numbers (such as IMEI).

Van Do et al. are so far the only ones to look at the provider side [312]. Their solution is based on encryption elimination detection and anomalies such as disappearance of a large group of phones in a geographical area, fed into a machine learning system. However, their approach has limited applicability, for real world networks: Disabling encryption is only found in older capturing IMSI catchers and disappearance detection has a latency up to 24 hours – the time scale of periodic location updates (i.e., the mobile phone's periodic reassurance to the network). This will only detect IMSI Catchers operating for an extended amount of time.

## 5.8 Conclusion

IMSI Catchers are still a major problem for today's networks: (i) Tracking IMSI Catchers work directly on GSM, UMTS, and LTE networks as Location/Tracking Update Rejects are excluded from cryptographic message integrity checks. Mutual authentication only

prevent plain capturing IMSI Catchers. (ii) These reject messages can be used to downgrade a phone until the next reboot to a lower access technology (e.g. GSM) without mutual authentication. Therefore, the weakest-link principle applies.

In this chapter we analyzed the different types of IMSI Catchers and their produced artifacts as well as if and how they can be detected from the network operator's side. Due to our cooperation with an European carrier we have been able to systematically perform real-world experiments and test our detection methods on real world-data.

Strong indicators we identified are for example the usage of invalid LACs (which are transmitted by the phones when they fall back to the genuine network after an attack), or the usage of weak ciphers to detect downgrade attacks for devices that were previously able to use strong ones. Additionally we showed that a number of weak indicators can be correlated geographically, temporally, and on subscriber basis e.g., for detecting targeted attacks, similar to current fraud detection schemes used by credit card companies. This includes fingerprinting devices based on profiles, unusual movements, and implausible location update trails. We also addressed corner cases and how to deal with them.

As mobile networks were initially designed with the reduction of signaling traffic in mind, not all of the necessary information is readily available for analysis, or even not collected centrally and in a scalable fashion. Some of the indicators we identified therefore demand changes in the monitoring systems currently used in such networks. However, based on already available data from a real-world mobile network, we were able to show the practical applicability for multiple of our methods.





## **Part III**

# **Mobile phones and terminals**



## Side-channels Uncovering Browsers' History

In this chapter we show that HTTP Strict Transport Security (HSTS) headers [151] and long-term cookies (like those used for user tracking) are so prevailing that they allow a malicious Wi-Fi operator to gain significant knowledge about the past browsing history of users. We demonstrate how to combine both into a history stealing attack by including specially crafted references into a captive portal or by injecting them into legitimate HTTP traffic.

Captive portals are used on many Wi-Fi Internet hotspots to display the user a message, like a login page or an acceptable use policy before they are connected to the Internet. They are typically found in public places such as airports, train stations, or restaurants. Such systems have been known to be troublesome for many reasons. In this paper we show how a malicious operator can not only gain knowledge about the current Internet session, but also about the user's past. By invisibly placing vast amounts of specially crafted references into these portal pages, we can lure the browser into revealing a user's browsing history by either reading stored persistent (long-term) cookies or evaluating responses for previously set HSTS headers. An occurrence of a persistent cookie, as well as a direct call to the pages' HTTPS site is a reliable sign of the user having visited this site earlier. Thus, this technique allows for a site-based history stealing, similar to the famous link-color history attacks [57]. For the Alexa Top 1,000 sites, between 82% and 92% of sites are affected as they use persistent cookies over HTTP. For the Alexa Top 200,000 we determined the number of vulnerable sites between 59% and 86%.

We extended our implementation of this attack by other privacy-invading attacks that enrich the collected data with additional personal information.

This chapter is based on our paper [89].

The chapter is structured as follows: In Section 6.2 and 6.3 we expand on the technical background of history stealing and why it is such a lucrative attack. Section 6.4 describes the attack in detail whereas Section 6.5 expands on the differences with regards to mobile operating systems. We estimate the impact and applicability of the attack in Section 6.6 by crawling Alexa's Top 200,000 sites. After describing our proof-of-concept implementation (Section 6.7) and its extensions and limitations (Section 6.8), we present our conclusion in Section 6.9.

## 6.1 Overview

*Browser history stealing* discloses information about user's past browsing behavior without their knowledge, e.g., by visiting a website that mounts such an attack. The history is directly or indirectly extracted from the browser itself. One of the first and widely known methods facilitated the `:visited` attribute of links. Visited link targets can be rendered differently by the browser to ease navigation. Variants of this attack examine the displayed color [57] or load external images [69, 261] to determine if the user visited a specific URL earlier. Jang et al. [162] showed that several sites use these techniques to spy on their users.

Browser manufacturers reacted in two ways: First, they fixed a number of vulnerabilities and included mitigations, and secondly they introduced a *privacy mode* that exempts specific browsing sessions and associated data from appearing in the browsing history.

Captive portals are a technique to redirect the user on her first request to a portal website. These are heavily used in public Wi-Fi hotspot systems such as cafés, restaurants, airports, and hotels. The user is informed about the sponsor of the access, possible restrictions as well as potential payment methods and has to accept terms and conditions. After completion, the user is given access to the Internet. Especially in transit areas the majority of users use hand-held devices such as tablets and smart phones. Additionally, these devices automatically connect to known Wi-Fi access points. We specifically elaborate on these cases and their implications.

Public hotspots have been known to be troublesome in many regards: They are often unencrypted, fake access points can easily attract legitimate users, and they offer a single point through which all traffic has to pass. This gives the Wi-Fi or captive portal operator great power and insight on the users' current sessions.

In this paper, we describe how such an operator can also gain knowledge about a user's past. The attack can be carried out by the legitimate hotspot operator, but also by an evil-twin network pretending to be the real network [184], or generally by any man-in-the-middle attacker. They can trick the browser to disclose information that allows conclusions about the user's past browsing history. This also applies to VPN users, as they have to go through the login process before starting the VPN session and the browser keeps only one history regardless of the connection type.

## 6.2 Motivation

A browsing history is a comprehensive picture not only about a user's past activities, but also about their interests, political opinion, sexual preference, geographical or ethnic heritage, spoken languages, social contacts and so forth. For example, a user who visits localized websites is most likely from that region or speaks its distinct language (e.g. *amazon.fr*, *amazon.jp*). Visitors of *grindr.com* or *transblog.de* most likely have a very specific sexual orientation, whereas *okcupid.com* or *parship.com* visitors are probably currently single. Likewise, history entries of websites for certain medical conditions (e.g. pregnancy, AIDS, depression), political campaign websites, or those of religious communities do not require much imagination to draw conclusions about a particular user. A reference to *intranet.somecompany.com* tells much about the employer.

In the past, URL-based history attacks have also been used to uncover social network contacts and de-anonymize users [328]. This information is of great interest for targeted advertisement [162], but also for targeted attacks.

Multiple history stealing methods have been presented in the literature. They range from analyzing the link color (visited links can be displayed differently) via Javascript [57] or conditionally loading external resources based on CSS rules [261] to GPU timing attacks [187].

## 6.3 Background

In this section we briefly describe the technical foundations of this work.

*Captive Portals* are a technique to display a certain content when a new user connects to a (wireless) network. There are multiple ways to achieve that goal. The most common is to intercept the first HTTP request from the user's browser to any site and redirect it (spoofing *HTTP 302 Moved Temporarily* response) to the web page of the operator's choice. This web site (the portal) will explain the user the terms and conditions for the usage of the network and its Internet connectivity. Sometimes this includes payment, other times just advertisement, or acceptance of a legal disclaimer. After completion, the user's physical MAC address or local IP address is put on a whitelist and its traffic is passed unaltered to and from the Internet.

*HTTP cookies* [58] are a technique (to be precise: an optional header) within HTTP requests to add a common state between the browser and the server to the otherwise stateless HTTP protocol. On a technical level, cookies are small pieces of data (directly or indirectly) set by the server that the client's browser attaches every time it sends a request to a specific site. Cookies are often used to either directly store user configuration (e.g., language choice), indirectly store a state (e.g., a session identifier with the actual data stored on the server), authenticate a user (e.g., a login cookie), or uniquely identify a client (e.g., tracking cookie).

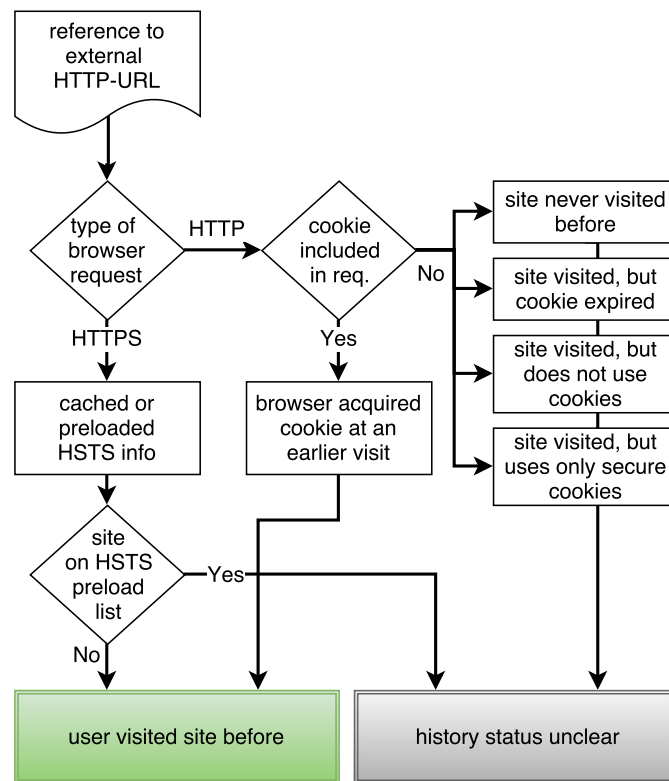


Figure 6.1: History stealing flow chart on users browser reaction.

There are multiple ways how cookies can be set in a browser. In the traditional way, the server sends a `Set-Cookie` HTTP response header to the client, giving the cookie a name and a value. Additionally, cookies can be set using Javascript on the client side. Many tracking libraries (e.g., Google Analytics) use this method.

By default, a cookie is only valid for the lifetime of a browser session. By setting an additional expiry date, the cookie becomes a *persistent* or *permanent cookie* which is able to survive multiple browser sessions. An *httpOnly cookie* can be set and read by the server, but not through Javascript. This is an option introduced against cookie stealing attacks via cross site scripting (XSS). A *secure cookie* is only presented by the client to the server on encrypted connections (i.e., HTTPS). Cookies can also be bound to a specific domain and path where it will be also used for subordinate domain names and paths (e.g., a cookie set for the domain *example.com* is also used by the browser for *docs.example.com*).

## 6.4 History Stealing in Wi-Fi Captive Portals

Wi-Fi hotspots are a popular method to access the Internet in public places. They are typically faster than mobile data connections and do not count towards a monthly mobile

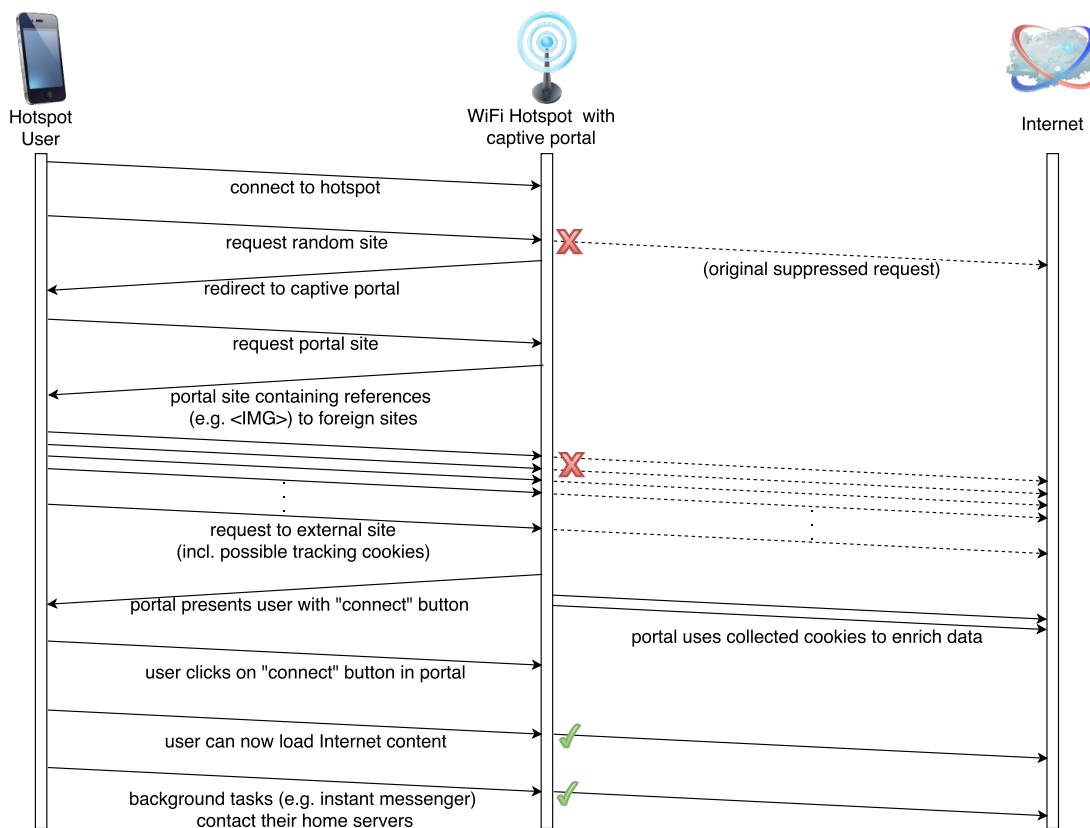


Figure 6.2: History stealing scenario in a public Wi-Fi hotspot setting.

data plan. For foreign visitors, they are often the only way to save on excessive data roaming fees.

Most users are familiar with captive portals. At the first request, the web browser is redirected to a portal page that in many cases contains the terms and conditions, a word from the sponsor, and – if applicable – payment options.

### 6.4.1 Stealing History

Just like most other browser-based history stealing attacks, this attack scheme requires a list of URLs or domains of interest, and testing each of them for their occurrence in the browser history. The history stealing method works by inserting image references such as `<img href="http://somedomain.com/nonexisting-file?customtag">` for each site into the captive portal's landing page. The client will look up the domain and try to fetch the alleged external resource. Each of these requests is (again) intercepted by the captive portal. The chosen filename and/or an added tag ensures that the portal can identify these requests, record the request (and its cookies) and return a dummy file (e.g., a 1×1 pixel image or an empty document).

The browser will include cookies for that site into the request if they are available in the local cookie store. Thus, a cookie included in the provoked request is proof of the user visiting the site before (Figure 6.1). A missing cookie could indicate that (i) the user has not visited the website, (ii) the website is not setting any cookies, (iii) the cookies already expired, or (iv) the website is setting cookies for a restricted path or sub-domain, or just for secure connections (*secure* cookie).

Websites trying to protect their users using HTTP Strict Transport Security (HSTS) [151] can also accidentally leak history information. HSTS allows websites to declare that browsers should only use encrypted connections. This information is transferred at the first visit of a browser to a particular website. Apart from about 6,500 sites on the public *preload HSTS list* used by particular browsers [6], a cached HSTS entry indicates that the user visited the site before.

Even though the staged requests are intercepted, they take some time. A waiting screen (e.g., “Please wait while you are connected to the Internet”) can put off the user only for a limited time before she/he unnervingly closes the window (or the app). We therefore also inject these tagged image references into later HTTP requests with HTML content using a transparent man-in-the-middle proxy on the gateway.

Image (`<img>`) tags are not the only option to generate external requests by the browser. Basically any method used in Cross-Site-Request-Forgery (XSRF) attacks works. Image tags have the advantage of not requiring any Javascript on the client side. They work considerably faster than `<iframe>` which produce considerable overhead at the client.

The attack heavily relies on the usage of long-term cookies by websites. We elaborate on the prevalence of long term cookies in Section 6.6.

### 6.4.2 Unintentional Connection

Once a Wi-Fi network is known to a phone or tablet, every further connection will be made automatically in most cases (e.g., Android, iOS).

For our attack, we will either operate a Wi-Fi hotspot that users actively connect to or simulate a known one (*Evil-Twin* attack [184]). The Wigle project [14] provides a collection of popular Wi-Fi names (SSID). For example *attwifi*, *BTOpenzone*, *public*, *Guest*, *Free Public WiFi* or *BTWIFI* are often used for public hotspots. Default Wi-Fi names are also a good guess, as many smart phone users might have used them once before: *linksys*, *default*, *dlink*, *belkin54g*, or *ZyXEL*.

## 6.5 Avoiding Minimalistic Browsers for Captive Portal Login

Computer users will typically use their default desktop browser to be captured and perform the login procedure. Thus, they expose the history of a full-fledged browser that is most likely used for their day-by-day browsing.



In contrast, modern mobile phone operating systems try to detect captive portals and offer the user a way to quickly log on with a minimalistic browser. This browser does not share the history with the main browser.

Android and iOS perform different connectivity checks and display a notification if they assume that user interaction is needed. For example, iOS performs a captive portal test since version 4 and Android since version 4.2. Before Android 5.0, the default system browser was used to load the captive portal which directly exposes the history. Since Android 5.0, the operating system starts a captive portal browser, basically a lightweight browser in privacy mode (e.g., without history).

To prevent the usage of a stripped-down browser, the attacker can fool the online check into believing it is connected. This way, the victims will not get a notification and will use the default browser for the captive portal, thus exposing their actual browsing history.

As stated above, this does not typically apply for users with a desktop browser, as they will use their default browser. However, we know of two exceptions: Chrome OS/Chromium and Mac OS X since 10.7. Both perform the exact same connectivity test as their mobile OS counterparts and can be circumvented in the same manner.

### 6.5.1 Circumventing the Connectivity Test on Android

Android, Chromium [11], and Desktop Chrome [18]) creates an HTTP request to one of the Google servers and checks for a specific return code. A plain captive portal will try to redirect the user during this request and rewrite the return code.

The connection manager binds an HTTP client to a specific interface (e.g. Wi-Fi) and tries to request `http://clients3.google.com/generate_204`. A response with a 204

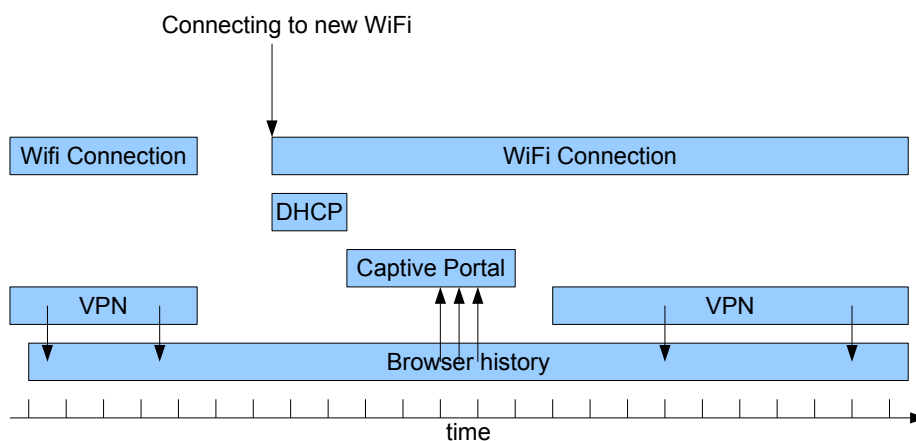
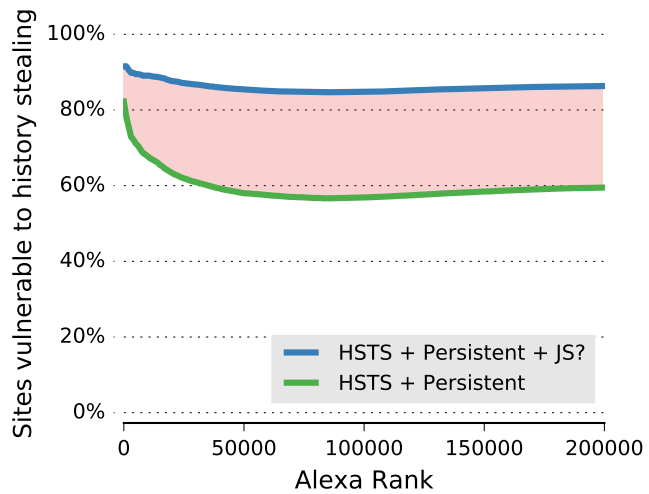


Figure 6.3: History stealing also works, when user is using a VPN for common browsing, as the history is shared between the captive portal and the VPN sessions.

Figure 6.4: Percentage of scanned sites that are vulnerable to our history stealing attack in the Alexa Top 200,000, grouped by steps of 1,000 pages (cumulated). The lower bound represents all pages that either use HSTS or set a persistent cookie. The upper bound additionally includes cookies set through Javascript (includes session cookies). The real number of vulnerable sites is somewhere in the highlighted area.



HTTP status code [16, Section 10] indicates an open Internet connection, anything else a captive portal. The attacker can test if the circumvention was successful by testing for the User-Agent string of the lightweight browser (and other minor differences [249]).

### 6.5.2 Circumventing the Connectivity Test on iOS

Apple iPhones use a very similar technique to Android. Apple's Captive Network Assistant (CNA) downloads URLs with a known content such as `http://captive.apple.com/hotspot-detect.html` or `http://www.apple.com/library/test/success.html`. The list of tested URLs changed significantly between different versions of the operating system and expanded considerably in later versions. However, the CNA uses a very distinct User-Agent string ("wispr") which has proven to be a good indicator [7, 8, 283].

## 6.6 Assessment of Applicability

To get an impression of the impact of our cookie-based history stealing attack, we analyzed how many webpages actually either use persistent cookies or set up HSTS on their servers. Therefore, we crawled Alexa Top 200,000 websites with a headless browser and recorded the network traffic.

There are two ways how a cookie can be set, either through the `Set-Cookie` header or by using Javascript's `document.cookie` property. Furthermore, a persistent cookie needs to have an expiration date set which lies in the future<sup>1</sup> otherwise the cookie will be deleted as soon as the browser session is closed. Additionally, for the cookie stealing attack to work, the secure flag of the cookie must not be set.

<sup>1</sup>Past dates are used to delete such cookies prematurely.

### 6.6.1 Lower Bound Estimation

Cookies set directly through the server are easy to detect and filter according to the above mentioned criteria. They can be observed in the server response without the need to evaluate Javascript. However, they only determine the lower bound of pages vulnerable to our history stealing attack.

### 6.6.2 Upper Bound Estimation

For an upper bound estimation we also included cookies for which we did not observe the `Set-Cookie` header, but the cookie appeared in a request from the browser to the server. We removed all session (non-persistent) cookies that were set by the server. However, the remaining set will include persistent and non-persistent session cookies set by Javascript. The persistence of a cookie is not indicated by the browser in the requests sent to servers. Therefore, the upper bound overestimates the number of sites by those which use Javascript to set session cookies, but no persistent cookies.

A short manual inspection showed that many of the Javascript cookies we observed are set by tracking scripts like the `__utma` persistent cookie set by *Google Analytics*<sup>2</sup>. Therefore the real numbers are more likely closer to the upper than the lower bound.

Additionally, the crawling technique might have missed cookies set under rare conditions (such as sub-pages, configuration, tracking opt-out cookies) making the real numbers even higher.

### 6.6.3 Methodology

We set up headless browsers on Amazon EC2 Spot instances to visit the Alexa Top 200,000 websites in an attempt to determine the pervasiveness of persistent cookies.

For every website, we crawled the main page and three randomly selected sub-pages and collected the corresponding responses. We then extracted the headers from each of the requests and searched for the `Set-Cookie` headers. If these headers are found in one of the responses, the corresponding expiry date is set to a future date<sup>3</sup>, and the `secure` flag is not set, we count the page as setting a persistent cookie. Furthermore we also look for the `Strict-Transport-Security` header to see if HSTS is enabled for the page. For the bounds calculation, entries in the HSTS pre-load list used by several browsers were excluded from the set of possibly vulnerable pages.

### 6.6.4 Results

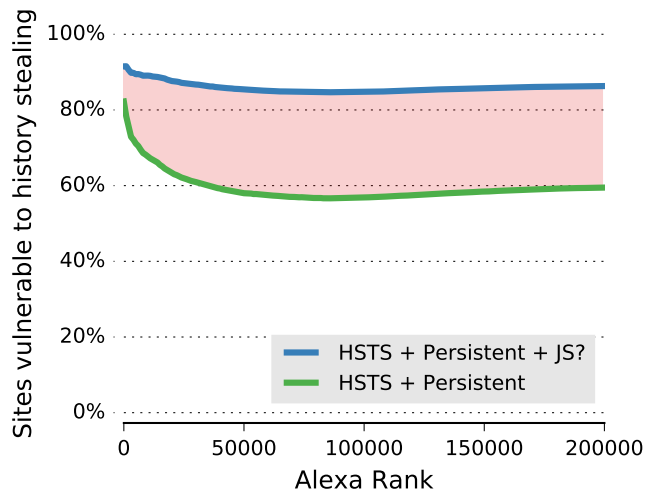
As the graph in Figure 6.5 shows, even without considering cookies set through Javascript, on average about 59.47% of all pages are either using HSTS or are setting at least some

---

<sup>2</sup><https://developers.google.com/analytics/devguides/collection/analyticsjs/cookie-usage?hl=en> (Accessed: 2016-01-29)

<sup>3</sup>includes the *Expiry* and the *Max-age* option

Figure 6.5: Percentage of scanned sites that are vulnerable to our history stealing attack in the Alexa Top 200,000, grouped by steps of 1,000 pages (cumulated). The lower bound represents all pages that either use HSTS or set a persistent cookie. The upper bound additionally includes cookies set through Javascript (includes session cookies). The real number of vulnerable sites is somewhere in the highlighted area.



kind of persistent cookie. For the top 1,000 pages this persistent cookie usage spikes up to 82.24%. As mentioned before, these results only indicate a lower bound. If we also count cookies that were set through Javascript, the numbers for vulnerable pages rise to 91.52% for the top 1,000 pages and 86.31% on average.

These results clearly indicate that there exists a large attack surface for the captive-portal-based history stealing attack.

## 6.7 Proof-of-Concept Implementation

We created a proof-of-concept privacy-invading Wi-Fi hotspot that might also be used for raising awareness in educational settings or to perform public demonstrations (cf. [54]). It implements the history stealing attack as well as several additional attacks and provides a per-user result page showing in plain language which data has been automatically collected.

Figure 6.6 gives an overview of the implementation structure. It runs on a virtual machine connected to an USB Wi-Fi adapter<sup>4</sup> in access point (AP) mode. A set of Python scripts extends a Wi-Fi Router/NAT iptables setup.

HTTP traffic is diverted to a man-in-the-middle (MITM) proxy that can intercept and manipulate passing HTTP requests. This is the main component of the history stealing attack (see Section 6.7.1).

To further demonstrate other risks of public hotspots, we also added a small number of additional attacks. A number of passive attacks is implemented by simple packet sniffing on unencrypted non-HTTP protocols, such as plain-text passwords or DNS-snooping

<sup>4</sup>TP-LINK TL-WN722N

(see Section 6.7.2). This includes the detection of installed applications based on simple network patterns, deriving the full name and gender of the user from the device name or through third-party websites (using the captured cookies). We also added active exploits for security vulnerabilities (e.g., WebView) by injecting code into the HTTP traffic.

### 6.7.1 Implementation of History Stealing

In principle, there are many ways to compel a browser to make external requests. We tested three methods: Javascript XML-RPC requests, IFrames, and IMG tags. The latter turned out to be the fastest. It is lightweight and facilitates the browsers ability to parallelize requests. These requests are inserted into the portal webpage and/or into other HTML pages loaded via HTTP. These induced requests are detected by the MITM proxy based on marker strings within the request (e.g., as part of the URL). The mitm-proxy collects these requests and answers them locally. The latter serves two purposes: (a) it speeds up the requests and (b) it prevents the client from collecting additional cookies through the probing requests.

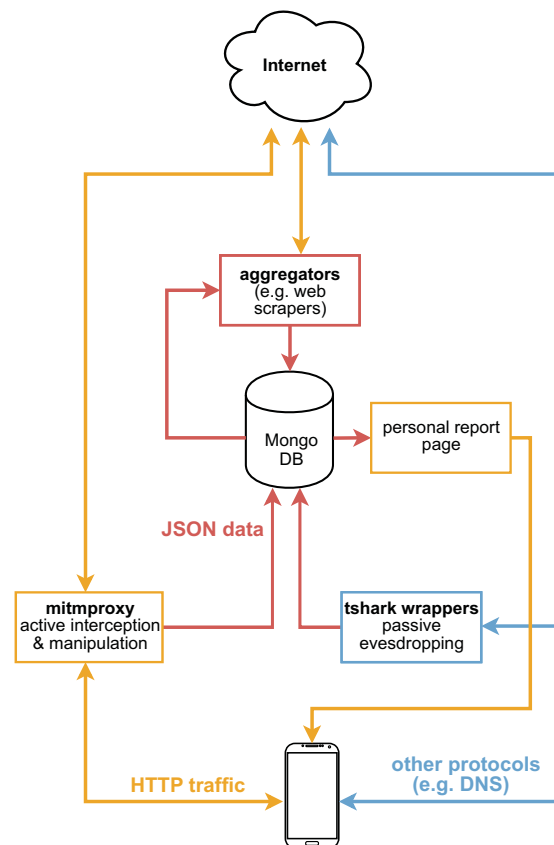


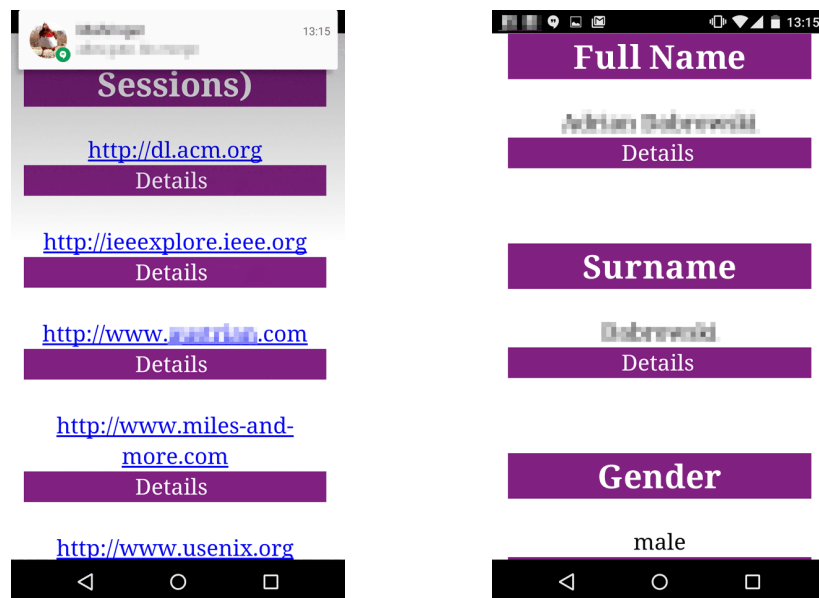
Figure 6.6: Structure of the implementation

### 6.7.2 Further Attacks by Malicious Hotspots

For Android phones we implemented sample exploits based on the WebView component [61, 224]. WebView is a UI element which allows developers to simply display local or remote HTML content. It offers a Javascript application bridge that allows arbitrary calls to public functions in the application, including *Java Reflection*. It is still present in today's applications if they are compiled for Android versions below 4.3. One of our test scripts steals the latest photos shot with the phone's camera, while another one lists files from the download folder. Additionally, a network traffic sniffer extracts the listened-to music titles from Shazam's [13] traffic.

### 6.7.3 Personal Data Enrichment

Based on the data acquired by the MITM proxy and passive sniffing, small demonstration scripts try to enrich the dataset with additional information from other external sources. Amazon's public profile feature can be used to acquire the user's full name, even if she or he is not actively logged in, since it is enabled by default. Subsequently we use the name to determine the gender of the user via a public demographic database API.



(a) Stolen history: Sites the user visited before connecting to the demo hotspot.

(b) The full name was derived via a Amazon plain-text cookie, and the gender via an analysis of the first name.

Figure 6.7: Report screenshots

### 6.7.3.1 Amazon Name Disclosure

Amazon.com has three states for user sessions: A user can be either *logged in*, *half logged in*, or *not logged in* at all. The half-logged-in state is a specialty of Amazon. Amazon recognizes – with high probability – based on an earlier long term plain-text cookies a particular user even if the user’s session has expired. Amazon allows the user to add products to the cart, but will ask for the user’s password before any further action. By requesting `http://amazon.com/gp/profile` (or the appropriate localized Amazon site) a public profile including wish lists, reviews, and (by default) the full name of the user can be accessed. A user can turn off the visibility of most elements in the account settings.

### 6.7.3.2 Gender Estimation

Genderize.io [3] offers a simple service to look up the gender of a person based on their first name. We display this information in the user report.

### 6.7.3.3 Device (and User) Name

During setup, iOS suggests to use *[Firstname]’s iPhone* as the device name. This name is broadcast on the Wi-Fi using mDNS as part of the zero-config peer discovery protocol *Bonjour*. It offers another way to get hold of a user’s first name.

### 6.7.4 Device Parameters

During the captive portal phase – and later due to HTTP injection – the browser can be tricked into loading and executing arbitrary Javascript code. They can read out device parameters (model and brand, screen size, installed extensions), but also track a user’s site visits (scrolling, clicking).

### 6.7.5 Installed Applications based on Network Traffic

Some ad networks and most instant messengers regularly connect to their *home* servers. In the latter case, this is necessary to ensure that incoming instant messages are delivered over the current Internet connection. Therefore, most such applications subscribe to system messages indicating a change in the network connectivity. Connecting to a Wi-Fi network (like our malicious one) is such an event. While many instant messengers switched to encrypted traffic, the target of the traffic (and the plain text DNS lookup before that) give away enough information to identify the application<sup>5</sup>. We included a few handwritten rules that identify WhatsApp, Facebook Messenger, Instagram, Snapchat, Skype, and several other services. Other works, such as NetworkProfiler [96] and AppScanner [286] could easily enhance this detection.

---

<sup>5</sup>This also works in many cases where the application is using a cloud messaging service such as Google Cloud Messaging (GCM), because it needs to communicate the GCM token to its home server.

### 6.7.6 Report Page

The report page sorts the collected information by its technicality and privacy invasiveness. Thus, it presents high-level information (e.g. the user's real name) at the top, and more technical details (e.g. history) below. This allows a non-technical user to read the important information first.

## 6.8 Limitations and Future Work

One of the major speed limitations of our implementation is the DNS lookup round-trip time that each new site requires. We can pre-load them into the cache on our hotspot, but not to the client. For the browser, each reference needs to look like a new site. Depending on the used mobile phone the attack performs up to roughly 50-100 history tests per second. Several optimization ideas (such as prematurely killing connections as soon as data is collected) are left for future work.

We are currently injecting history stealing references into the portal and into HTTP traffic. However, we do not catch HTTPS connections and therefore cannot detect cached HSTS entries. Additionally, we experimented with SSL-stripping and SSL-proxying, however, the failure rate was so high that we are currently not using it. It would require extensive white- or black-listing approaches to only affect sites and apps that do no or insufficient certificate validation. We see this as a positive sign of increased awareness among developers.

A limitation of our current hardware setup is the usage of the *TL-WN722N* Wi-Fi stick as an access point. In general, the chipset has good Linux support, but the firmware is limited to seven clients in access point mode.

Since our implementation is meant to evolve to an educational or demonstrative tool (e.g. for school children, students, or not so technology versed people) we like to add more automated aggregation scripts and app reconnaissance mechanisms such as NetworkProfiler [96] or AppScanner [286]. This should serve the goal of increasing awareness regarding the risks of public Wi-Fi hotspots.

## 6.9 Conclusion

History stealing is one of the most privacy-invasive attacks. Several methods have been developed in the past, most of which have to probe each individual URL and are now mitigated by current browsers.

We described a site-level history stealing attack which is easy to perform by Man-in-the-Middle attackers and (even more easily) by operators of captive portals, such as those found in many public Wi-Fi hotspots. Captive portals enjoy a privileged network position easing the implementation of the attack. It affects notebooks, mobile phone, and tablet users alike. The attack is difficult to prevent even for VPN users.



The attack operates by supplying the client browser with a vast numbers of external references (e.g., images) which it will try to download. It is not necessary to filter out these requests at the gateway, but it speeds up the attack. During these requests, the browser will use cookies from its persistent cookie database. A presented cookie for a particular domain is a clear indicator that the user has visited that site before. We also described how a cached HSTS entry reveals a user's past visit.

We further investigated the prevalence of such long-term cookies to estimate the attack surface. Out of the Alexa Top 1,000 sites, between 82.24% and 91.52% are affected (depending on how they set their cookies). The rate approaches 59.47% to 86.31% for the Alexa Top 200,000 dataset.

As a proof of concept we implemented this attack together with several other privacy-invading attacks to create an environment where participants can educate themselves about data leakage in untrusted Internet environments. Additional scripts enrich this data with information from other sites (e.g. Amazon's real name disclosure). It is meant as a tool to create awareness regarding privacy issues for people without a technical background.

As an immediate countermeasure, we encourage web developers to switch all their sites to HTTPS, optionally add them to the HSTS preload list, and to only use secure cookies. Additionally, all operating systems should offer a stripped-down browser to log on to captive portals with a less predictable connectivity check.



# Introducing Decoding Ambiguity with Error-Correcting Codes

Data decoding, format, or language ambiguities have been long known for amusement purposes (e.g., programming languages polyglots). Only recently it came to attention that they also pose a security risk [38–40, 161, 199].

In this chapter, we present decoder manipulations based on deliberately caused ambiguities facilitating the error correction mechanisms used in several popular data transfer methods and applications, including mobile phones. This technique can be used to encode data in multiple formats or the same format with different content within the same data stream or packet. Implementation details of the decoder or environmental differences decide which data the decoder locks onto. This leads to different users or application layers receiving different content based on decoding ambiguities.

In general, ambiguity is not desired, however in special cases it can be particularly harmful. Format dissectors can make divergent decisions, e.g. a firewall scans based on one format but the user decodes a different harmful content.

We implement and demonstrate this behavior with popular 2D barcodes and argue that it can be used to deliver exploits based on the software installed, or use probabilistic effects to divert a small percentage of users to fraudulent sites. This implementation remotely resembles Packet-in-Packet attacks on radio devices and protocol decoding mismatch in network protocols as shown by Goodypeed et al. [136].

This chapter is an extended version of several papers [83, 87].

The rest of this chapter is structured as follows: In Section 7.2 we discuss the connection to language security and the necessary background on ECC and polyglots. In Section 7.3 we show how they can be combined and in Section 7.4 and 7.5 we present a proof-of-concept implementation based on barcodes and their evaluation with different decoders.

*Future Work* (Section 7.6) and the conclusion in Section 7.7 discuss the implications, countermeasures and open problems.

## 7.1 Overview and Context

Albertini [38] and Magazinius et al. [199] researched so called *binary polyglots* that are for example valid PDF, JPEG, and ZIP files at once [39]. In contrast to computer language polyglots (source code) which compile in different programming languages, these are binary formats carefully stuffed together to conform to multiple file standards or at least be understood by those file parsers.

However, ambiguity is in general not desired. Especially when it leads to different decoders reading different content, e.g. a firewall scans a JPEG but the user decodes a ZIP archive with harmful content out of the same data stream. Jana et al. [161] and Alvarez et al. [40] have shown how to abuse file-type fingerprinting and parsing differences of anti-virus tools to evade detection.

On a more theoretical level polyglots lead to a language decoding ambiguity. This gets even more interesting when a decoder supports multiple languages but makes a hard decision on reading and interpreting it one way or another. Any ambiguity is therefore a potential security risk [40, 161, 268].

In this chapter we show that error-correcting codes (ECC) are a convenient tool to construct and provoke such ambiguities.

## 7.2 Background

For this chapter we will use *file format*, *data format* (e.g. on a network) and *language* interchangeably. After all, a file format specification defines semantics, symbols, and a grammar and therefore a language. A parser by itself also implicitly defines a language that under best circumstances is equivalent to the specification or at least contains the specification as a subset or substantially overlaps it.

Likewise, a *file*, a *transmission*, a *data stream*, or a single *data packet* are also the same on an abstract level: data arrives from an external source and has to be decoded and/or parsed (often multiple times, based on layered network and software structure) to be turned into an useful (internal) representation. This includes storage of data and transmissions over wire, radio, optical networks, or visual symbologies (e.g. barcodes, as used in this chapter).

### 7.2.1 Binary Polyglots

In computing, a polyglot used to be a source code valid in multiple programming languages. However, this is easily extendable to binary formats, as each parser forms its own language.

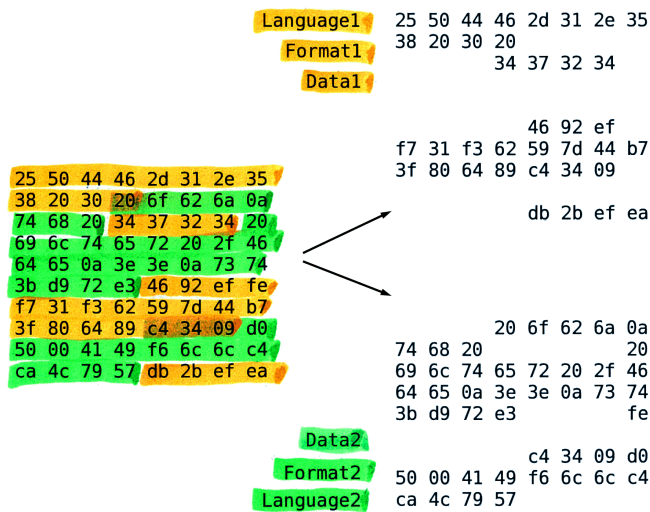


Figure 7.1: Illustration of a binary polyglot: data is interleaved using references with unused space in between, reserved or unused fields, etc.

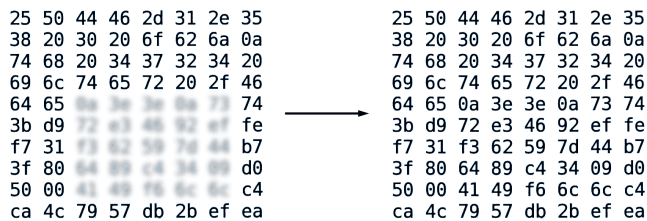


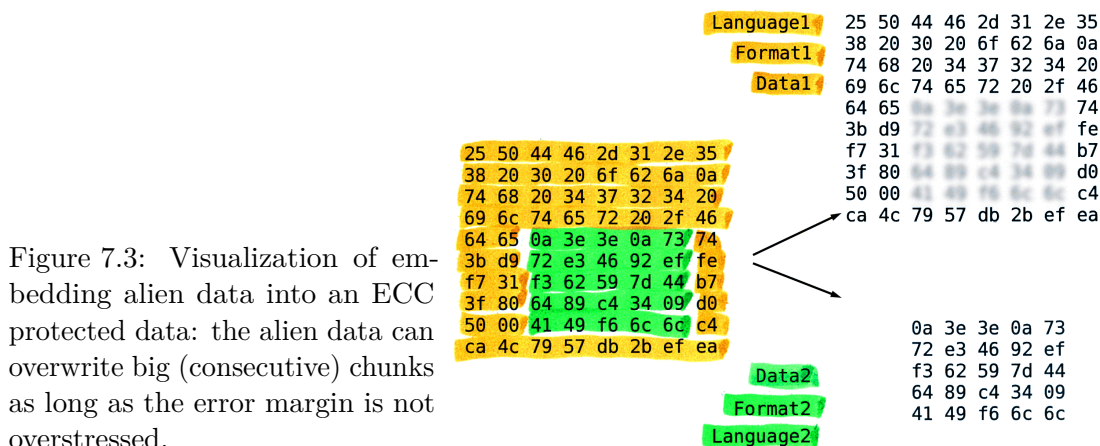
Figure 7.2: Visualization of an error-correction code: the destroyed data is reconstructed.

As seen in the visualization (Figure 7.1), some parts of the file are used exclusively for one format (language) or another, whereas other parts can be shared. This stuffing is made possible by the different format semantics, loosely validating parsers, lax or ambiguous format specifications, and extensive use of *comment blocks* as well as *ignored*, *reserved*, and fields *for future use*.

This way, for example, a file can simultaneously be a valid PDF document, JPEG picture, and ZIP archive. Typically, parsers start their work by finding an identifier and proceed from there. In above example, JPEG needs a correct signature string right at the start, PDF within the first 1024 bytes, and ZIP starts decoding archives from the end. (Binary) polyglots can be build as academic challenge, or to deceive file type dissectors and bypass scanning or filtering.

## 7.2.2 Error Correction

Forward error correction (FEC) is a technique used to cope with errors in data transmission over unreliable or noisy communication channels without the need for a reverse channel. The sender encodes their message in a redundant way by using an error-correcting code (ECC). Thus allowing the receiver to detect and correct a limited number of errors that may occur anywhere in the message without retransmission. Simple codes such as



Parity and Hamming codes can only detect and correct a very limited number of errors, advanced codes such as Reed-Solomon, MDS codes, or Turbo Codes provide versatile configuration options. For example, on noisy channels, the code can be configured to allow 30% of the data to be damaged without an impact on consistency.

The model behind these techniques is a noisy transmission line, random physical errors on a data carrier, or interference in transmission. They have not been designed to withstand a crafted attack. In their effort to protect and reconstruct the original data they can be abused up to a point where (part of) the data can mean something completely different.

Extensive error correction is used in many applications such as digital radio (DAB) and video transmission (terrestrial and via satellite, DVB-T, DVB-S, DVB-C), cellular phones (GSM, UMTS, LTE), wireless networks (WIFI), digital tapes, hard discs, optical discs, raid arrays, flash drives, cloud storage, server RAM, and barcodes. We will use barcodes for demonstration purposes.

### 7.3 Error Correction as a Hideout

For the rest of the chapter, ECC can be viewed as configurable black boxes for encoding data into code words and vice versa. Among other parameters, the configuration determines the recovering abilities (error margin); i.e. the fraction of destroyable code words without an effect on the decoded data. In normal operation the modified code words remain transparent to the reading application as error correction is typically done as one of the first steps in acquiring data (Figure 7.3). The amount of foreign data is limited primarily by the error recovering abilities of the used code. It is not dependent on the actual type of code used (e.g. Turbo Code, Reed-Solomon).

We can utilize the latter to override chunks with parasitic data at almost freely chooseable places, as long as the error margin is not overstressed and no other vital information is harmed (e.g. ECC-header, synchronization). This includes additional synchronization

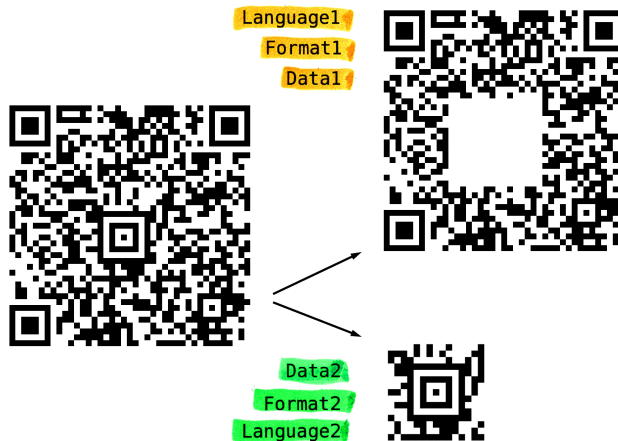


Figure 7.4: Similar to Figure 7.3 a Barcode-in-Barcode can be constructed. Note: both parts are valid barcodes.

patters and headers if necessary. The parasitic data can be of a different language or the same language as the carrier data. The former will exploit multi-language synchronizers, dissectors and parsers which have to decide on the type of data. The latter will primarily exploit the inability of simple synchronizers to distinguish between different data streams.

This remotely resembles Packet-in-Packet attacks on radio devices and protocol decoding mismatch in network protocols [136]. In contrast to Packet-in-Packet attacks, we do not have to modify the content of the user data. The latter is transparently restored by the ECC mechanism.

## 7.4 Implementation using Barcodes

While the visualization in Figure 7.3 might look far-fetched or abstract, we can do *exactly that* with barcodes (Figure 7.4).

*Linear barcodes* or *1D codes* are used to provide a machine-readable form of printed information. In cases where higher data density is required, *matrix* or *2D barcodes* are preferentially deployed. Such codes are used in industrial applications, e.g. logistics or tracking of individual components during the production process.

There are various *2D* or *matrix barcode* symbologies. Each of them tends to be dominant in one or more particular fields of application. This makes it necessary for many devices to support more than one standard. An optical transmission media (printed barcode scanned with a camera under imperfect conditions) is subject to noise, distortions, camera artifacts, uneven illumination, and other effects. Thus, these symbologies typically employ multiple strategies to cope with them, one of which is the extensive use of *error-correcting codes*.

In everyday life, electronic tickets are issued with 2D barcodes, and web links are conveyed via 2D barcodes on billboards and in printed ads. Additionally, they are used in security-sensitive applications such as monetary transactions: Paypal and Bitcoin allow shoppers

Figure 7.5: Popular 2D Barcodes with rectangular pixels: Quick Response, Aztec, Data Matrix



to pay for goods and services using apps that generate QR codes readable by merchants' existing scanning devices [285]. Threema [291] uses QR codes to exchange keys and authenticate users.

The application of such codes is not without security risks: Different ways of using QR codes as an attack vector have been proposed [100, 175, 176]. In 2012, hackers showed that *Unstructured Supplementary Service Data* (USSD) codes encoded in 2D barcodes can be used to wipe a phone or execute other system functions [219]. On some phones, they can be used to generate premium rate SMS messages. QR codes can also be used to trigger vulnerabilities in the reader software, the operating system, or a remote site such as SQL injections [176]. The *Ninjhax* exploit [250] uses a custom QR code to perform a buffer overflow on the (locked down) Nintendo 3DS portable game console allowing it to install custom software. Peng et al. [243] found code injection vulnerabilities in several QR libraries. QR Codes are also used to spread malware [100] and for phishing attacks.

For our proof-of-concept implementation we used three popular code types with rectangular pixels, thus ensuring a uniform visual appearance when used together: Quick Response (QR) [10], Aztec [160], and Data Matrix (DM) [9]. All codes employ ECC codes, but for practical reasons (Section 7.4.3) QR codes are suited best for hosting alien data. With the exception of Aztec, all standards mandate a white space (quiet zone) around the symbol, but our tests have shown that most decoders require much less if not none.

A crafted 2D barcode that conforms to multiple standards (or an embedded barcode inside another) is hardly detectable by an untrained human viewer.

#### 7.4.1 Full Scan: Multiple Standards Ambiguity

The primary case we are discussing here is that of a full scan. The decoder is presented with a choice of different codes within the same area. Decoding software usually employs multiple computationally cheap finders for specific symbologies, e.g. a detector for a specific visual marker of a symbology (Figure 7.6). In other contexts, this is called *synchronization pattern*, *preamble*, *magic value*, or *format signature*. When one is found, an appropriate decoder retrieves the data and presents it to the user or the calling application. Although the symbologies are standardized, the dissector decision tree (and its detection order) is not.



### 7.4.2 Partial Scan

A partial scan (e.g. the user trying to find the right angle and distance for scanning) makes it highly probable that an (*inner*) embedded code is inside the imaging frame before the full *outer* (or *host*) barcode, favoring the detection of the inner barcode.

### 7.4.3 Embedding Criteria

Embedding one code into another requires distinct characteristics of the standards for the outer as well as the inner code.

The outer code has to (a) provide a continuous area of a certain size to shelter the other, and (b) a sufficiently robust data correction (or another way to include alien data).

QR and DataMatrix provide a relatively large continuous area to hide other codes. In our tests, QR's error correction performed much better than Data Matrix's. It is configurable in 4 steps from 7% to 30% (i.e. 30% of the data can be destroyed and still reconstructed). Therefore, currently, the QR symbology provides the best host platform to embed other codes. As versatile as QR's error correction code is, not all parts are protected equally. Some elements are vital and needed before the FEC bits can even be read or applied. Therefore, the embedded code must not interfere with these elements (Figure 7.8):

#### 7.4.3.1 Finder or Location Markers

These visually prominent markers (including the quiet zone around them) are used by the detector to locate a barcode in an image and correct possible distortions.

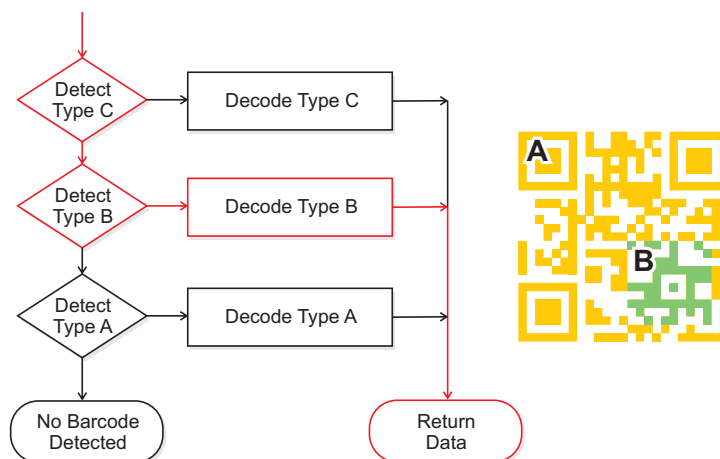


Figure 7.6: Decoding ambiguity: the detector for a particular code is tested first, therefore the others are not considered.

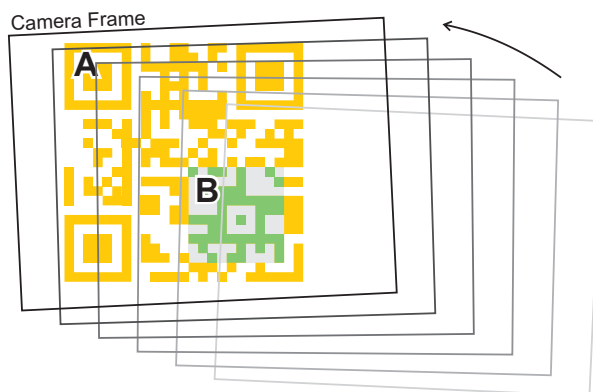


Figure 7.7: Sliding over the barcode will make the smaller inner barcode fully visible before the entire (outer) barcode.

#### 7.4.3.2 Quiet Zone

The QR standard defines a large white space around each barcode. Most readers still require at least 1 pixel white border around the location pattern, whereas some also manage without a quiet zone.

#### 7.4.3.3 Timing Patterns

These dotted patterns run horizontally and vertically between the inner corners of the three location markers. They are used to synchronize rows and column pixels and are essential for most readers.

#### 7.4.3.4 Alignment Markers

They are only built into bigger codes to help handling distortions. They are less important for most decoders and a limited number of them can be destroyed without reducing readability.

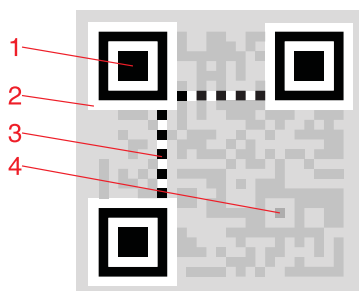


Figure 7.8: Critical areas of an QR Code: location markers (1), quiet zone (2), timing pattern (3), and alignment markers (4).

## 7.5 Experimental Results

For the evaluation of our attack we selected the most popular barcode reader applications in the Google Play Store as well as iTunes App Store. The apps were chosen based on popularity and whether they support multiple 2D barcode standards (Table 7.1). We tested combined barcodes with 5 applications for iPhone and 7 for Android which were all available for free. The goal was to test the applicability, not to provide a market analysis. Furthermore, we tested a professional handheld device as used in retail and logistics.

The chosen handheld decoder was a Motorola(Symbol) DS6708 scanner. *Symbol Technologies* (acquired by Motorola in 2007) is the long term leader in handheld scanner devices and predates the mobile phone ecosystem by decades. The device has been reconfigured to enable all supported symbologies.

Measurements have been conducted with an iPhone 4, an iPhone 5, a Samsung Galaxy Nexus, a LG Nexus 4 Android phone, and above-mentioned handheld scanner under normal office illumination. For each barcode we made at least 10 scanning attempts – more if the results were inconclusive. We varied the distance and rotation between the

Table 7.1: Tested applications and their barcode standard support, as well as other features (as of 2015).

OS/Type	Name	QR	Data Matrix	Aztec	Auto-load URLs	Show barcode
iPhone	NeoReader [221]	✓	✓	✓	✓	✗
	Qrafter [174]	✓	✓	✓	✗	✗
	i-nigma [28]	✓	✓	✗	✗	✗
	QR Code Reader and S. [277]	✓	✓	✓	✓	✗
	ScanLife [270]	✓	✓	✗	✗	✗
Android	ZXing Barcode Reader [344]	✓	✓	✗	(✗) <sup>1</sup>	✓
	UberScanner [298]	✓	✓	✓	✗	(✓) <sup>2</sup>
	ScanLife [271]	✓	✓	✗	✓	✗
	i-nigma [29]	✓	✓	✗	✗	✗
	AT&T Code Scanner [49]	✓	✓	✗	✓	✗
	NeoReader [222]	✓	✓	✓	✗	✗
	ShopSavvy [278]	✓	✓	✗	✓	✗
Handheld	Symbol DS6708 [215]	✓	✓	✓	–	–

<sup>1</sup> Retrieves URL in background to extract page title    <sup>2</sup> Picture excerpt without bounding box

printed barcode and the sensor.

We refrain from giving percentage numbers of decodes, as exact reproduction depends on multiple parameters, such as illumination, angle, movement, distance and so on. Instead we documented if and how each barcode decoded, and if there was a major preference for one or another behavior.

While iPhone readers display a large variation of behaviors, a large share of Android apps use the free *Zebra Crossing* (ZXing) [12] barcode library for scanning. Thus, Android applications show a much more uniform behavior. Differences are minimal and are caused by different versions of the used library included. For example, the *ZXing Barcode Reader* (the demo app for the open source ZXing library) does not support Aztec codes. However, *UberScanner* uses a (newer) beta version of the same library and does supports them.

### 7.5.1 Aztec in QR

Aztec is a very good choice for being embedded into another code. By standard it does not require a quiet zone. However, our tests have shown that corner placement (and therefore offering a partial quiet zone) provides a higher decodability rate with the Symbol device.

Qrafter was neither able to decode the inner nor the outer barcode, while NeoReader strongly prefers the inner Aztec code. This is probably a case where the Aztec finder is called before the QR finder. Although tempting, it is not possible to use both finder patterns to overlay each other. The Aztec finder pattern is always centric and a bit taller than QR's. In our tests, QR readers could not be made to accept an Aztec finder pattern as one of the three QR finder patterns or vice versa.

Orientation and placement of the pattern did not play a significant role in decodability.



App/Device	Outer	Inner
NeoReader	✓	✓pref.
Qrafter	✗	✗
i-nigma	✓	–
QR Code R.S.	✓	✗
ScanLife	✓	–
ZXing B.S.	✓	–
UberScanner	✓	✓
ScanLife	✓	–
i-nigma	✓	–
AT&T Code S.	✓	–
NeoReader	✓	✓
ShopSavvy	✓	–
DS6708	✓	✓

Figure 7.9: Aztec in QR: NeoReader on iOS strongly prefers Aztec over QR

### 7.5.2 Data Matrix in QR

The weakness of Data Matrix is the lack of a distinct visual marker. On the one hand, this makes the code very compact, on the other hand the decoder gets fewer visual clues.

In the first two experiments, we hid the Data Matrix in the bottom right corner of the host QR code without any white space around them (Figure 7.10). Most readers did not detect the embedded inner code. For NeoReader, the orientation of the Data Matrix code proved to be of importance: a version of the Data Matrix code with the solid line facing the QR's outer border was detected (Figure 7.11). Interestingly, i-nigma had problems reading the outer QR code: the iOS version did not decode at all and the Android version decoded only after numerous attempts.



App/Device	Outer	Inner
NeoReader	✓	✗
Qrafter	✓	✗
i-nigma	✓	✗
QR Code R.S.	✓	✗
ScanLife	✓	✗
ZXing B.S.	✓	✗
UberScanner	✓	✗
ScanLife	✓	✗
i-nigma	✓	✗
AT&T Code S.	✓	✗
NeoReader	✓	✗
ShopSavvy	✓	✗
DS6708	✓	✗

Figure 7.10: Data Matrix (bottom right) in QR



App/Device	Outer	Inner
NeoReader	✓	✓
Qrafter	✓	✗
i-nigma	✓	✗
QR Code R.S.	✓	✗
ScanLife	✓	✗
ZXing B.S.	✓	✗
UberScanner	✓	✗
ScanLife	✓	✗
i-nigma	(✓)	✗
AT&T Code S.	✓	✗
NeoReader	✓	✓
ShopSavvy	✓	✗
DS6708	✓	✗

Figure 7.11: Data Matrix (bottom right, rotated) in QR

In a second test, the embedded Data Matrix code was positioned in the center of the QR

code: first without a white border (Figure 7.12), and then with a white border (Figure 7.13). The former was not detected by any reader. The latter was decoded by almost all scanners, whereas NeoReader on iOS completely ignores the outer QR code. On Android, ScanLife and the AT&T Scanner only decoded the inner Data Matrix when panning over the image.



App/Device	Outer	Inner
NeoReader	✓	✗
Qrafter	✓	✗
i-nigma	✓	✗
QR Code R.S.	✓	✗
ScanLife	✓	✗
ZXing B.S.	✓	✗
UberScanner	✓	✗
ScanLife	✓	✗
i-nigma	✓	✗
AT&T Code S.	✓	✗
NeoReader	✓	✗
ShopSavvy	✓	✗
DS6708	✓	✗

Figure 7.12: Data Matrix (center) in QR




App/Device	Outer	Inner
NeoReader	✗	✓
Qrafter	✓	✓
i-nigma	✓	✓
QR Code R.S.	✓	✗
ScanLife	✓pref.	✓
ZXing B.S.	✓	✓
UberScanner	✓	✓
ScanLife	✓	(✓swipe)
i-nigma	✓	✓
AT&T Code S.	✓	(✓swipe)
NeoReader	✓	✓
ShopSavvy	✓	✓
DS6708	✓	✓

Figure 7.13: Data Matrix (center, white space added) in QR

### 7.5.3 QR in QR

QR in QR is a special case. The finder markers compete against each other and may strongly confuse the detector. Additionally, it is easier to be noticed by a human. In this case, the camera frame and angle of rotation can be significant for the software's decoding decision. The results also indicate that some finder pattern algorithms require



App/Device	Outer	Inner
NeoReader	✓	✗
Qrafter	✗	✗
i-nigma	✓	✓
QR Code R.S.	✗	✗
ScanLife	(✓rot.)	✓
ZXing B.S.	✗	(✓swipe)
UberScanner	✗	(✓swipe)
ScanLife	✗	✗
i-nigma	✓	✗
AT&T Code S.	✗	✗
NeoReader	✓	✗
ShopSavvy	(✓)	✗
DS6708	✓	✓pref.

Figure 7.14: QR in QR, corner, w/o white space

a white space around the marker, and some do not. However, the white space around the whole barcode as defined in [10] is not a necessity for any of the tested readers.

For this series of tests, we increased the level of FEC for the outer barcode, as the inner barcode consumes considerably more area than Aztec or Data Matrix with the same content.

In the first case, as depicted in Figure 7.14 (QR put in a corner, without additional white space) a significant number of readers had problems decoding any of the QR codes. ShopSavvy decoded very rarely. For the inner code, most of the ZXing based software picked it up panning slowly over the barcode in the moment when the top left corner of the camera frame is aligned with the barcode corner. The DS6708 strongly preferred the embedded code, while i-nigma on iOS was indifferent. ScanLife on iOS picked up the outer barcode only after a significant rotation.

In the case where the embedded code is not exactly in the corner, the recognition matrix changes slightly (Figure 7.15). i-nigma decodes the inner code slightly better when rotated 45° and slide into the image, while the alignment trick does not work for ZXing-based readers. On Android, i-nigma only decodes the inner QR when it is embedded in the center.

Adding white space around the finder markers of the embedded code (Figure 7.16) increases the readability dramatically, practically disabling the outer code for many applications. Presumably, implementations prefer markers in close vicinity to each other.

Qrafter and ShopSavvy need noticeably longer for decoding, but do so only for the embedded code. i-nigma on Android prefers the outer code when the phone is held further away, and the inner code when held closer to the barcode. QR Code Reader and Scanner on iOS has major troubles with decoding. In our tests it eventually returned the inner code and in one case returned a garbage string.



App/Device	Outer	Inner
NeoReader	✓	✗
Qrafter	✓	✗
i-nigma	✓	✓
QR Code R.S.	✓	✗
ScanLife	✓	✗
ZXing B.S.	✓	✗
UberScanner	✓	✗
ScanLife	✓	✗
i-nigma	✓	✓cent.
AT&T Code S.	✓	✗
NeoReader	✓	✗
ShopSavvy	✓	✗
DS6708	✓	✓

Figure 7.15: QR in QR, semi corner and center, w/o white space



App/Device	Outer	Inner
NeoReader	✗	✓
Qrafter	✗	✓
i-nigma	✓	✓
QR Code R.S.	✗	(✓)
ScanLife	✗	✓
ZXing B.S.	✗	✓
UberScanner	✗	✓
ScanLife	✗	✓
i-nigma	✓	✓
AT&T Code S.	✗	✓
NeoReader	✗	✓
ShopSavvy	✗	(✓)
DS6708	✓pref.	✓

Figure 7.16: QR in QR, center with white space

Presumably, these implementations prefer markers in close vicinity to each other, except for the handheld device by Symbol. The latter prefers the largest area between multiple found markers.

## 7.6 Discussion, Countermeasures and Future Work

We demonstrated the ECC hiding and decode ambiguity problem with popular barcodes. We argue that it can be used to deliver exploits based on the software installed, or use probabilistic effects to divert a small percentage of users to fraudulent sites (e.g. a donation site where some transactions are diverted to a different account). It could also be used for fare-dodging or circle-routing parcels, as different stations along the logistics



chain read different tracking IDs.

More in general, the technique can be used anywhere where ECC is employed. This includes satellite or terrestrial digital video transmission (e.g. DVB-S and DVB-T) where different content is decoded by different viewers. This might also have implications on computer (anti-)forensics when dealing with ECC-protected data (tapes, hard disc arrays, ECC RAM).

The main conceptional problem with countermeasures is that ECC are designed to transparently hide any modifications from the processing layers above. The host data as well as the parasitic data are actually valid and conform to the specifications. Therefore, effective mitigation often heavily depends on the threat model, the application, and whether this case can be escalated to the layers above – potentially reaching a user interactively.

### **7.6.1 Countermeasures for Barcodes**

For the barcode example, several mitigation strategies arise. It should be noted that user involvement is an easy option for interactive applications, but less suitable for automated processes (e.g. sorting machines in logistics).

#### **7.6.1.1 Stringent Priority**

While the code formats themselves have been standardized, the order of detection is chosen by the software designers. As this is the root cause for multi-standard code ambiguity, a stringent prioritization should be defined.

#### **7.6.1.2 Notification on all Codes Found**

Barcode readers detecting the presence of code ambiguity (same standard or multiple) should present all of them to the user. This requires that software does not stop after the detection of the first code.

#### **7.6.1.3 Alien Data Warning**

A reader that detects alien data, multiple standards, or multiple decodings using the same standard should warn the user. This might include standards that it might not be able to decode but is able to detect based on its marker signature. However, this also bears the risk of false positives, as the decoder cannot verify if the marker actually belongs to a valid code.

#### **7.6.1.4 Scanned Photo Excerpt**

Interactive barcode readers can present the decoded image and highlight the area of interest containing the decoded barcode for visual inspection.

### 7.6.1.5 Only Decode What you are Looking for

Limit decoding to only the standards the intended purpose needs.

### 7.6.2 Generic Countermeasures

Generic solutions are hard to come by, as both data transmissions are perfectly valid considering the specification. The threat model determinates whether the decoder should escalate the condition to a higher layer, fail-safe by discarding all data, or try to decide on the benign data by itself.

The latter is not a trivial problem. In our visual QR examples above, dissecting parasitic data from the host transmission is much easier than the general case. Spotting multiple synchronization markers on an asynchronous data channel and verifying their belonging to non-overlapping data packets is potentially a hard problem as it might include decoding and verification of all the possible data interpretations itself.

Another (simpler) heuristic could involve sorting possible decoding variants of the data by its amount of bits (whether before or after ECC) and assuming that the longest data stream is the host and therefore legit.

However, any heuristic (or *guessing*) is a risk, and might lead to exactly the problems described in Section 7.1. These are open problems left for future work.

## 7.7 Conclusion

Data decoding, format, or language ambiguities have been long known for amusement purposes. Only recently it came to attention that they also pose a security risk, for example by deceiving file dissectors of firewalls and virus scanners. These binary polyglots mostly arise from poor parsers, lax data format specifications, and undefined multi-standard compatibility. In contrast to carefully crafted polyglots, data formats protected by error-correcting codes (ECC) provide a very convenient way of constructing decoder or parser ambiguities.

This can be used to encode data in multiple formats or even the same format with different content. Implementation details of the decoder or environmental effects decide which data the decoder locks onto. This leads to different users receiving different content based on a language decoding ambiguity.

We demonstrated this behavior with popular 2D barcodes but the method is not limited to these. However, evasion or mitigation strategies are not easy transferable and mostly application specific.

The main conceptional problem with countermeasures is that ECCs are designed to transparently hide any modifications from the processing layers above. The host data as well as the parasitic data are actually valid and conform to the specifications. Where applicable, stringent language descriptions that include dissection rules for multiple

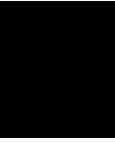
languages/standards and multiple synchronization markers are a good start. In the general case, spotting multiple synchronization markers on an asynchronous data channel and verifying their belonging to non-overlapping data packets is not a trivial task. The area of defenses leaves plenty of opportunities for future work.



**Part IV**

**Individual Picture Privacy**





# Fabric-based Passive Picture Privacy

We put forward a framework to address a problem created by the rapidly spreading use of imaging devices and related to involuntarily or unintentionally photographed individuals: their pictures can accumulate additional meta information via face recognition systems and can be manually tagged via social networks and publishing platforms. With this framework a user can express his/her picture privacy policy in a machine readable format and (to some extent) automatically enforce it. An easily understandable flag system is used to define restrictions on picture usage and linkability. This policy is encoded in an unobtrusive way into wardrobe patterns and accessory designs with almost no impact on apparel appearance or social interaction.

This chapter is an extended and updated version of [95] and parts of [179] from our series of publications about individual picture privacy [88, 165, 180, 198] that have been created over the years in cooperation with the Echizen group at the National Institute of Informatics in Tokyo, Japan and Matthew Smith from University of Bonn, Germany with substantial support from the Internet Foundation Austria (IPA).

## 8.1 Preface

Imaging devices have infiltrated every corner of modern life. They are omnipresent in multiple forms such as photographic cameras, security cameras, and mobile phones. Products like Google Glass [5] have introduced wearable computing devices to the public, potentially enabling the recording of everything at anytime (c.f. Omniveillance [66]). Not only do these devices digitally document the life of the user, they also capture other individuals nearby.

People can feel uncomfortable about losing control over their pictures, and there are serious privacy implications [36] due to the massive publication of private pictures and other information along with them. Face recognition is built into many picture publishing systems such as Picasa and iPhoto and into social networks such as Facebook. Pictures enriched with personal (meta) information (Figure 8.1) can eventually end up in search machine indices, hence providing searchability by name, face similarity, date, and/or geographic location.

Picture rights, privacy, and publishing related problems are not a novelty of the Internet age, but they are amplified by it. Pictures in print media and their privacy implications are also regularly the subject of legal proceedings.

## 8.2 Motivation

Many countries define rights regarding a person's own image. However, they are not easy for a person to enforce. The image of a person might have been unintentionally captured by a photographer without the person noticing that his/her picture was being taken, the person may simply not know the photographer, or the person may not know when and where his/her picture was published and in which context. This lack of knowledge can hinder the person from exercising his/her legal rights. Moreover, the person has no way to inform potential or actual picture takers of their self-chosen restrictions on how their image shall be handled.

Likewise, a conscientious photographer might not have the chance to ask all the people whose image he/she captured for their consent to use their images. In any case, the person's right to control how his/her image is used is lost due to a gap in the communication and control path from the person to the photographer and/or publisher of the photo. Additionally, different countries regulate this right differently: some tie it to the act of publishing the picture while others tie it to the act of taking the picture.

A possible solution to the communication gap problem is to create a central database of privacy policies. Such a database could either use face recognition features or a unique code embedded in each person's clothing as a lookup key. However, this method facilitates

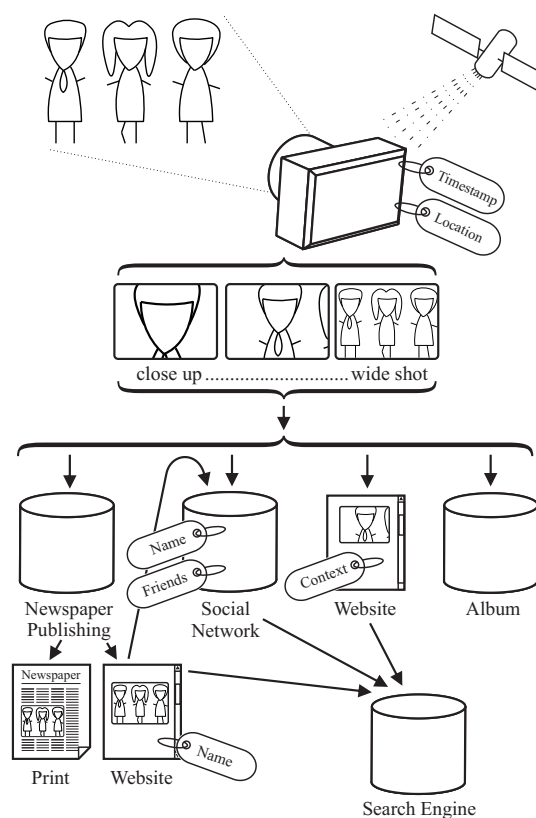


Figure 8.1: Illustration of how a picture can accumulate meta information and end up in various places.



identification and creates a linkability capability that might contravene a person's picture privacy policy. Furthermore, it creates a central database that could be a single point of failure and that could be misused for surveillance purposes.

### 8.3 Contribution

Our proposed *Personal Picture Policy Framework* (P3F) eliminates the gap in communication from the photographed person to the photographer and/or publisher of the person's picture. It incorporates a simple flag-based system that covers the most important restrictions a person might impose on his/her own picture. It is similar to the Creative Commons [2] system used for copyright restrictions on creative works (e.g., by photographers for their pictures).

A modular visual coding system is used to convey the policy information across the communication gap described above. The policy is embedded in the visual information of the photograph (e.g., as part of the clothing), making it an inseparable part of the picture so that it is highly likely to survive along the publishing path (Figure 8.1). Under favorable conditions, this information is hidden in such a way that it is unnoticed by the human eye. Hence, we call it *Privacy Policy Hiding*.

Our proposed formal logic is used to combine multiple policies found in one picture and to determine how to handle potential usage changes when the picture is passed from one entity to another for which other parts of the policy can be relevant (e.g. a picture and its meta information on a social networking site being indexed by a third-party search engine).

Our proposed automated system can be built into publishing software, social networks, and search engines so that they handle pictures appropriately on the basis of the relevant policy (blur out faces of people who do not want their images to be published, discard specified meta information, etc.).

### 8.4 Related Work

The World Wide Web Consortium (W3C) created a *Platform for Privacy Preferences* (P3P) *Specification* [317] to enable automated processing as well as human readable display of web site privacy practices. Since these policies can be quite complex, several authors created simplified human readable iconic representations [60, 102, 146], with Rundle [262] being one of the first. However, as Parsons in his "Privacy Commons" [238] points out, most icon sets lack the visual and semantic clarity and simplicity needed for the broad public. Most attempts have tried to cover too much detail to be understandable to users without an in-depth introduction. To be practical, such a system must focus on the main properties, even if it does not cover all special cases.

Several systems have been proposed for the World Wide Web for minimizing the personal digital footprint. Besides the *Do Not Track* (DNT) header for web browsers currently

being standardized by the W3C [318], there are a number of cookie, tracking, and advertisement blocker extensions for most browsers.

Various methods have been proposed for self-defending an individual's privacy against face recognition. As face mummification is not socially or legally accepted everywhere, several methods attempt to defeat the face-finding algorithms used for pictures. The most common algorithms use Haar-like feature classifiers [316] for computationally lightweight face detection before further processing using more resource-consuming algorithms (e.g., bio-metric identification).

To inhibit the feature response of these algorithms, Harvey uses hair styles and make-up [148], while Yamada et al. [332, 333] uses infrared light sources in a pair of goggles that are visible to most camera sensors but invisible to the human eye. The hair-style and make-up approach is very time consuming in preparation and visually very dominant. It therefore hinders everyday social interaction and can provoke unwanted reactions. The goggles approach is much less intrusive but requires a constant power supply and infrared LEDs that can keep up with the ambient light. Another approach [92] is to bombard the camera with enough infrared light to create a back-lit condition that darkens the rest of the image. This is only feasible at short distances indoors as it is hard to compete against the much stronger daylight.

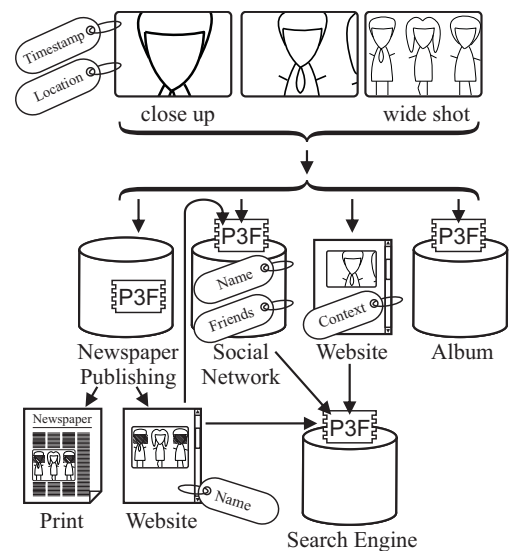


Figure 8.2: Example where P3F sits in the distribution path

## 8.5 Structure and Environmental Considerations

Our proposed P3F sits on neuralgic nodes in the distribution path (Figure 8.2) where it decodes the privacy policies and automatically applies them as described below.

The framework has many constraints to fulfill to be feasible for the user as well as the publishing site or search engine operator. Users need a simple system that is easy to understand and use and that does not hinder them in any of their choices (like their wardrobe style or social interaction). Operators (once they agree or are forced to comply) want a computationally lightweight solution. Technical difficulties arise in finding a coding scheme that can convey a privacy policy whether it is photographed as a close-up or wide shot. Furthermore, the coding scheme has to fulfill the aesthetic needs of the user as well as constraints due to work or social etiquette.

Table 8.1: Person-Related Privacy Policy Options and Usage Matrix

Personal Flag	Publish	Name, Identify	Index, Search
No Restriction (SIP)	✓	✓	✓
Do not Search (S)	✓	✓	✗
Do not Identify (I)	✓	✗	✗
Do not Publish (P)	blur face	✗	✗

Table 8.2: Picture-Related Privacy Policy Options

No Geotag (G)	remove location meta data
No Timestamp (T)	remove date and time information

### 8.5.1 Picture Privacy Policy

A usable policy needs to focus on key aspects to be easily understood and ultimately gain user acceptance. Our framework consists of three simple person-related restrictions (Table 8.1) and two picture-wide restrictions (Table 8.2).

#### 8.5.1.1 Personal Flags

Flags are attached to a specific person in a picture and are applied individually. This means that it might be possible to find a picture by using the meta data of one person in the picture but not that of another person in the picture. That is, each person in a picture can set his/her privacy settings individually and gain informational self-determination.

The *Do not Search* (S) flag specifies that the user does not want to be found through an internal or external search engine using a person-specific keyword. This includes the person's real name, user name, birth date, and any other indexable data. Furthermore, it includes other images (e.g., "find similar faces," "find other pictures of the same user") or joined data (e.g. "other customers who bought this product," "friend of the person"). In the case of Facebook, the user accepts being identified ("tagged") in a photo but does not want this photo to show up if someone searches on his/her name or visits his/her timeline. However, the picture still can be included in an index based on the geolocation or timestamp information (e.g., Flickr's "map this picture" feature).

The *Do not Identify* (I) flag specifies that the user does not want to be identified in a picture. This includes automatic face identification as well as manual name tagging by other users. If this information should become available by other means despite this specification, it is not to be included in a search index.

The *Do not Publish* (P) flag specifies that the user does not want to have any pictures of him or her published. If the person is not the main subject (e.g., his/her image was unintentionally captured) his or her face should be blurred, pixelated, or covered to

make identification impossible. The publisher (e.g., newspaper editor, blog writer, or uploading social network user) can also crop the picture to exclude the person in question. A modern publishing system can blur faces automatically in accordance with P3F policy. (For justified exceptions, see Section 8.5.2.)

### 8.5.1.2 Picture Flags

Two additional picture-related flags complete the privacy policy (see Table 8.2). The *No Geolocation data*  $\textcircled{G}$  flag specifies that geographic location should not be added, displayed, or indexed for this picture, and the *No Timestamp*  $\textcircled{T}$  flag specifies that a timestamp should not be processed for this picture.

### 8.5.1.3 Flag Precedence

Flags are encoded in symbols, markings similar to 2D barcodes, patterns similar to 1D barcodes, or facilitating other visual techniques (e.g. watermarking) onto one's wardrobe (e.g., shirt or jacket) or onto accessories (e.g., hat, cap, or button) as described in Section 8.7.1. Multiple visual encoding schemes are needed to blend unobtrusively into the desired wardrobe style, as this is often predetermined by external factors. Some of these codes might be so subtle that are unnoticed by the human eye - such as a 1D barcode in a stripe pattern on a tie or t-shirt.

However, it is infeasible for a person to carry around a set of shirts and change them in accordance with each occasion during the day. The person may therefore have different policies attached to different articles of clothing, with a defined precedence. The precedence order follows the ease with which a specific article is changeable in public. For example, one does not normally change a pair of trousers in public, so the policy attached to one's trousers has the lowest precedence. Since ties, scarves, and jackets are easier to change and since a cap or button can be changed on the fly, they have the highest precedence.

A person can display the *No restriction*  $\textcircled{SIP}$  policy flag encoded on an article of clothing with the highest precedence to cancel out his/her fallback or default policy encoded on another article of clothing.

Table 8.3: Policy Precedence

Precedence Level	Wardrobe Example
0	Trousers, Shirt, Belt
1	Tie, Scarf, Jacket
2	Cap, Hat, Button

If multiple policies are displayed on different articles of clothing with the same precedence level, the individual restrictions are added up. More specifically, policy  $P$  consists of an  $n$ -tuple of restrictions or flags  $\langle R_1, R_2, \dots \rangle$ . The  $n$ -th policy of precedence level  $p$  for an individual  $i$  is denoted as  $P_{i,p,n}$ . The effective policy  $EP$  is the  $n$ -tuple of all the strongest individual restrictions at the highest precedence level  $pmax_i$  found for  $i$ . For person-related restrictions, let  $\textcircled{\text{SIP}} < \textcircled{\text{S}} < \textcircled{\text{X}} < \textcircled{\text{P}}$ , whereas  $\textcircled{\text{G}} < \textcircled{\text{G}}$  and  $\textcircled{\text{T}} < \textcircled{\text{X}}$ .

$$EP_i = \langle \max(P_{i,pmax_i,1..n}[R_1]), \\ \max(P_{i,pmax_i,1..n}[R_2]), \dots \rangle$$

### 8.5.2 Photographer's and Publisher's Stakes

P3F was designed as an opt-out procedure to publishing systems for use by individuals wanting to restrict how their picture is used. Although this is not the most privacy supportive design, a system that does not include the current reality as the default will have a hard time gaining broad acceptance.

P3F does not strictly prevent publishing of pictures in contravention of the user's policy. Manual exceptions are allowable for two main reasons.

#### 8.5.2.1 Side Agreement

The photographed person gave his/her consent for publishing a specific picture to the photographer before or after the photo was shot.

#### 8.5.2.2 Justified Exceptions

There are generally several legal exceptions to the rights a person has regarding his/her picture. They include exceptions on the use of pictures by the media of people of public interest and on the use of police booking photographs.

These exceptions might be implemented in a publishing system (e.g. social network or newspaper system) as an additional manual work step to override a restriction. In such instances, the publisher (e.g. a social network user, blog poster, or newspaper editor) must confirm that he/she has the permission of the photographed person to publish the picture or that an exception applies.

As the original P3F policy remains part of the picture, another publisher that (re)uses the picture will also have to contact the photographed person for permission to publish or state the exception that applies.

This mechanism also accounts for possible mistakes and false positives that may occur during processing.

## 8.6 Enforcement

Since P3F is an opt-out system, the impact on the status quo for the industry is minimized. Nevertheless, reservations could remain and thus reduce the incentive for wardrobe producers to offer such coded clothing. However, two examples demonstrate how similar systems were successfully adopted in the past.

After a public outcry shortly after the introduction of *Google Street View*, the service started to blur faces and license plates [123]. In Germany, Google additionally agreed to provide an opt-out feature after the Minister of Justice of Rhineland-Palatinate, the data protection supervisor for Schleswig-Holstein, and Germany's Federal Consumer Protection Minister threatened the company with legal action. Since 2009, German home owners can blur the image of their home [74].

Another example is the integration of a banknote detection algorithm in popular software (e.g., Photoshop and PaintShop Pro), several printers, several scanners, and most color copying machines [218]. In 2004, the *Central Bank Counterfeit Deterrence Group* [1] (founded by the G10) published a *Counterfeit Deterrence System* software module for detecting banknotes that has subsequently found its way into many products although it is only available as a closed source module and there is no legal obligation for companies to include it.

## 8.7 Technical Architecture

The overall architecture of P3F is displayed in Figure 8.3. Clothing and accessory manufacturers can use the *P3F encoder* to create a visual marking or pattern that matches or blends into any wardrobe style. Individuals can use it as well for some accessories (e.g., a button) and some articles of clothing (e.g., homemade articles).

Once a picture of the user wearing something with the user's policy embedded ends up in a publishing system, two other components are used to implement the policy. The *P3F decoder* searches for and decodes the embedded policy and attaches it to the appropriate person (or face). System integrators might decide to keep the original image and the extracted P3F information in their database and apply them at the point of publishing (Figure 8.3, "Publishing 1"). Others might decide to immediately run the picture through

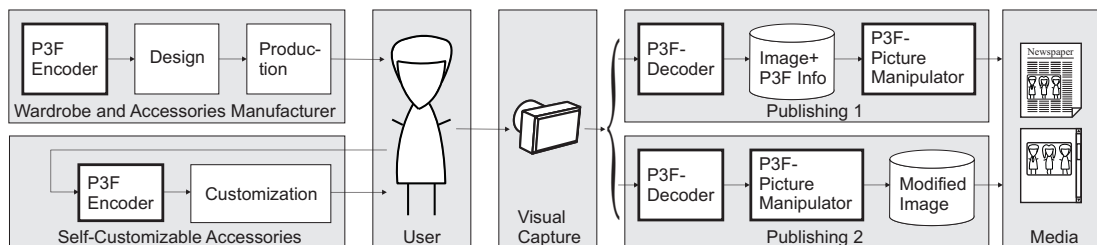


Figure 8.3: P3F system architecture

the *P3F picture manipulator* to remove the meta data and blur the appropriate faces (Figure 8.3, "Publishing 2").

### 8.7.1 Visual encoding

In its current form (Section 8.5.1), an individual's policy is encodable in six bits. Thus, the encoding scheme does not have to offer a high data density, but it must meet certain other technical requirements.

#### 8.7.1.1 Technical requirements

**Illumination stability** The code should be decodable under a wide range of lighting conditions. However, under conditions making face identification impossible, a decoding failure is tolerable.

**Blurriness tolerance** Picture blurriness can arise from sub-optimal auto-focus mechanisms because the photographer actually focused on another object or person or moved the camera during exposure (a common problem with amateur photographers).

**Size and clipping invariance** The code should be decodable from shots with different fields of view. Therefore, it should be so redundant that a partial capture in a close-up produces results as good as those in a wide shot. Furthermore, in a wide shot, a larger part of the code is recorded but with a reduced resolution compared to a close-up. Fine encoding that repeats multiple times is better for close-up shots while coarse encoding is better for wide shots. Ideally, a code unifies both traits.

**Distortion stability** People do not always face the camera head-on, especially when they are being photographed unintentionally. Furthermore, the human body is not a flat board, and loose clothing tends to fold and wrinkle. Another faults may arise from lense distortion or improper washing or drying of the person's clothing.

**Noise robustness** Another artifact introduced by cameras is noise, especially in low-light and low-contrast situations due to the automatic camera gain amplifying the sensors background noise.

**Computational weight** The detection algorithm should be lightweight because operators of publishing systems will most likely demand one that conserves computational resources.

**Compression stability** Digital photography greatly depends on picture compression algorithms. They commonly destroy details in pictures and introduce artifacts. These algorithms are often based on a psycho-visual model of human visual perception and are therefore not optimized for computer vision purposes. A very common compression method for photographs on the Internet is JPEG.

**Blind decoding ability** The decoder should have the ability to decode the data without prior knowledge of the original pattern used to encode the data or the data that is being looked for (a common prerequisite for some watermarking techniques).

**Detection accuracy** Detection accuracy should be high with a slight bias toward false positives since people typically feel more comfortable with more privacy than with less. False positives can still be overridden by the publisher if necessary.

**Error detection and correction** The encoding scheme should have an error detection or correction code to avoid producing erroneous results.

### 8.7.1.2 Aesthetic Demands

**Dress code** Dress codes are often imposed by society, the employer, or another external entity. The coding scheme should thus produce markings and patterns that blends into the imposed dress code.

**Fashion** People additionally often have their own fashion demands. The coding scheme should thus produce markings that blends into the individual's fashion style.

**Adaptive** Clothing is sold in many different colors and shapes. The code should thus be versatile and work with many different colors and shapes.

**Unobtrusive** The application of P3F should require only a slight adjustment in clothing style. The code should be subtle with low visual impact. It should be unrecognizable by other people, thus minimizing social complications.

### 8.7.2 One encoding to rule them all?

As this wide range of partially contradictory requirements and demands suggests, it is unlikely that there is one encoding and marking scheme that meets all of them and therefore is suitable in all situations. P3F is thus based on a modular design with different

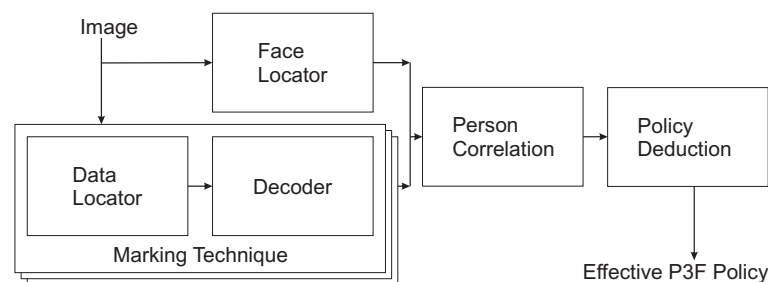


Figure 8.4: P3F policy decoder



encoding schemes. As Figure 8.4 illustrates, these schemes ideally come in a split form to reduce the computational impact. A lightweight detector finds candidates for code occurrences that are subsequently fed into a potentially computational more demanding decoder.

The *P3F decoder* additionally searches for individuals by using common face detection techniques and assigns all found policies to these persons. Finally, an effective P3F policy is deduced for each person by using the precedence rules described in Section 8.5.1.3.

### 8.7.3 Commonly used and available visual encoding schemes

#### 8.7.3.1 1D Barcodes

Linear barcodes are the obvious choice for all articles of clothing with a stripe pattern. They are computationally easy to detect using frequency analysis. Clipping stability can be achieved by continuous repetition. However, common linear barcodes typically lack size invariance and the ability to repeat them continually due to quiet zones and distinctive begin and end markers.

Table 8.4: Problematic properties of commonly available 1D and 2D barcodes for P3F

Barcode	Type	Disguisability	Seamless pattern	Quiet zone	Visual marker	Perspective distortion	Non-linear distortion
EAN & UPC	1D	low	no	yes	modest	low <sup><math>\beta</math></sup>	no
Codebar	1D	low	no <sup><math>\beta</math></sup>	no <sup><math>\beta</math></sup>	no	low <sup><math>\beta</math></sup>	no
Code 39	1D	low	yes	no	no	low	no
Code 93	1D	low	yes	no	thick start/stop markers	low	no
Code 128	1D	low	no	yes	no	low	no
2of5	1D	low	yes	no	no	low	no
MSI	1D	low	yes	no	no	low	no
PDF417	2D	low	no	no	prominent side bars	low	no
Aztech	2D	low	yes	no	prominent center marker	no	no
DataMatrix	2D	low	no	required	thin border	medium	no
MaxiCode	2D	low	no	yes	prominent center marker	low	no
QR Code	2D	medium	no	yes <sup><math>\gamma</math></sup>	prominent corner marker	medium	low
Microsoft Tag <sup><math>\alpha</math></sup>	2D	very good	no	yes	prominent border	medium	no

<sup>$\alpha$</sup>  requires online connection in vendor's design     <sup>$\beta$</sup>  with typical unmodified decoder

<sup>$\gamma$</sup>  required by standard, most decoders are very tolerant

### 8.7.3.2 2D Barcodes

Some barcode schemes (e.g. Microsoft Tag [208]) offer customizability up to the point where the data is completely disguised by a picture (Figure 8.5). However, most barcodes require an easily spottable synchronization marker or a quiet zone around them. Both make it difficult to hide them visually in a repeatable pattern. See Table 8.4 for a comparison.

### 8.7.3.3 Augmented Reality Markers

These are similar to 2D barcodes but are highly distortion invariant and enable calculation of the relative distance and angle of the barcode surface to the camera. However, they are visually very prominent as they are optimized for real-time applications.

### 8.7.3.4 Symbols

Some symbols might be suitable for encoding. However, they are likely to be more intrusive and easily spotted by other people.

### 8.7.3.5 Watermarking techniques

Watermarking and steganography are often used in the context of *digital rights management* (DRM) and have been extensively researched over many years. However, their use for redigitized surfaces and especially for textiles is a new research area.

Watermarking can be roughly split [56] into techniques for *detecting* a known pattern (1-bit information) and those for *reading* arbitrary data. Furthermore, some algorithms (*non-blind*) need either the original image or original pattern whereas *blind* decoding algorithms can recover the watermark without a priori knowledge. Most watermarking and steganographic techniques are designed for natural images and patterns and perform poorly for geometric shapes and drawn images.

Robustness against a *print-scan attack* [276] most closely resembles the normal use case in our application: a printed image or pattern with a watermark is redigitized with an optical sensor. Therefore, this use case is our main selection criteria.

Xin et al. [330] proposed watermarking for textiles by using the structure of the woven fabric. However, the watermark is only readable from very close distances.



Figure 8.5: Microsoft Tag offers great customizability for concealing the data pattern but still needs a distinctive feature (i.e., a bulky border) for synchronization [208]

Otori and Kuriyama [236] developed a very promising watermarking technique for natural textures that creates watermarks robust against analogue transportation paths (e.g., visual capture). Their implementation, however, is neither size and cropping invariant nor self synchronizing (i.e., it requires border lines).

Shirali-Shahreza et al. [276] put forward a very simple method for watermarking textiles based on *collage steganography* (i.e. object position in pictures). However, this method is non-blind and therefore not suitable for our application.

Zhu et al. [342] developed a watermarking system (originally for GIS applications) that is robust against resizing and cropping.

## 8.8 Limitations

The proposed framework does not try to mimic compulsory DRM systems. It is a best-effort system with an optional manual override. Furthermore, it does not allow the permissions to be changed after picture capture, unlike a central-database-based approach.

Although some visual marking systems are promising candidates as a foundation for further adaptations, they all have their limits on strictly plain color cloths.

While the framework can handle multiple people in an image, partially hidden people could pose a challenge in matching policies to the individuals.

The proposed framework is aimed at the recognition of faces in pictures. While there are other forms of identification, such as identification of movement dynamics, face recognition is more commonly used and offers better distinction properties.

## 8.9 Future Work and Enhancements

As none of the examined marking techniques fulfills all of the requirements we identified, there is a need for further research and development. Our next task is the implementation and evaluation of a prototype system based on our proposed framework.

Additionally, the development of a stable watermarking technique with a high data capacity would enable the use of a per-entity permission model based on private/public key cryptography. The individual would encode his/her publishing permission in the watermark and provide trusted publishers and/or friends with the decryption key. An automated P3F enforcer built into the social network or publishing system could then verify the permission by using a local permission-key database. To avoid introducing a new way of linkability, the watermarking algorithm should be parameterized with a part of the key. A publisher without the key would be unable to decode a deterministic bit pattern from the watermark.

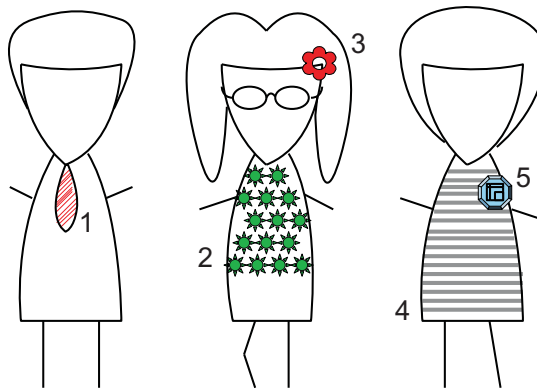


Figure 8.6: P3F encoding examples: (1) stripe pattern on tie, (2) watermarked pattern, (3) symbol or accessory, (4) stripe (1D-Barcode) on shirt, (5) button with 2D barcode

## 8.10 Reception and Evaluation

As a follow-up, we conducted a study on requirements and user acceptance of picture privacy systems [179]. We tested three different prototypical systems (including a P3F-style system) in two different locations (and scenarios people were in) with fundamentally different privacy requirements with 20 qualitative semi-structured interviews.

### 8.10.1 Scenarios

By choosing different locations we also chose different scenarios our subject were immersed and involved in. Thus, putting them into a different mind set prototypical for different situations in life. Therefore, subjecting them first-hand to typical constrains that a particular privacy-mediating solution experiences in that very situation.

The first situation was a local beach where we approached people wearing bathing wear. In this setting we expected the need for privacy to be above average and the limitations on things a person can take, wear, or use to be more restrictive.

The second situation was a local outdoor cafe environment in a rather trendy museums area. A place to meet and chill out with friends.

During all interviews one of the interviewer was wearing a Google Glass [5] as a prototypical device that can covertly record pictures and videos at any time. Thus, urging the confrontation of the candidate with potential recordings or snapshots of them.

### 8.10.2 Tested Picture Privacy Enhancing Technologies

In our study [179], we chose three conceptional different Picture-PETs that were explained to the participants and offered visual representations. Interviews were conducted in German, the visualizations below were translated for the reader's convenience.

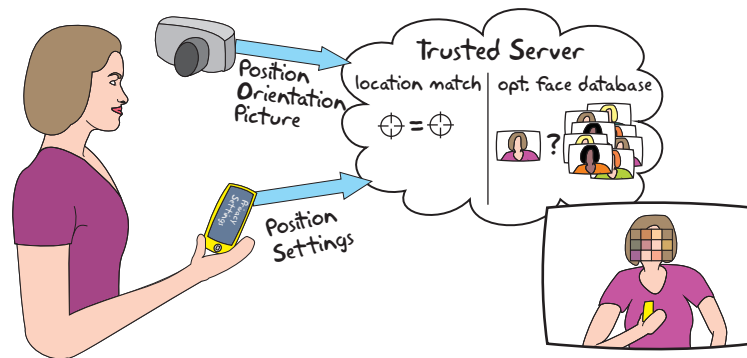


Figure 8.7: Diagram explaining the Privacy App concept (translated)

### 8.10.2.1 The Privacy App

The *Privacy App* is mainly inspired by the SnapMe [149] and FaceBlock [339] apps. Both apps have a range of configuration options. For the purpose of this study, we defined that the location of the app user and the location of the nearby cameras are transmitted to a photo sharing server together with the privacy preferences of the user. Due to the co-location information the photo sharing service can blur the faces of people with corresponding app configurations when a photo is uploaded. This feature is additionally supported by face recognition software. This concept represents the traditional technology approach.

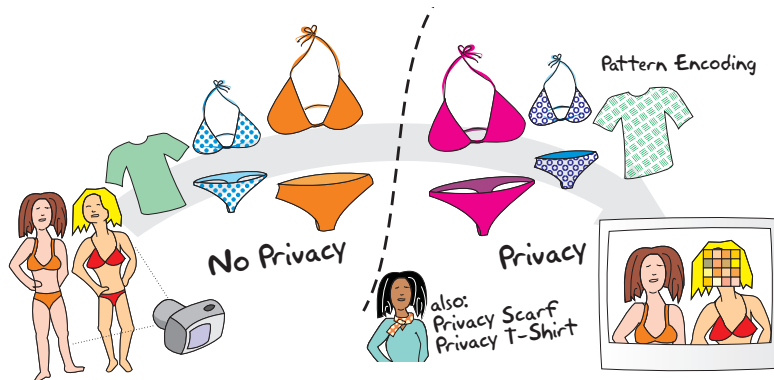


Figure 8.8: Diagram explaining the Privacy Fabric concept (translated)

### 8.10.2.2 The Privacy Fabric

The *Privacy Fabric* is a piece of cloth to communicate a user-defined privacy policy. The concept is inspired by P3F [95] and privacy hats and scarfs by Schiff et al. [272]. It is based on pattern recognition and works without additional hardware. To create a privacy cloth, e.g., swimming trunks, T-shirts or any other piece of clothing with a

privacy pattern, clothing and accessory manufacturers can use a specific encoder to create a visual marking or pattern that matches any wardrobe style. Either the wearable doing the recording or the photo sharing service can detect if a person is wearing a piece of clothing with a privacy preference encoded in it and can blur those peoples' faces. The main advantage of this method is that it is unobtrusive as no piece of technology needs to be operated. This concept represents the most *wearable* PET and we hypothesized that users would prefer this in the beach scenario since it would allow them to express their privacy preferences in an unobtrusive way. It works offline and does not need to identify or link the participants.

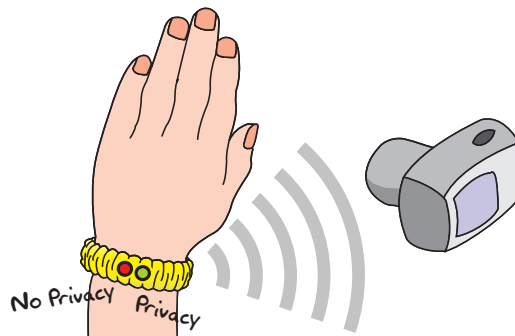


Figure 8.9: Diagram explaining the Privacy Bracelet concept (translated)

### 8.10.2.3 The Privacy Bracelet

We designed the concept of the *Privacy Bracelet* as a mix between the privacy smartphone app and the privacy fabric. While it uses technology of similar power to the smartphone, it is wearable, similar to fitness trackers (e.g., FitBit). This concept was not based on related work but was designed to give us a half-way point between the two PETs described above and allow us to have middle ground during the interviews to be able to contrast between the two technologies described above. In our concept the privacy bracelet has a simple button to turn privacy on and off. If the privacy button is turned on, the device emits a signal that wearable cameras would be able to detect and blur the faces of the bracelet wearers.

### 8.10.2.4 Results

The full results, methodology, and questionnaires are described in Krombholz et al. [179]. Herein we describe only the most important aspects regarding P3F and other fabric-based picture-privacy enhancing technologies.

In general, all participants expressed a strong interest in a picture privacy enhancing (or mediating) technology to communicate their privacy preferences towards photographers. On average, our participants indicated a high interest of 4.3 out of 5 on a Likert scale. The lowest indicated number was 3.

After presenting the three methods (as described above), 13 participants preferred the privacy bracelet. Four preferred the app and only two the privacy fabric. One participant said that he finds all of the suggested methods useless.

Interestingly, we saw no trend difference between participants preferences based on the location we conducted the interview at.

We were somewhat surprised by these results. We expected a candidates to favor privacy fabric in the beach scenario and a more indifferent result with a small lean toward the traditional smartphone app in the cafe environment. Thus proving our assumptions wrong.

The main reason for the preference as explained by the interviewees was convenience and the ease of use. Bracelet supporters highlighted the very intuitive the user interface. Some also specifically mentioned the anonymity aspect of the data transmission to the camera. Many participants indicated that they do not want facial recognition and location tracking to be performed in the background, as done by the privacy app concept. They perceived the use of such methods in privacy tools as paradoxical.

Although most preferred the bracelet, eight participants also liked the idea behind the privacy fabric. However, they were concerned it could restrict their personal styling preferences and mentioned it could be complicated to adjust their clothing in the field based on their context-related privacy preferences.

## 8.11 Proof of Concept Implementation

In a (simplified) proof of concept (PoC) we used Matlab to experience possible problems and hurdles of an implementation. It is available as an open-source download [86]. This implementation doesn't strive to be complete nor do we claim to advance the field of computer vision. We expect the latter to solve many problems that we experienced, with novel techniques.

- While face recognition is pretty easy (i.e., many off-the-shelf algorithms available), segmenting the body and connecting all of them into one entity needs much more work. This is necessary, as encoding covers the body, but effects the head. This becomes especially hard if bodies are close to each other, partially covered by each other, or intertwined.
- Our requirements from Section 8.7.1 are hard to achieve all at once. The idea of self-synchronization through fractal patterns needs to be researched more.
- CPU power on miniature computing platforms – such as Google Glass – is extremely limited. Code will need to be heavily optimized for these platforms, which contradicts the universality P3F wants to achieve. It is however feasible to deploy in online services backed up by data centers.

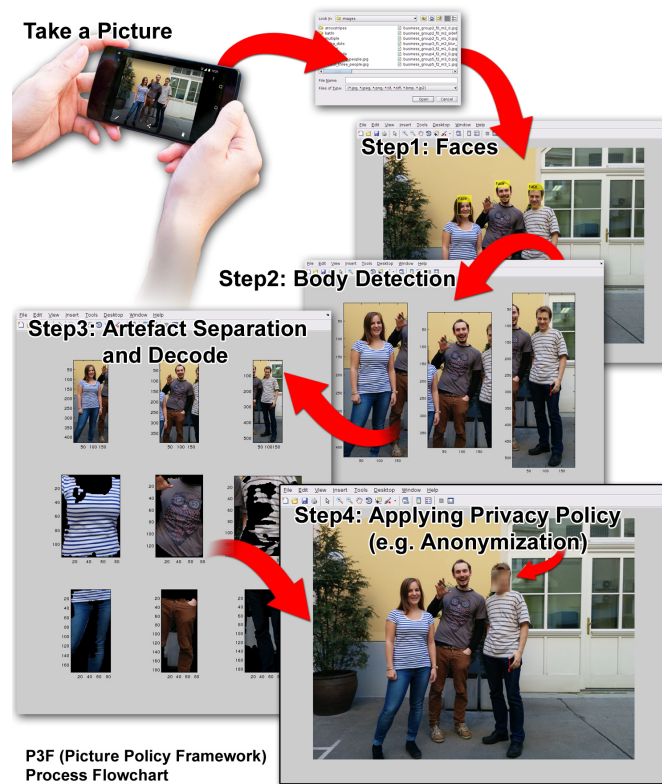


Figure 8.10: PoC workflow in Matlab

- Development of custom marking symbologies requires a standardized set of test cases – e.g., perspectives, illumination, wrinkles, body position, and so on – to achieve comparability and a minimum usability standard.

## 8.12 Conclusion

The framework we have presented enables involuntarily or unintentionally photographed individuals to express their picture privacy policy in a machine readable format and (to some extent) automatically enforce it. A flag system is used to restrict usage and linkability. This information is encoded in an unobtrusive way in wardrobe patterns and accessory designs. The encoding can be done using barcodes, watermarking, steganography, etc (Figure 8.6). While these forms of encoding are often used in context of DRM by corporations to protect their interests, they are used here for the benefit of individuals.

None of the techniques examined for building patterns around barcodes or the schemes examined for encoding information in a natural pattern meet the identified requirements and demands for a system that would fit into everyday life. This creates room for further research and development of which a few have been implemented.



However, in a following user study we found, that the flexibility offered by our system is actually confusing to many users. What we thought where easily understandable flags turned out to be far to complex for an average user. The complete passive and offline approach of our system did not out-weight the simplicity of an active bracelet with a simple on/off interface.

Introduction and mandating the use of our or any other picture privacy enhancing technique will require public pressure and most likely also regulatory policies. The following section looks at the technology impact and market roll-out aspects of these systems.



**Part V**  
**Conclusion**



# Conclusion

## 9.1 Systemic View

Large-scale and critical infrastructure face unique security challenges. Properties that work on smaller, homogeneous, or island systems change their shape at large scale.

In our work, we encountered different emergent and systemic processes which led to security problems.

**Independence Premise:** Many of today's networks incorporate overprovisioning for efficiency reasons (i.e., promising more capacity than is available). For example, just as airlines know that a certain number of passengers won't show up, Internet providers know that individual customers rarely use the full data capacity at their disposal. Likewise, server operators recognize that not all Internet users connect to their service at once, and thus only needs to provide a particular capacity. History has shown, that this approach works under the assumption of independent entities, but actually makes a system prone to collusion, i.e., (distributed) denial of service attacks.

In our work, we demonstrated that most networked devices are part of a cyber-physical system and thus are connected to a second, often neglected, physical network. The latter is not a one-way connection, e.g., it is not just energy flowing out of the wall socket, but the users' demand is traveling back into the grid to the generators. In power grids it is mostly not the actual power capacity that matters, but the rate at which it changes, thus making them vulnerable to a whole new set of denial-of-service attacks.

**Composition:** Compositional insecurity stems from multiple sources, but typically involves numerous methods or standards. When either stacked upon each other, put on equal processing level next to each other, or introduced as an optional extension, they

interfere destructively on security properties, i.a., introducing side channels, downgrade possibilities, or semantic ambiguities.

In this regard we worked on history-stealing side channels in HSTS, an HTTP extension that was originally developed to enhance security and privacy for users. Similarly, error-correcting codes are meant to increase the robustness of data transmission and storage. By using this feature to craft ambiguous messages, systems involved in handling this information do not agree on its content and thus cannot agree on its security requirements or properties. In social networks, it is the enrichment of otherwise non-machine-understandable pictures with linkable and searchable data that creates privacy problems for the individual. Radio networks used for cell phones were standardized individually without incorporating the operational context (e.g., running in parallel with older, less-secure standards). Thus, improved security properties nullify when composed with unchecked downgrade possibilities.

**Unverified Trust:** Trust-propagating hierarchies (or sequenced processing steps) create a monopoly for truth – which impersonating attackers can misuse. In mobile phone networks, this applies especially to fake base stations and their ability to control and persistently reconfigure the baseband radio properties. It is also the case for inter-operator communication networks, such as SS7. The unchallenged assumption is the always benign and omniscient operator, a paradigm found in older and centralized communication networks.

## 9.2 Offensive versus Defensive Approaches

In general, security is often considered easier to falsify than to verify. The argument runs as follows [150]: an attacker has to find just one hole, whereas the defender has to think about all the possibilities to circumvent a system. However, this does not necessarily "diminish" the work of attackers. Ideally, they highlight a new class of vulnerabilities or a surprising and novel nuance of already known problems. Either way, they are contributing to the body of knowledge.

During this work, we approached security and privacy threefold:

**Offensive:** We discovered, described, and implemented attacks and subsequently measured (or simulated) the impact. That includes our work on novel power grid attacks, browser side channels (e.g., allowing history stealing), and decoding ambiguities due to error-correcting codes.

**Defensive:** We worked on the detection, mitigation and eventually protection against attacks on already existing networks. Thus, under the premise of existing standards, we drafted and implemented effective defense methods and strategies. We approached fake base stations and race-condition attacks from the client's and operator's side of cellular

networks, but the work likely applies to other, similarly structured radio networks as well.

**Standardization and Conceptualization:** While the above-mentioned approaches aim at short-term remission from vulnerabilities, we also need to work constructively on improving standards or creating new ones as part of a long-term strategy. Thus, we took our work on systematization and root causes of security problems in cell phone networks to the appropriate standardization bodies (ITU, ETSI). We described how they span multiple network generations and why they should be tackled in a way to eliminate classes of vulnerabilities. On the conceptual and explorative front, we created concepts, implemented demo applications and evaluated methods to give individuals more control over pictures of them.

### 9.3 Resume

In an increasingly connected world many – formerly offline – components are pieced together to new implicit or explicit large-scale systems (e.g., Internet of Things, cyber-physical systems), often just by thoughtless connecting them to a network. Thus, it is of absolute importance to include security and privacy considerations early into the design process. We hope that this thesis contributes to shifting the attacker-defender asymmetry at least slightly towards the defense.





# List of Figures

2.1	Visualization of Attacks 1 and 2: The botnet can modulate the power demand much faster than power plants can react. . . . .	14
2.2	Simplified schematic of attack 3 . . . . .	22
2.3	Model of botnet zombie and method of measurement . . . . .	24
2.4	Measurement setup in detail . . . . .	25
2.5	Example: low-end office PC. PSU ramps up power within 2-3 AC cycles . . . . .	25
2.6	LG 24" TFT screen needs 8 s before going to sleep mode. . . . .	25
2.7	Gaming PC: PSU ramp up . . . . .	26
2.8	Power usage of a Brother HL2150 SOHO printer . . . . .	26
2.9	Model for static load attack (primary control) . . . . .	29
2.10	Jump function of the additional load fort the static load attack in multiples of ENTSO-E's reference incident (RI=3,000 MW) . . . . .	29
2.11	Impact of static load attack on frequency in a grid . . . . .	30
2.12	Dynamic load attack (1.5 reference incidents) . . . . .	31
2.13	Dynamic load attack at different levels of total network power and rotational inertia . . . . .	31
2.14	Inter-zone attack: Load added in one zone is removed from the other (reference incident) . . . . .	32
2.15	Transit power and frequency deviation in inter-zone attack . . . . .	33
2.16	Model for control area including primary and secondary control . . . . .	34
3.1	Generic mobile network architecture and the herein focused areas. . . . .	43
3.2	All cells of a Location Area are paging the same phones . . . . .	46
3.3	A UE can listen to any paging channel in the Location Area, but has to inform the HSS/HLR about entrance into a new Location Area. . . . .	47
3.4	Systematization methodology for Vulnerabilities . . . . .	48
3.5	Visualization of Systematization including Attack Aims, Attacks, Causes, and Root Causes. . . . .	57
3.6	Downgrade attack from 4G to 2G using <i>Access Technology not allowed</i> messages (simplified) . . . . .	70
3.7	Downgrade attack from 4G to 2G using <i>Radio Resource Control</i> messages to reassign an GSM ARFCN to the UE . . . . .	72

4.1	Dedicated device: block diagram . . . . .	86
4.2	Construction of the dedicated stationary unit . . . . .	87
4.3	Screenshots of the mICC . . . . .	88
4.4	Maximum number of unique distinct cells received throughout the day (sICC)	90
4.5	Cell ID lifetime throughout the experiment . . . . .	92
4.6	Field test for all three GSM networks . . . . .	92
4.7	Number of TMSIs to (re)appear in the n-th paging resend within a 10 second window. . . . .	95
4.8	sICC: Color coded by signal strength of received cells. (Google Maps) . .	97
4.9	Stationary IMSI Catcher Catcher, Demonstrating the tropospheric ducting: Range up to 80 km. Color coded by mobile operator. (Google Maps) . . .	98
5.1	A tracking IMSI Catcher identifies a phone and drops it back into the real network. . . . .	104
5.2	Phone models that produce a new LUR after a Location Update Reject .	105
5.3	Cipher usage on 2G nationwide . . . . .	105
5.4	Origin LAC provided at Location Update Requests . . . . .	106
5.5	Some of the tested phones . . . . .	106
5.6	A man-in-the-middle IMSI Catcher identifies a phone . . . . .	108
5.7	Authorization round trip time: Distribution of time between <i>Authentication Request</i> and <i>Authentication Response</i> on a real network. . . . .	109
5.8	Normalized distribution of authorization round trip time broken up by phone models . . . . .	110
5.9	Location update tunneling effect . . . . .	112
6.1	History stealing flow chart on users browser reaction. . . . .	122
6.2	History stealing scenario in a public Wi-Fi hotspot setting. . . . .	123
6.3	History stealing also works, when user is using a VPN for common browsing, as the history is shared between the captive portal and the VPN sessions.	125
6.4	Percentage of scanned sites that are vulnerable to our history stealing attack	126
6.5	Percentage of scanned sites that are vulnerable to our history stealing attack	128
6.6	Structure of the implementation . . . . .	129
6.7	Report screenshots . . . . .	130
7.1	Illustration of a binary polyglot: data is interleaved using references with unused space in between, reserved or unused fields, etc. . . . .	137
7.2	Visualization of an error-correction code: the destroyed data is reconstructed.	137
7.3	Visualization of embedding alien data into an ECC protected data: the alien data can overwrite big (consecutive) chunks as long as the error margin is not overstressed. . . . .	138
7.4	Similar to Figure 7.3 a Barcode-in-Barcode can be constructed. Note: both parts are valid barcodes. . . . .	139
7.5	Popular 2D Barcodes with rectangular pixels: Quick Response, Aztec, Data Matrix . . . . .	140

7.6	Decoding ambiguity: the detector for a particular code is tested first, therefore the others are not considered. . . . .	141
7.7	Sliding over the barcode will make the smaller inner barcode fully visible before the entire (outer) barcode. . . . .	142
7.8	Critical areas of an QR Code: location markers (1), quiet zone (2), timing pattern (3), and alignment markers (4). . . . .	142
7.9	Aztec in QR: NeoReader on iOS strongly prefers Aztec over QR . . . . .	144
7.10	Data Matrix (bottom right) in QR . . . . .	145
7.11	Data Matrix (bottom right, rotated) in QR . . . . .	145
7.12	Data Matrix (center) in QR . . . . .	146
7.13	Data Matrix (center, white space added) in QR . . . . .	146
7.14	QR in QR, corner, w/o white space . . . . .	147
7.15	QR in QR, semi corner and center, w/o white space . . . . .	148
7.16	QR in QR, center with white space . . . . .	148
8.1	Illustration of how a picture can accumulate meta information and end up in various places. . . . .	156
8.2	Example where P3F sits in the distribution path . . . . .	158
8.3	P3F system architecture . . . . .	162
8.4	P3F policy decoder . . . . .	164
8.5	Microsoft Tag offers great customizability for concealing the data pattern but still needs a distinctive feature (i.e., a bulky border) for synchronization [208]	166
8.6	P3F encoding examples . . . . .	168
8.7	Diagram explaining the Privacy App concept (translated) . . . . .	169
8.8	Diagram explaining the Privacy Fabric concept (translated) . . . . .	169
8.9	Diagram explaining the Privacy Bracelet concept (translated) . . . . .	170
8.10	PoC workflow in Matlab . . . . .	172



# List of Tables

2.1	Emergency routines in case of under-frequency in Germany [313, p65] similar to the ENTSO-E policies [303, p26] . . . . .	16
2.2	Modulated load by device . . . . .	27
2.3	Prototypical computer hardware configurations with expected modulatable load . . . . .	35
2.4	IoT scenarios . . . . .	35
2.5	Infections needed . . . . .	35
3.1	Categorization of mobile security attacks by their aim . . . . .	53
3.2	Root causes related to causes . . . . .	55
3.3	Cipher Overview . . . . .	60
3.4	Overview on forwarding techniques and its effects . . . . .	73
4.1	IMSI Catcher detection matrix . . . . .	83
7.1	Tested applications and their barcode standard support, as well as other features (as of 2015). . . . .	143
8.1	Person-Related Privacy Policy Options and Usage Matrix . . . . .	159
8.2	Picture-Related Privacy Policy Options . . . . .	159
8.3	Policy Precedence . . . . .	160
8.4	Problematic properties of commonly available 1D and 2D barcodes for P3F	165



# Bibliography

- [1] Central Bank Counterfeit Deterrence Group. <http://www.rulesforuse.org/>, accessed 2013-05-07.
- [2] Creative Commons. <http://creativecommons.org/>.
- [3] Determine the gender of a first name. <https://genderize.io/>, accessed 2016-01-20.
- [4] Digital cellular telecommunications system (Phase 2+); Interworking between Phase 1 infrastructure and Phase 2 Mobile Stations (MS). URL: [http://www.etsi.org/deliver/etsi\\_ts/101600\\_101699/101644/05.01.00\\_60/ts\\_101644v050100p.pdf](http://www.etsi.org/deliver/etsi_ts/101600_101699/101644/05.01.00_60/ts_101644v050100p.pdf).
- [5] Google Glass. <http://www.google.com/glass/start/>, accessed 2013-05-06.
- [6] HSTS pre-load submission. <https://hstspreload.appspot.com/>, accessed 2016-01-31.
- [7] iOS 7 and captive portal - a guide to captive portal requirements. <http://blog.tanaza.com/blog/bid/318805/ios-7-and-captive-portal-a-guide-to-captive-portal-requirements>, accessed 2015-01-20.
- [8] iOS 7 and captive portals-changes to apple request URL. <http://stackoverflow.com/questions/18891706/ios7-and-captive-portals-changes-to-apple-request-url>, accessed 2015-01-20.
- [9] ISO/IEC 16022: Information technology – Automatic identification and data capture techniques – Data Matrix bar code symbology specification.
- [10] ISO/IEC 18004: Information technology – Automatic identification and data capture techniques – QR Code 2005 bar code symbology specification.
- [11] Network Portal Detection - The Chromium Projects. <https://www.chromium.org/chromium-os/chromiumos-design-docs/network-portal-detection>, accessed 2015-01-20.

- [12] Official ZXing ("Zebra Crossing") project home. <https://github.com/zxing/zxing>, accessed July 18th 2014.
- [13] Shazam - Music Discovery, Charts & Song Lyrics. <http://www.shazam.com/>, accessed 2016-02-01.
- [14] WiGLE Stats. <https://wagle.net/stats#ssidstats>, accessed 2015-01-25.
- [15] Digital cellular telecommunications system (Phase 2+); Mobile Stations (MS) features, 1999.
- [16] Hypertext Transfer Protocol - HTTP/1.1- Section 10, Status Code Definitions. RFC 2616, 1999. <https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>.
- [17] Android Issue 5353: Cipherring Indicator, 2009. <https://code.google.com/p/android/issues/detail?id=5353>, accessed 2013-07-14.
- [18] Google Chrome Privacy Whitepaper, 12 2015. <https://www.google.com/chrome/browser/privacy/whitepaper.html>, accessed 2015-01-20.
- [19] 3GPP. Location Services (LCS); Mobile Station (MS) - Serving Mobile Location Centre (SMLC) Radio Resource LCS Protocol (RRLP). TS 04.31, 3rd Generation Partnership Project (3GPP), 06 2007. URL: <http://www.3gpp.org/ftp/Specs/html-info/0431.htm>.
- [20] 3GPP. SIM/USIM internal and external interworking aspects. TR 31.900, 3rd Generation Partnership Project (3GPP), 12 2009. URL: <http://www.3gpp.org/ftp/Specs/html-info/31900.htm>.
- [21] 3GPP. 3G security; Security architecture. TS 33.102, 3rd Generation Partnership Project (3GPP), 12 2010. URL: <http://www.3gpp.org/ftp/Specs/html-info/33102.htm>.
- [22] 3GPP. Service aspects; Charging and billing. TS 22.115, 3rd Generation Partnership Project (3GPP), 04 2010. URL: <http://www.3gpp.org/ftp/Specs/html-info/22115.htm>.
- [23] 3GPP. Service requirements for the Evolved Packet System (EPS). TS 22.278, 3rd Generation Partnership Project (3GPP), 10 2010. URL: <http://www.3gpp.org/ftp/Specs/html-info/22278.htm>.
- [24] 3GPP. Technical realization of the Short Message Service (SMS). TS 23.040, 3rd Generation Partnership Project (3GPP), 09 2010. URL: <http://www.3gpp.org/ftp/Specs/html-info/23040.htm>.
- [25] 3GPP. 3GPP System Architecture Evolution (SAE); Security architecture. TS 33.401, 3rd Generation Partnership Project (3GPP), 06 2011. URL: <http://www.3gpp.org/ftp/Specs/html-info/33401.htm>.



- [26] 3GPP. Circuit Switched (CS) fallback in Evolved Packet System (EPS); Stage 2. TS 23.272, 3rd Generation Partnership Project (3GPP), 06 2011. URL: <http://www.3gpp.org/ftp/Specs/html-info/23272.htm>.
- [27] 3GPP. Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3. TS 24.301, 3rd Generation Partnership Project (3GPP), 06 2011. URL: <http://www.3gpp.org/ftp/Specs/html-info/24301.htm>.
- [28] 3GVision. i-nigma. Apple App Store. <https://itunes.apple.com/en/app/id388923203>, accessed July 17th 2014.
- [29] 3GVision. i-nigma Barcode Scanner. Google Play Store. <https://play.google.com/store/apps/details?id=com.threegvision.products.inigma.Android>, accessed July 17th 2014.
- [30] 3rd Generation Partnership Project. Non-Access-Stratum (NAS) Functions related to Mobile Station (MS) in Idle Mode. 3GPP TS 23.122 v8.2.0.
- [31] 3rd Generation Partnership Project. Technical Specification Group Core Network and Terminals; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS). 3GPP TS 24.301.
- [32] 50Hertz Transmission GmbH. Grid load in the 50Hertz control area, 2017. URL: <http://www.50hertz.com/en/Grid-Data/Grid-load>.
- [33] Aaron Brown. WARNING: This text message will CRASH and reboot YOUR iPhone. <http://www.express.co.uk/life-style/science-technology/580211/iPhone-Messages-iMessage-Bug-Text-Reboot-Crash>, June 2015. accessed 2018-03-02.
- [34] Ability Computers and Software Industries Ltd. 3G Interception. Sales brochure. [https://wikileaks.org/spyfiles/files/0/80\\_ABILITY-GSM\\_3G\\_Intercept.pdf](https://wikileaks.org/spyfiles/files/0/80_ABILITY-GSM_3G_Intercept.pdf), accessed 2014-02-25.
- [35] Yasemin Acar, Michael Backes, Sven Bugiel, Sascha Fahl, Patrick Mcdaniel, and Matthew Smith. SoK: Lessons Learned from Android Security Research for Appified Software Platforms. In *IEEE Symposium on Security and Privacy (SP)*, pages 433–451. IEEE, 2016.
- [36] Alessandro Acquisti. What Facial Recognition Technology Means For Privacy and Civil Liberties. Testimony at Committee on Judiciary, US Senate, July 2012.
- [37] Airpatrol. Smart Air Conditioner Controllers, 2017. accessed 2017-06-04. URL: <http://www.airpatrol.eu/>.
- [38] Ange Albertini. corkami: Reverse engineering and visual documentations. [http://code.google.com/p/corkami/#Binary\\_files](http://code.google.com/p/corkami/#Binary_files), accessed September 6th 2014.

- [39] Ange Albertini. This PDF is a JPEG; or, This Proof of Concept is a Picture of Cats. In *PoC || GTFO 0x03*. March 2014. <http://corkami.googlecode.com/svn/trunk/doc/pocorgtfo/pocorgtfo03.pdf>.
- [40] S Alvarez and T Zoller. The death of AV defense in depth? - revisiting anti-virus software, 2008. <http://cansecwest.com/csw08/csw08-alvarez.pdf>.
- [41] S. Amini, H. Mohsenian-Rad, and F. Pasqualetti. Dynamic load altering attacks in smart grid. In *2015 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5, Feb 2015. doi:10.1109/ISGT.2015.7131791.
- [42] S. Amini, F. Pasqualetti, and H. Mohsenian-Rad. Dynamic Load Altering Attacks Against Power System Stability: Attack Models and Protection Schemes. *IEEE Transactions on Smart Grid*, pages 1–1, 2017.
- [43] R. Anderson and S. Fuloria. Who Controls the off Switch? In *2010 First IEEE International Conference on Smart Grid Communications*, pages 96–101, Oct 2010. doi:10.1109/SMARTGRID.2010.5622026.
- [44] Andrew Dalton. Florida man fined 48k for jamming cellphones while driving. <https://www.engadget.com/2016/05/25/florida-man-fined-48k-fcc-jamming-cellphones/>, May 2016. accessed 2017-05-22.
- [45] Iosif Androulidakis, Dionisios Pylarinos, and Gorazd Kandus. Ciphering Indicator Approaches and User Awareness. *Maejo International Journal of Science and Technology*, 6(3):514–527, 2012.
- [46] Manos Antonakakis, Tim April, Michael Bailey, Matt Bernhard, Elie Bursztein, Jaime Cochran, Zakir Durumeric, J. Alex Halderman, Luca Invernizzi, Michalis Kallitsis, Deepak Kumar, Chaz Lever, Zane Ma, Joshua Mason, Damian Menscher, Chad Seaman, Nick Sullivan, Kurt Thomas, and Yi Zhou. Understanding the Mirai Botnet. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 1093–1110, Vancouver, BC, 2017. USENIX Association. URL: <https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis>.
- [47] Apple Inc. Apple Push Notification service Overview. [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/APNSOverview.html#/apple\\_ref/doc/uid/TP40008194-CH8-SW1](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/APNSOverview.html#/apple_ref/doc/uid/TP40008194-CH8-SW1). accessed 2017-04-06.
- [48] Myrto Arapinis, Loretta Mancini, Eike Ritter, Mark Ryan, Nico Golde, Kevin Kevin Redon, and Ravishankar Borgaonkar. New Privacy Issues in Mobile Telephony: Fix and Verification. In *ACM Conference on Computer and Communications Security (CCS)*, pages 205–216. ACM, 2012.

- [49] AT&T Services Inc. AT&T Code Scanner: QR,UPC & DM. Google Play Store. <https://play.google.com/store/apps/details?id=com.mtag.att.codescanner>, accessed July 17th 2014.
- [50] Austrian Regulatory Authority for Broadcasting and Telecommunication RTR. Current utilization for GSM of the GSM 1800 frequency band. <https://www.rtr.at/de/tk/1800MHzGSM>.
- [51] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr. Basic Concepts and Taxonomy of Dependable and Secure Computing. *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 1(1):11–33, 2004.
- [52] F M Aziz, J S Shamma, and G L Stüber. Resilience of LTE Networks against Smart Jamming Attacks: Wideband Model. In *IEEE Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1344–1348. IEEE, 2015.
- [53] Balancing Act. Egypt tries to control the use of GPS by banning except with individual licences, 2008. <https://www.balancingact-africa.com/news/telecoms-en/2765/egypt-tries-to-control-the-use-of-gps-by-banning-except-with-individual-licences>, accessed 2018-03-02.
- [54] Rebecca Balebako, Jaeyeon Jung, Wei Lu, Lorrie Faith Cranor, and Carolyn Nguyen. Little brothers watching you: Raising awareness of data leaks on smartphones. In *Proceedings of the Ninth Symposium on Usable Privacy and Security*, page 12. ACM, 2013.
- [55] Elad Barkan, Eli Biham, and Nathan Keller. Instant Ciphertext-only Cryptanalysis of GSM Encrypted Communication. *Journal of Cryptology*, 21(3):392–429, aug 2008.
- [56] Mauro Barni, Franco Bartolini, Vito Cappellini, Enrico Magli, and Gabriella Olmo. Watermarking-based protection of remote sensing images: requirements and possible solutions. pages 191–202, 2001. DOI 10.1117/12.449582. doi: 10.1117/12.449582.
- [57] David Baron. Bug 147777 - :visited support allows queries into global history, 2002. [http://bugzilla.mozilla.org/show\\_bug.cgi?id=147777](http://bugzilla.mozilla.org/show_bug.cgi?id=147777), accessed 2016-01-21.
- [58] Adam Barth. HTTP State Management Mechanism. RFC 6265, 2011.
- [59] Ramzi Bassil, Imad H. Elhajj, Ali Chehab, and Ayman Kayssi. Effects of Signaling Attacks on LTE Networks. In *IEEE Advanced Information Networking and Applications Workshops (WAINA)*, pages 499–504. IEEE, 2013.
- [60] Ralf Bendrath. Icons of Privacy, May 2007. <http://bendrath.blogspot.jp/2007/05/icons-of-privacy.html>, accessed 2013-05-03.

- [61] Neil Bergman. Abusing WebView JavaScript Bridges. <http://d3adend.org/blog/?p=314>, December 2012. Accessed: 2014-09-15.
- [62] Sulabh Bhattarai, Sixiao Wei, Stephen Rook, Wei Yu, Robert F Erbacher, and Hasan Cam. On Simulation Studies of Jamming Threats Against LTE Networks. In *IEEE International Conference on Computing, Networking and Communications (ICNC)*, pages 99–103. IEEE, 2015.
- [63] Sam Biddle. Long-Secret Stingray Manuals Detail how Police an Spy on Phones, sep 2016. URL: <https://theintercept.com/2016/09/12/long-secret-stingray-manuals-detail-how-police-can-spy-on-phones/>.
- [64] Eli Biham, Orr Dunkelman, and Nathan Keller. A Related-Key Rectangle Attack on the Full KASUMI. In Bimal Roy, editor, *Advances in Cryptology - ASIACRYPT 2005*, volume 3788 of *Lecture Notes in Computer Science*, pages 443–461. Springer Berlin Heidelberg, 2005. doi:10.1007/11593447\_24.
- [65] Alex Biryukov, Adi Shamir, and David Wagner. Real Time Cryptanalysis of A5/1 on a PC. In *International Workshop on Fast Software Encryption (FSE)*. Springer, 2000.
- [66] Josh Blackman. Omniveillance, Google, Privacy in Public, and the Right to Your Digital Identity: A Tort for Recording and Disseminating an Individual’s Image over the Internet. *Santa Clara Law Review*, 49:313, 2008.
- [67] Ravishankar Borgaonkar, Lucca Hirshi, Shinjo Park, Altaf Shaik, Andrew Martin, and Jean-Pierre Seifert. New Adventures in Spying 3G & 4G Users: Locate, Track, Monitor. In *BlackHat*, 2017.
- [68] Ravishankar Borgaonkar, Altaf Shaik, N. Asokan, Valtteri Niemi, and Jean-Pierre Seifert. LTE and IMSI catcher myths. Black Hat Europe 2015, <https://www.blackhat.com/eu-15/briefings.html>.
- [69] Zibi Braniecki. CSS allows to check history via :visited. <https://bugzilla.mozilla.org/224954>. 2003.
- [70] Brian Harrell. Why the Ukraine power grid attacks should raise alarm, 2017. URL: <http://www.csoonline.com/article/3177209/security/why-the-ukraine-power-grid-attacks-should-raise-alarm.html>.
- [71] Marc Briceno, Ian Goldberg, and David Wagner. An implementation of the GSM A3A8 algorithm. (Specifically, COMP128.), 1998. <http://www.scard.org/gsm/a3a8.txt>, accessed 2016-06-24.
- [72] Marc Briceno, Ian Goldberg, and David Wagner. A Pedagogical Implementation of A5/1., 1999. URL: <http://www.scard.org/gsm/a51.html>.

- [73] Ian Briceno, Marc and Goldberg and Wagner David. GSM Cloning, 2002. accessed 2016-01-18. URL: <http://www.isaac.cs.berkeley.edu/isaac/gsm.html>.
- [74] British Broadcasting Cooperation. Thousands of Germans opt out of Google Street View, October. <http://www.bbc.co.uk/news/technology-11595495>, accessed 2013-05-13.
- [75] buildcomputers.net. Power Consumption of PC Components in Watts, 2017. accessed 2017-05-06. URL: <http://www.buildcomputers.net/power-consumption-of-pc-components.html>.
- [76] Pierre-Marc Bureau. Malware Trying to Avoid Some Countries, 2009. accessed 2017-05-30. URL: <https://www.welivesecurity.com/2009/01/15/malware-trying-to-avoid-some-countries/>.
- [77] Sreepriya Chalakkal, Hendrik Schmidt, and Shinjo Park. Practical Attacks on VoLTE and VoWiFi. Technical report, ERNW Enno Rey Netzwerke, 2017. URL: [https://www.ernw.de/download/newsletter/ERNW\\_Whitepaper\\_60\\_Practical\\_Attacks\\_On\\_VoLTE\\_And\\_VoWiFi\\_v1.0.pdf](https://www.ernw.de/download/newsletter/ERNW_Whitepaper_60_Practical_Attacks_On_VoLTE_And_VoWiFi_v1.0.pdf).
- [78] Chris Paget. Practical Cellphone Spying. In *DEFCON 19*, 2010.
- [79] Michael Ciuffo. Transistor Clock Part 1: Power and Time Base, 2012. accessed 2017-06-05. URL: <http://ch00ftech.com/2012/06/20/2279/>.
- [80] Alan J. Cooper. The Electric Network Frequency (ENF) as an Aid to Authenticating Forensic Digital Audio Recordings – an Automated Approach. In *Audio Engineering Society Conference: 33rd International Conference: Audio Forensics-Theory and Practice*, Jun 2008. URL: <http://www.aes.org/e-lib/browse.cfm?elib=14411>.
- [81] M. Costache, V. Tudor, M. Almgren, M. Papatrantafileou, and C. Saunders. Remote Control of Smart Meters: Friend or Foe? In *2011 Seventh European Conference on Computer Network Defense*, pages 49–56, Sept 2011. doi:10.1109/EC2ND.2011.14.
- [82] Tom Court and Neil Biggs. WAP just happened to my Samsung Galaxy?, 2017. <https://www.contextis.com/resources/blog/wap-just-happened-my-samsung-galaxy/>, accessed 2017-02-05. URL: <https://www.contextis.com/resources/blog/wap-just-happened-my-samsung-galaxy/>.
- [83] Adrian Dabrowski, Isao Echizen, and Edgar R. Weippl. Error-Correcting Codes as Source for Decoding Ambiguity. In *Proceedings of Workshops at IEEE Security & Privacy 2015, Workshop on Language-Theoretic Security (LangSec)*. IEEE, 05 2015.

- [84] Adrian Dabrowski, Markus Kammerstetter, Eduard Thamm, Edgar Weippl, and Wolfgang Kastner. Leveraging Competitive Gamification for Sustainable Fun and Profit in Security Education. In *2015 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 15)*, Washington, D.C., 2015. USENIX Association. URL: <http://blogs.usenix.org/conference/3gse15/summit-program/presentation/dabrowski>.
- [85] Adrian Dabrowski, Thomas Klepp, and Nicola Pianta. IMSI Catcher Catcher source code. accessed 2014-09-30. URL: <http://sourceforge.net/p/icc/>.
- [86] Adrian Dabrowski and Katharina Krombholz. Downloads p3f, 2017. <https://p3fproject.wordpress.com/downloads/>, accessed 2017-02-20.
- [87] Adrian Dabrowski, Katharina Krombholz, Johanna Ullrich, and Edgar Weippl. QR Inception: Barcode-in-Barcode Attacks. In *Proceedings of the 4th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM 2014)*. ACM, 11 2014.
- [88] Adrian Dabrowski, Katharina Krombholz, Edgar Weippl, and Isao Echizen. Smart Privacy Visor: Bridging the Privacy Gap. In *Proceedings of Workshop on Privacy by Transparency in Data-Centric Services (PTDCS) at 18th International Conference on Business Information Systems (BIS2015)*, Poznan, Poland, 2015. Springer.
- [89] Adrian Dabrowski, Georg Merzdovnik, Nikolaus Kommenda, and Edgar Weippl. Browser History Stealing with Captive Wi-Fi Portals. In *Proceedings of Workshops at IEEE Security & Privacy 2016, Mobile Security Technologies (MoST)*. IEEE, 05 2016.
- [90] Adrian Dabrowski, Georg Petzl, and Edgar R. Weippl. The Messenger Shoots Back: Network Operator Based IMSI Catcher Detection. In *Research in Attacks, Intrusions, and Defenses: 19th International Symposium (RAID)*, pages 279–302. Springer, 2016. URL: [http://dx.doi.org/10.1007/978-3-319-45719-2\\_13](http://dx.doi.org/10.1007/978-3-319-45719-2_13), doi:10.1007/978-3-319-45719-2\_13.
- [91] Adrian Dabrowski, Nicola Pianta, Thomas Klepp, Martin Mulazzani, and Edgar Weippl. IMSI-Catch Me If You Can: IMSI-Catcher-Catchers. In *ACM Annual Computer Security Applications Conference (ACSAC)*, pages 246–255. ACM, 2014. doi:10.1145/2664243.2664272.
- [92] Adrian Dabrowski and Martin Slunsky. Hacking CCTV. 22nd Chaos Communication Congress (22C3), 2005 December. <http://events.ccc.de/congress/2005/fahrplan/events/605.de.html>, accessed 2013-05-14.
- [93] Adrian Dabrowski, Johanna Ullrich, and Edgar Weippl. Grid Shock: Coordinated Load-Changing Attacks on Power Grids. In *Proceedings of the Annual Computer Security Applications Conference (ACSAC 2017)*. ACM, 12 2017.

- [94] Adrian Dabrowski and Edgar R. Weippl. Mobile Phone's Wi-Fi Presence for Continuous Implicit Secondary Deauthentication. In *11th International Conference on Passwords*, 12 2016.
- [95] Adrian Dabrowski, Edgar R. Weippl, and Isao Echizen. Framework based on Privacy Policy Hiding for Preventing Unauthorized Face Image Processing. In *Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC2013*, pages 455–461. IEEE SMC, 10 2013. DOI 10.1109/SMC.2013.83. doi: 10.1109/SMC.2013.83.
- [96] Shuaifu Dai, Alok Tongaonkar, Xiaoyin Wang, Antonio Nucci, and Dong Song. Networkprofiler: Towards automatic fingerprinting of android apps. In *INFOCOM, 2013 Proceedings IEEE*, pages 809–817. IEEE, 2013.
- [97] Mathias Dalheimer. An open-source infrastructure for power grid monitoring, 2016. Github repository. URL: <https://github.com/netzsinus>.
- [98] Mathias Dalheimer. Momentane Frequenzabweichung im Stromnetzes, 2017. in German, accessed 2017-06-05. URL: <https://netzsin.us/>.
- [99] Dancho Danchev. How much does it cost to buy 10,000 U.S.-based malware-infected hosts?, 2013. accessed 2017-05-30. URL: <https://www.webroot.com/blog/2013/02/28/how-much-does-it-cost-to-buy-10000-u-s-based-malware-infected-hosts/>.
- [100] Matthew DeCarlo. AVG: QR code-based malware attacks to rise in 2012, 2012. <http://www.techspot.com/news/47189-avg-qr-code.html>, accessed July 18th 2014.
- [101] Department of Energy. Home Heating Systems, 2017. accessed 2017-06-08. URL: <https://energy.gov/energysaver/home-heating-systems>.
- [102] Disconnect, Inc. and Mozilla Foundation. Privacy Icons, 2011. <https://icons.disconnect.me/icons> and [https://wiki.mozilla.org/Privacy\\_Icons](https://wiki.mozilla.org/Privacy_Icons), accessed 2013-05-05.
- [103] Orr Dunkelman, Nathan Keller, and Adi Shamir. A Practical-Time Attack on the A5/3 Cryptosystem Used in Third Generation GSM Telephony, 2010. URL: <http://eprint.iacr.org/2010/013>.
- [104] Orr Dunkelman, Nathan Keller, and Adi Shamir. A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony. *Journal of Cryptology*, 27(4):824–849, 2014. URL: <http://dx.doi.org/10.1007/s00145-013-9154-9>, doi:10.1007/s00145-013-9154-9.
- [105] Dynamic Demand. Dynamic Demand, 2017. accessed 2017-06-05. URL: <http://www.dynamicdemand.co.uk/grid.htm>.

- [106] P. Ekdahl and T. Johansson. Another attack on A5/1. *Information Theory, IEEE Transactions on*, 49(1):284–289, Jan 2003. doi:10.1109/TIT.2002.806129.
- [107] electronicsforu. How to Build: Cell Phone Jammer. <https://electronicsforu.com/electronics-projects/build-cell-phone-jammer>, October 2017. accessed 2018-03-02.
- [108] Stefan Emeis, Klaus Schafer, and CHRISTOPH Munkel. Surface-based remote sensing of the mixing-layer height a review. *Meteorologische Zeitschrift*, 17(5):621–630, 2008.
- [109] William Enck, Patrick Traynor, Patrick McDaniel, and Thomas La Porta. Exploiting Open Functionality in SMS-capable Cellular Networks. In *ACM Conference on Computer and Communications Security (CCS)*, pages 393–404. ACM, 2005.
- [110] Tobias Engel. Locating Mobile Phones using Signalling System 7. In *Chaos Communication Congress*, 25C3, dec 2008. URL: [http://events.ccc.de/congress/2008/Fahrplan/attachments/1262\\_25c3-locating-mobile-phones.pdf](http://events.ccc.de/congress/2008/Fahrplan/attachments/1262_25c3-locating-mobile-phones.pdf).
- [111] Tobias Engel. SS7: Locate. Track. Manipulate. In *Chaos Communication Congress*, 31C3, 2014. URL: <https://berlin.ccc.de/~tobias/31c3-ss7-locate-track-manipulate.pdf>.
- [112] Ericsson. Handling of Signaling Storms in Mobile Networks: The Role of the User Data Management System. Technical report, Ericsson, 2015. URL: <https://www.ericsson.com/res/docs/2015/handling-of-signaling-storms-in-mobile-networks-august.pdf>.
- [113] ETSI. Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Numbering, addressing and identification, 2014. URL: [http://www.etsi.org/deliver/etsi\\_ts/123000\\_123099/123003/12.04.01\\_60/ts\\_123003v120401p.pdf](http://www.etsi.org/deliver/etsi_ts/123000_123099/123003/12.04.01_60/ts_123003v120401p.pdf).
- [114] Ettus Research. Universal Software Radio Peripheral. <https://www.ettus.com/product>. accessed 2017-05-22.
- [115] European Network of Transmission System Operators for Electricity. Analysis of CE Inter-Area Oscillations Of 19 and 24 February 2014, 2011. URL: [https://www.entsoe.eu/fileadmin/user\\_upload/\\_library/publications/entsoe/RG\\_SOC\\_CE/Top7\\_110913\\_CE\\_inter-area-oscil\\_feb\\_19th\\_24th\\_final.pdf](https://www.entsoe.eu/fileadmin/user_upload/_library/publications/entsoe/RG_SOC_CE/Top7_110913_CE_inter-area-oscil_feb_19th_24th_final.pdf).
- [116] European Network of Transmission System Operators for Electricity. Report on Blackout in Turkey on 31st March 2015, 2015.



- [117] European Network of Transmission System Operators For Electricity. Power Statistics, 2017. accessed 2017-06-06. URL: <https://www.entsoe.eu/data/statistics/Pages/default.aspx>.
- [118] E:V:A (pseudonym) et al. Galaxy S III - "secret codes" and hidden features, 2012. Pseudonymized online discussion forum, <http://forum.xda-developers.com/showthread.php?t=1687249>, accessed 2013-07-14.
- [119] Cyrus Farivar. Apple removes GPS functionality from Egyptian iPhones, 2008. [http://www.macworld.com/article/1137410/Apple\\_removes\\_GPS\\_func.html](http://www.macworld.com/article/1137410/Apple_removes_GPS_func.html).
- [120] FirstNet. FirstNet: First Responder Network Authority. <http://www.firstnet.gov/>. accessed 2017-05-22.
- [121] Dan Forsberg, Huang Leping, Kashima Tsuyoshi, and Seppo Alanärä. Enhancing Security and Privacy in 3GPP E-UTRAN Radio Interface. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2007.
- [122] Forum Netztechnik. Technische Anforderungen an die automatische Frequenzentlastung, 2012. In German.
- [123] Andrea Frome, German Cheung, Ahmad Abdulkader, Marco Zennaro, Bo Wu, Alessandro Bissacco, Hartwig Adam, Hartmut Neven, and Luc Vincent. Large-scale Privacy Protection in Google Street View. In *IEEE International Conference on Computer Vision*, 2009.
- [124] Gamma Group. 3G-GSM Interception & Target Location. Sales brochure. <https://info.publicintelligence.net/Gamma-GSM.pdf>, accessed 2015-11-02.
- [125] General Electric Appliances. GE WiFi Connect - Air Conditioners, 2017. accessed 2017-06-04. URL: <http://www.geappliances.com/ge/connected-appliances/air-conditioners.htm>.
- [126] G-NetTrack phone measurement capabilities. <http://www.gyokovsolutions.com/survey/surveyresults.php>, accessed 2013-07-15.
- [127] Younghwan Go, Eunyong Jeong, Jongil Won, Yongdae Kim, Denis Foo Kune, and Kyoungsoo Park. Gaining Control of Cellular Traffic Accounting by Spurious TCP Retransmission. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2014.
- [128] Younghwan Go, Denis Foo Kune, Shinae Woo, KyoungSoo Park, and Yongdae Kim. Towards Accurate Accounting of Cellular Data for TCP Retransmission. In *ACM Workshop on Mobile Computing Systems and Applications (HotMobile)*. ACM, 2013.

- [129] Thomas Gobmaier. Measurement of the mains frequency, 2017. accessed 2017-06-05. URL: <http://www.mainsfrequency.com/>.
- [130] Ian Goldberg, David Wagner, and Lucky Green. The (Real-Time) Cryptanalysis of A5/2. In *Rump session of Crypto'99*, 1999.
- [131] Nico Golde and Daniel Komaromy. Breaking Band. In *Recon*, jun 2016. URL: [https://comsecuris.com/slides/recon2016-breaking\\_band.pdf](https://comsecuris.com/slides/recon2016-breaking_band.pdf).
- [132] Nico Golde, Kévin Redon, and Jean-Pierre Seifert. Let Me Answer That For You: Exploiting Broadcast Information in Cellular Networks. In *USENIX Security Symposium (SSYM)*, pages 33–48. USENIX Association, 2013.
- [133] Jovan Dj. Golić. Cryptanalysis of Alleged A5 Stream Cipher. In *International Conference on Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pages 239–255. Springer, 1997.
- [134] Ismael Gomez-Miguel, Andres Garcia-Saavedra, Paul D Sutton, Pablo Serrano, Cristina Cano, and Doug J Leith. srsLTE: an open-source platform for LTE evolution and experimentation. In *ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, pages 25–32. ACM, 2016.
- [135] Dan Goodin. Software Flaw puts Mobile Phones and Networks at Risk of Complete Takeover, July 2016. URL: <https://arstechnica.com/security/2016/07/software-flaw-puts-mobile-phones-and-networks-at-risk-of-complete-takeover/>.
- [136] Travis Goodspeed, Sergey Bratus, Ricky Melgares, Rebecca Shapiro, and Ryan Speers. Packets in packets: Orson welles' in-band signaling attacks for modern radios. In *Proceedings to WOOT 2011*, pages 54–61, August 2011.
- [137] Google Firebase. Firebase cloud messaging. accessed 2017-04-09.
- [138] Adrian W. Graham, Nicholas C. Kirkman, and Peter M. Paul. *Mobile Radio Network Design in the VHF and UHF Bands*. John Wiley & Sons Ltd, 2007.
- [139] GSM Association. IR.50 2G 2.5G 3G Roaming v4.0, 2008. <http://www.gsma.com/newsroom/all-documents/ir-50-2g2-5g3g-roaming/>, accessed 2015-11-25.
- [140] Prohibiting A5/2 in mobile stations and other clarifications regarding A5 algorithm support. [http://www.3gpp.org/ftp/tsg\\_sa/TSG\\_SA/TSGS\\_37/Docs/SP-070671.zip](http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_37/Docs/SP-070671.zip).
- [141] GSMA Press Release. Number of Mobile Subscribers Worldwide Hits 5 Billion. June 2017, <https://www.gsma.com/newsroom/press-release/number-mobile-subscribers-worldwide-hits-5-billion/>, accessed 2018-03-10.

- [142] GSMK mbH. GSMK debuts new security systems to protect mobile network operators against eavesdropping and fraud. <http://www.cryptophone.de/en/company/news/gsmk-debuts-new-security-systems-to-protect-mobile-network-operators-against-eavesdropping-and-fraud/>, 2017. accessed 2017-04-05.
- [143] Payas Gupta, Bharat Srinivasan, Vijay Balasubramaniyan, and Mustaque Ahamad. Phoneyptot : Data-driven Understanding of Telephony Threats. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2015.
- [144] Mordechai Guri, Yisroel Mirsky, and Yuval Elovici. 9-1-1 DDoS via Malicious Baseband Firmware. In *IEEE European Symposium on Security and Privacy (EuroSP)*. IEEE, 2017.
- [145] Tim Güneysu, Timo Kasper, Martin Novotny, Christof Paar, and Andy Rupp. Cryptanalysis with COPACOBANA. *IEEE Transaction on Computers*, 57(11), November 2008.
- [146] Paulina Haduong, Anthony Tordillos, and Machiste Quintana. Privacy Simplified - Icons, 2012. <http://yale.edu/self/psicons.html>, accessed 2013-05-05.
- [147] Chan-Kyu Han and Hyoung-Kee Choi. Security Analysis of Handover Key Management in 4G LTE/SAE Networks. *IEEE Transactions on Mobile Computing (TMC)*, 13(2):457–468, 2014.
- [148] Adam Harvey. CV Dazzle, 2010-2012. <http://ahprojects.com/projects/cv-dazzle>, accessed 2013-05-02.
- [149] Benjamin Henne, Christian Szongott, and Matthew Smith. Snapme if you can: privacy threats of other peoples’ geo-tagged media and what we can do about it. In *Proceedings of the sixth ACM conference on Security and privacy in wireless and mobile networks*, pages 95–106. ACM, 2013.
- [150] C. Herley and P. C. v. Oorschot. Sok: Science, security and the elusive goal of security as a scientific pursuit. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 99–120, May 2017. doi:10.1109/SP.2017.38.
- [151] J. Hodges, C. Jackson, and A. Barth. HTTP Strict Transport Security (HSTS). RFC-6797, 2012.
- [152] S. Holtmanns and I. Oliver. SMS and one-time-password interception in LTE networks. In *IEEE International Conference on Communications (ICC)*. IEEE, 2017.
- [153] Silke Holtmanns, Siddharth Prakash Rao, and Ian Oliver. User Location Tracking Attacks for LTE Networks using the Interworking Functionality. In *IFIP Networking Conference and Workshops*, pages 315–322, 2016.

- [154] Byeongdo Hong, Sangwook Bae, and Yongdae Kim. GUTI Reallocation Demystified: Cellular Location Tracking with Changing Temporary Identifier. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2018.
- [155] Hyunwook Hong, Hyunwoo Choi, Dongkwan Kim, Hongil Kim, Byeongdo Hong, Jiseong Noh, and Yongdae Kim. When Cellular Networks Met IPv6: Security Problems of Middleboxes in IPv6 Cellular Networks. In *IEEE European Symposium on Security and Privacy (EuroSP)*. IEEE, 2017.
- [156] David Hulton and Steve. Cracking GSM. Black Hat DC 2008, 03 2008.
- [157] Tim Hummel and Linus Neumann. Xgoldscanner, 12 2013. <https://opensource.srlabs.de/projects/mobile-network-assessment-tools/wiki/Xgoldscanner>, accessed 2014-02-19.
- [158] Nicolas Höning. Remote "shut-off" option cancelled for Dutch smart meters, 2013. accessed 2017-09-24. URL: <https://www.nicolashoening.de/?energy&nr=238>.
- [159] Ia-sadosky. Heap memory corruption in ASN.1 parsing code generated by Objective Systems Inc. ASN1C compiler for C/C++, 2016. URL: <https://github.com/programa-stic/security-advisories/tree/master/ObjSys/CVE-2016-5080>.
- [160] ISO/IEC 24778: Information technology – Automatic identification and data capture techniques – Aztec Code bar code symbology specification.
- [161] Suman Jana and Vitaly Shmatikov. Abusing File Processing in Malware Detectors for Fun and Profit. In *Proceedings of the 33rd IEEE Symposium on Security & Privacy*, San Francisco, CA, May 2012.
- [162] Dongseok Jang, Ranjit Jhala, Sorin Lerner, and Hovav Shacham. An Empirical Study of Privacy-violating Information Flows in JavaScript Web Applications. In *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS '10*, pages 270–283, New York, NY, USA, 2010. ACM. URL: <http://doi.acm.org/10.1145/1866307.1866339>, doi:10.1145/1866307.1866339.
- [163] Markus Jaschinsky. aktuelle Netzfrequenz (47,5-52,5 Hz) - Netzfrequenz.info, 2017. accessed 2017-06-05. URL: <https://www.netzfrequenz.info/aktuelle-netzfrequenz-full>.
- [164] Keting Jia, Christian Rechberger, and Xiaoyun Wang. Green Cryptanalysis: Meet-in-the-Middle Key-Recovery for the Full KASUMI Cipher. Cryptology ePrint Archive, Report 2011/466, 2011. <http://eprint.iacr.org/2011/466>.
- [165] Alberto Escalada Jimenez, Adrian Dabrowski, Noburu Sonehara, Juan M Montero Martinez, and Isao Echizen. Tag Detection for Preventing Unauthorized Face Image

- Processing. In *Proceedings of the 13th International Workshop on Digital-Forensics and Watermarking (IWDW 2014)*. LNCS, Springer, 10 2014.
- [166] Frick Joachim and Bott Rainer. Method for identifying a mobile phone user or for eavesdropping on outgoing calls, 2000. Patent, Rohde & Schwarz, EP1051053.
- [167] Joint Research Centre of the European Commission. Smart Metering deployment in the European Union, 2017. URL: <http://ses.jrc.ec.europa.eu/smart-metering-deployment-european-union>.
- [168] Roger Piqueras Jover. Security Attacks against the Availability of LTE Mobility Networks: Overview and Research Directions. In *IEEE Symposium on Wireless Personal Multimedia Communications (WPMC)*. IEEE, 2013.
- [169] Roger Piqueras Jover. LTE security, protocol exploits and location tracking experimentation with low-cost software radio. *arXiv*, 2016. URL: <http://arxiv.org/abs/1607.05171>, arXiv:1607.05171.
- [170] Mateusz Kajstura, Agata Trawinska, and Jacek Hebenstreit. Application of the Electrical Network Frequency (ENF) Criterion. *Forensic Science International*, 155(2):165–171, 2005. URL: <http://dx.doi.org/10.1016/j.forsciint.2004.11.015>, doi:10.1016/j.forsciint.2004.11.015.
- [171] Dejan Kaljevic. SIMSCAN v2.01. <http://dejankaljevic.org/>, accessed 2017-03-09, 2003.
- [172] G Kambourakis, C Koliass, S Gritzalis, and J Park. DoS Attacks Exploiting Signaling in UMTS and IMS. *Computer Communications*, 34(3):226–235, 2011.
- [173] Auguste Kerckhoffs. La cryptographie militaire. *Journal des Sciences Militaires*, pages 5–83, 1883.
- [174] Kerem Erkan. Qrafter. Apple App Store. <https://itunes.apple.com/us/app/id416098700>, accessed July 17th 2014.
- [175] Amin Kharraz, Engin Kirda, William Robertson, Davide Balzarotti, and Aurelien Francillon. Optical Delusions: A Study of Malicious QR Codes in the Wild. In *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 06 2014.
- [176] Peter Kieseberg, Sebastian Schrittwieser, Manuel Leithner, Martin Mulazzani, Edgar Weippl, Lindsay Munroe, and Mayank Sinha. Malicious Pixels Using QR Codes as Attack Vector. In Ismail Khalil and Teddy Mantoro, editors, *Trustworthy Ubiquitous Computing*, volume 6 of *Atlantis Ambient and Pervasive Intelligence*, pages 21–38. Atlantis Press, 2012. URL: [http://dx.doi.org/10.2991/978-94-91216-71-8\\_2](http://dx.doi.org/10.2991/978-94-91216-71-8_2), doi:10.2991/978-94-91216-71-8\_2.

- [177] Hongil Kim, Dongkwan Kim, Minhee Kwon, Hyungseok Han, Yeongjin Jang, Dongsu Han, Taesoo Kim, and Yongdae Kim. Breaking and Fixing VoLTE : Exploiting Hidden Data Channels and Misimplementations. In *ACM Conference on Computer and Communications Security (CCS)*, pages 328–339. ACM, 2015.
- [178] Christian Krieg, Adrian Dabrowski, Heidelinde Hobel, Katharina Krombholz, and Edgar Weippl. Hardware Malware. *Synthesis Lectures on Information Security, Privacy, and Trust*, 4(2):1–115, 2013. URL: <http://www.morganclaypool.com/doi/abs/10.2200/S00530ED1V01Y201308SPT006>, arXiv:<http://www.morganclaypool.com/doi/pdf/10.2200/S00530ED1V01Y201308SPT006>, doi:10.2200/S00530ED1V01Y201308SPT006.
- [179] Katharina Krombholz, Adrian Dabrowski, Matthew Smith, and Edgar Weippl. Exploring Design Directions for Wearable Privacy. In *Proceedings of USEC Mini Conference 2017*. Internet Society, 02 2017. URL: <https://www.internetsociety.org/doc/exploring-design-directions-wearable-privacy>.
- [180] Katharina Krombholz, Adrian Dabrowski, Matthew Smith, and Edgar R. Weippl. Ok Glass, Leave me Alone: Towards a Systematization of Privacy Enhancing Technologies for Wearable Computing. In *Proceedings of Workshop on Wearable Security and Privacy co-located with Financial Cryptography and Data Security 2015*, 01 2015.
- [181] Denis Foo Kune, John Koelndorfer, Nicholas Hopper, and Yongdae Kim. Location Leaks on the GSM Air Interface. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2012.
- [182] Ulrich Kühn. Cryptanalysis of Reduced-Round MISTY. In *Advances in Cryptology – EUROCRYPT 2001*, pages 325–339. Springer Verlag, 2001.
- [183] SR Labs. Kraken: A5/1 Decryption Rainbow Tables. via Bittorent, 2010. <https://opensource.srlabs.de/projects/a51-decrypt>, accessed 2015-11-12.
- [184] Fabian Lanze, Andriy Panchenko, Ignacio Ponce-Alcaide, and Thomas Engel. Undesired relatives: protection mechanisms against the evil twin attack in IEEE 802.11. In *Proceedings of the 10th ACM symposium on QoS and security for wireless and mobile networks*, pages 87–94. ACM, 2014.
- [185] Mats Larsson, Walter Sattinger, Luis-Fabiano Santos, and Roland Notter. *2013 IEEE Power & Energy Society General Meeting*, chapter Practical Experience with Modal Estimation Tools at Swissgrid. Institute of Electrical and Electronics Engineers, 2013. URL: <https://library.e.abb.com/public/503f299a520c490c972def08598f6d7b/Practical%20Experience%20with%20Modal%20Estimation.pdf>.

- [186] Patrick P. C. Lee, Tian Bu, and Thomas Woo. On the Detection of Signaling DoS Attacks on 3G/WiMax Wireless Networks. *Computer Networks*, 53(15):2601–2616, 2009.
- [187] Sangho Lee, Youngsok Kim, Jangwoo Kim, and Jong Kim. Stealing Webpages Rendered on Your Browser by Exploiting GPU Vulnerabilities. In *Security and Privacy (SP), 2014 IEEE Symposium on*, pages 19–33, May 2014. doi:10.1109/SP.2014.9.
- [188] Wai Kay Leong, Aditya Kulkarni, Yin Xu, and Ben Leong. Unveiling the Hidden Dangers of Public IP locations in 4G/LTE Cellular Data Networks. In *ACM Workshop on Mobile Computing Systems and Applications (HotMobile)*, pages 16:1–16:6. ACM, 2014.
- [189] LG. LG Smart AC with mobile app, 2017. accessed 2017-06-04. URL: <http://www.lg-dfs.com/smartac.aspx>.
- [190] Chi-Yu Li, Guan-Hua Tu, Songwu Lu, Xinbing Wang, Chunyi Peng, Zengwen Yuan, Yuanjie Li, Songwu Lu, and Xinbing Wang. Insecurity of Voice Solution VoLTE in LTE Mobile Networks. In *ACM Conference on Computer and Communications Security (CCS)*, pages 316–327. ACM, 2015.
- [191] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu. Securing smart grid: cyber attacks, countermeasures, and challenges. *IEEE Communications Magazine*, 50(8):38–45, August 2012. doi:10.1109/MCOM.2012.6257525.
- [192] Zhenhua Li, Weiwei Wang, Christo Wilson, Jian Chen, Chen Qian, Taeho Jung, Lan Zhang, Kebin Liu, Xiangyang Li, and Yunhao Liu. FBS-Radar: Uncovering Fake Base Stations at Scale in the Wild. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2017.
- [193] Marc Lichtman, Roger Piqueras Jover, Mina Labib, Raghunandan Rao, Vuk Marojevic, and Jeffrey H. Reed. LTE/LTE-a Jamming, Spoofing, and Sniffing: Threat Assessment and Mitigation. *IEEE Communications Magazine*, 54(4):54–61, 2016.
- [194] Marc Lichtman, Jeffrey H Reed, T Charles Clancy, and Mark Norton. Vulnerability of LTE to Hostile Interference. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 285–288. IEEE, 2013.
- [195] Junrong Liu, Yu Yu, François Xavier Standaert, Zheng Guo, Dawu Gu, Wei Sun, Yijie Ge, and Xinjun Xie. Small Tweaks Do Not Help: Differential Power Analysis of MILENAGE Implementations in 3G/4G USIM Cards. In *European Symposium on Research in Computer Security (ESORICS)*, pages 468–480. Springer, 2015.
- [196] X. Liu and Z. Li. Local Load Redistribution Attacks in Power Systems With Incomplete Network Information. *IEEE Transactions on Smart Grid*, 5(4):1665–1676, 2014.

- [197] X. Liu, Z. Li, and Z. Li. Optimal Protection Strategy Against False Data Injection Attacks in Power Systems. *IEEE Transactions on Smart Grid*, pages 1–9, 2016.
- [198] Shimon Machida, Adrian Dabrowski, Edgar Weippl, and Isao Echizen. PrivacyTag: A Community-Based Method for Protecting Privacy of Photographed Subjects in Online Social Networks. In Arpan Kumar Kar, P. Vigneswara Ilavarasan, M.P. Gupta, Yogesh K. Dwivedi, Matti Mäntymäki, Marijn Janssen, Antonis Simintiras, and Salah Al-Sharhan, editors, *Digital Nations – Smart Cities, Innovation, and Sustainability*, pages 261–275, Cham, 2017. Springer International Publishing.
- [199] Jonas Magazinius, Billy K Rios, and Andrei Sabelfeld. Polyglots: crossing origins by crossing formats. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 753–764. ACM, 2013.
- [200] Luca Malette. Catcher Catcher. accessed 2013-07-12. URL: <https://opensource.srlabs.de/projects/catcher>.
- [201] Luca Malette. Catcher Catcher, 2015. accessed 2016-01-18. URL: <https://opensource.srlabs.de/projects/mobile-network-assessment-tools/wiki/CatcherCatcher>.
- [202] K. Manandhar, X. Cao, F. Hu, and Y. Liu. Detection of Faults and Attacks Including False Data Injection Attack in Smart Grid Using Kalman Filter. *IEEE Transactions on Control of Network Systems*, 1(4):370–379, 2014.
- [203] MaxMind Inc. GeoIP Products, 2017. accessed 2017-05-30. URL: <http://dev.maxmind.com/geoip/>.
- [204] Ulrike Meyer and Susanne Wetzel. A Man-in-the-Middle Attack on UMTS. In *ACM Workshop on Wireless Security (WiSe)*, pages 90–97. ACM, 2004.
- [205] Ulrike Meyer and Susanne Wetzel. On the Impact of GSM Encryption and Man-in-the-Middle Attacks on the Security of Interoperating GSM/UMTS Networks. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PRIMRC)*, pages 2876–2883. IEEE, 2004.
- [206] Ulrike Meyer and Susanne Wetzel. A Man-in-the-Middle Attack on UMTS. In *3rd ACM workshop on Wireless security*, pages 90–97, 2005. URL: <http://www.cs.stevens.edu/swetzel/publications/mim.pdf>.
- [207] Benoit Michau and Christophe Devine. How to not Break LTE Crypto. In *ANSSI Symposium sur la sécurité des technologies de l’information et des communications (SSTIC)*, 2016.
- [208] Microsoft Corporation. Microsoft Tag - Implementation Guide, 2011. <http://tag.microsoft.com/resources/implementation-guide.aspx>, accessed 2013-05-09.



- [209] D. Mills. Network Time Protocol (Version 3) Specification, Implementation and Analysis. RFC 1305 (Draft Standard), March 1992. Obsoleted by RFC 5905. URL: <https://www.rfc-editor.org/rfc/rfc1305.txt>, doi:10.17487/RFC1305.
- [210] David L. Mills. Clock Discipline Algorithm, 2014. accessed 2017-05-30. URL: <https://www.eecis.udel.edu/~mills/ntp/html/discipline.html>.
- [211] Miniwatts Marketing Group. World Internet Users Statistics and 2017 World Population Stats, 2017. accessed 2017-09-22. URL: <http://www.internetworldstats.com/stats.htm>.
- [212] S. Mishra, X. Li, A. Kuhnle, M. T. Thai, and J. Seo. Rate alteration attacks in smart grid. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2353–2361, 2015.
- [213] Mjølunes, Stig F. and Olimid, Ruxandra F. Easy 4G/LTE IMSI Catchers for Non-Programmers. In *Conference on Mathematical Methods, Models, and Architectures for Computer Network Security (MMM-ACNS)*, pages 235–246. Springer, 2017.
- [214] A. H. Mohsenian-Rad and A. Leon-Garcia. Distributed Internet-Based Load Altering Attacks Against Smart Power Grids. *IEEE Transactions on Smart Grid*, 2(4):667–674, Dec 2011. doi:10.1109/TSG.2011.2160297.
- [215] Motorola Inc. Symbol DS6708 Digital Scanner Product Reference Guide, 2009. [http://www.motorolasolutions.com/web/Business/Products/Barcode%20Code%20Scanning/Barcode%20Code%20Scanners/General%20Purpose%20Scanners/\\_Documents/static\\_file/ds6708.pdf](http://www.motorolasolutions.com/web/Business/Products/Barcode%20Code%20Scanning/Barcode%20Code%20Scanners/General%20Purpose%20Scanners/_Documents/static_file/ds6708.pdf).
- [216] Collin Mulliner, Nico Golde, and Jean-Pierre Seifert. SMS of Death: from Analyzing to Attacking Mobile Phones on a Large Scale. In *USENIX Security Symposium (SSYM)*, pages 363–378. USENIX Association, 2011.
- [217] Phil Muncaster. Chinese cops cuff 1,500 in fake base station spam raid. The Register, 26 Mar 2014. [http://www.theregister.co.uk/2014/03/26/spam\\_text\\_china\\_clampdown\\_police/](http://www.theregister.co.uk/2014/03/26/spam_text_china_clampdown_police/).
- [218] Steven J. Murdoch. Software detection of currency. University of Cambridge, <http://www.cl.cam.ac.uk/~sjm217/projects/currency/>, accessed 2013-05-07.
- [219] Bhavesh Naik. QR Code: USSD attack, 2012. <http://resources.infosecinstitute.com/qr-code-ussd-attack/>, accessed July 18th 2014.
- [220] NationalGridUSA Service Company, Inc. Electricity Transmission Operational Data, 2017. URL: <http://www2.nationalgrid.com/uk/industry-information/electricity-transmission-operational-data/>.

- [221] NeoMedia Technologies, Inc. NeoReader. Apple App Store. <https://itunes.apple.com/us/app/id284973754>, accessed July 17th 2014.
- [222] NeoMedia Technologies Inc. NeoReader QR & Barcode Scanner. Google Play Store. <https://play.google.com/store/apps/details?id=de.gavitec.android>, accessed July 17th 2014.
- [223] Nest Labs, Inc. Meet the Nest Learning Thermostat, 2017. accessed 2017-06-04. URL: <https://nest.com/thermostat/meet-nest-thermostat/>.
- [224] Matthias Neugschwandtner, Martina Lindorfer, and Christian Platzer. A View to a Kill: WebView Exploitation. In *Presented as part of the 6th USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*, 2013.
- [225] Navid Nikaein, Raymond Knopp, Florian Kaltenberger, Lionel Gauthier, Christian Bonnet, Dominique Nussbaum, and Riadh Ghaddab. OpenAirInterface: An Open LTE Network in a PC. In *ACM International Conference on Mobile Computing and Networking (MobiCom)*, pages 305–308. ACM, 2014.
- [226] Karsten Nohl. Breaking GSM phone privacy. Blackhat 2010.
- [227] Karsten Nohl. Rooting SIM cards. In *Blackhat*, 2013. URL: <https://media.blackhat.com/us-13/us-13-Nohl-Rooting-SIM-cards-Slides.pdf>.
- [228] Karsten Nohl. Mobile Self-Defense. In *Chaos Communication Congress*, 31C3, 2014. URL: [https://events.ccc.de/congress/2014/Fahrplan/system/attachments/2493/original/Mobile\\_Self\\_Defense-Karsten\\_Nohl-31C3-v1.pdf](https://events.ccc.de/congress/2014/Fahrplan/system/attachments/2493/original/Mobile_Self_Defense-Karsten_Nohl-31C3-v1.pdf).
- [229] Karsten Nohl and Luca Melette. Defending Mobile Phones. Chaos Communications Congress (28C3), 2011. URL: <https://events.ccc.de/congress/2011/Fahrplan/events/4736.en.html>.
- [230] Karsten Nohl and Luca Melette. GPRS Intercept: Wardriving your country. Chaos Communications Camp 2011, 2011.
- [231] Karsten Nohl and Sylvain Munaut. Wideband GSM Sniffing. Chaos Communications Congress (27C3), 2010.
- [232] Karsten Nohl and Chris Paget. GSM – SRSLY? In *Chaos Communication Congress*, 26C3, dec 2009.
- [233] Alexey Osipov and Alexander Zaitsev. Adventures in Femtoland: 350 Yuan for Invaluable Fun. Black Hat USA 2015, 08 2015.
- [234] Osmocom Project. OsmocomBB Open Source GSM Baseband Software Implementation. <http://bb.osmocom.org>. accessed 2016-07-12.

- [235] Osmocom Project. RTL-SDR - osmcomSDR. <http://sdr.osmocom.org/trac/wiki/rtl-sdr>, accessed 2014-03-05.
- [236] Hirofumi Otori and Shigeru Kuriyama. Data-Embeddable Texture Synthesis. In Andreas Butz, Brian Fisher, Antonio Krüger, Patrick Olivier, and Shigeru Owada, editors, *Smart Graphics*, volume 4569 of *Lecture Notes in Computer Science*, pages 146–157. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-73213-6. doi:10.1007/978-3-540-73214-3\_13.
- [237] Panagiotis Papantonakis, Dionisios Pnevmatikatos, and Ioannis Papaefstathiou. Fast, FPGA-based Rainbow Table creation for attacking encrypted mobile communications. In *Field Programmable Logic and Applications (FPL)*. IEEE, 2013.
- [238] Christopher Parsons. Thinking About a ‘Privacy Commons’, Nov 2009. <http://www.christopher-parsons.com/thinking-about-a-privacy-commons/>, accessed 2013-05-05.
- [239] Darren Pauli. Connected kettles boil over, spill Wi-Fi passwords over London, 2015. accessed 2017-05-04. URL: [https://www.theregister.co.uk/2015/10/19/bods\\_brew\\_ikettle\\_20\\_hack\\_plot\\_vulnerable\\_london\\_pots/](https://www.theregister.co.uk/2015/10/19/bods_brew_ikettle_20_hack_plot_vulnerable_london_pots/).
- [240] Stephanie K. Pell and Christopher Soghoian. Your secret stingray’s no secret anymore: The vanishing government monopoly over cell phone surveillance and its impact on national security and consumer privacy. *Harvard Journal of Law & Technology*, 28(1):1–76, 2014.
- [241] Chunyi Peng, Chi-yu Li, Guan-Hua Tu, Songwu Lu, and Lixia Zhang. Mobile Data Charging: New Attacks and Countermeasures. In *ACM Conference on Computer and Communications Security (CCS)*, pages 195–204. ACM, 2012.
- [242] Chunyi Peng, Chi-Yu Li, Hongyi Wang, Guan-Hua Tu, and Songwu Lu. Real Threats to Your Data Bills: Security Loopholes and Defenses in Mobile Data Charging. In *ACM Conference on Computer and Communications Security (CCS)*, pages 727–738. ACM, 2014.
- [243] Kevin Peng, Harry Sanabria, Derek Wu, and Charlotte Zhu. Security Overview of QR Codes. 2014. MIT Student Paper, available online <https://courses.csail.mit.edu/6.857/2014/files/12-peng-sanabria-wu-zhu-qr-codes.pdf>.
- [244] Guy Philippe, François Montaigne, Jean-christophe Schiel, Eric Georgeaux, Christophe Gruet, Pierre-yves Roy, Pierre Force, and Philippe Mège. LTE Resistance to Jamming Capability. In *Military Communications and Information Systems Conference (MCC)*, pages 7–9, 2013.
- [245] PKI Electronic Intelligence GmbH Germany. 3G UMTS IMSI Catcher. <http://www.pki-electronic.com/products/interception-and->

- monitoring-systems/3g-umts-imsi-catcher/, 2017. accessed 2017-04-03.
- [246] Phillip Porras, Hassen Saidi, and Vinod Yegneswaran. An Analysis of Conficker’s Logic and Rendezvous Points. Technical report, SRI International, 2009. accessed 2017-05-30. URL: <http://www.csl.sri.com/users/vinod/papers/Conficker/>.
- [247] Bernhard Postl. IMSI Catcher. Master’s thesis, Technikum Wien, 2012.
- [248] Vassilis Prevelakis and Diomidis Spinellis. The Athens Affair. *IEEE Spectrum*, 44(7):26–33, 2007.
- [249] provenweb (pseudonym). Is it possible to detect the Android captive portal browser? <http://stackoverflow.com/questions/32950326/is-it-possible-to-detect-the-android-captive-portal-browser>, accessed 2015-01-20.
- [250] Jordan Rabet. NINJHAX - 3DS Homebrew Exploit. <http://smealum.net/ninjhax/>, accessed March 3rd 2015.
- [251] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monroe, and Andreas Terzis. My Botnet is Bigger Than Yours (Maybe, Better Than Yours): Why Size Estimates Remain Challenging. In *Proceedings of the First Conference on First Workshop on Hot Topics in Understanding Botnets*, HotBots’07, pages 5–5, Berkeley, CA, USA, 2007. USENIX Association.
- [252] RangeNetworks. OpenBTS. <http://openbts.org/>. accessed 2017-04-05.
- [253] J.R. R Rao, P. Rohatgi, H. Scherzer, and S. Tinguely. Partitioning Attacks: or how to Rapidly Clone some GSM Cards. In *IEEE Symposium on Security and Privacy (SP)*, pages 31–41. IEEE, 2002.
- [254] Raghunandan M Rao, Sean Ha, Vuk Marojevic, and Jeffrey H Reed. LTE PHY Layer Vulnerability Analysis and Testing Using Open-Source SDR Tools. *arXiv preprint arXiv:1708.05887*, 2017.
- [255] Siddharth Prakash Rao, Silke Holtmanns, Ian Oliver, and Tuomas Aura. Unblocking Stolen Mobile Devices using SS7-MAP Vulnerabilities: Exploiting the Relationship between IMEI and IMSI for EIR access. In *IEEE Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1171–1176. IEEE, 2015.
- [256] Siddharth Prakash Rao, Bhanu Teja Kotte, and Silke Holtmanns. Privacy in LTE networks. In *ICST International Conference on Mobile Multimedia Communications*, pages 176–183, 2016.

- [257] Regional Group Continental Europe and Synchronous Area Great Britain. Solar Eclipse 2015 - Impact Analysis, 2015.
- [258] Richard's wireless blog. Hidden menus in Android phone, 2009. [http://rwireless.blogspot.co.at/2009\\_03\\_23\\_archive.html](http://rwireless.blogspot.co.at/2009_03_23_archive.html), accessed 2013-07-14.
- [259] Rafael A. Rodríguez-Gómez, Gabriel Maciá-Fernández, and Pedro García-Teodoro. Survey and Taxonomy of Botnet Research Through Life-cycle. *ACM Comput. Surv.*, 45(4):45:1–45:33, August 2013.
- [260] Rohde & Schwarz. Countering threats early on. [www.idexuae.ae/ExhibitorLibrary/1328/Countering\\_threats\\_early\\_on\\_2.pdf](http://www.idexuae.ae/ExhibitorLibrary/1328/Countering_threats_early_on_2.pdf), accessed 2013-07-14.
- [261] Jesse Ruderman. Bug 57351 - css on a:visited can load an image and/or reveal if visitor been to a site, 2000. [http://bugzilla.mozilla.org/show\\_bug.cgi?id=57351](http://bugzilla.mozilla.org/show_bug.cgi?id=57351), accessed 2016-01-21.
- [262] Mary Rundle. International Data Protection and Digital Identity Management Tools. Internet Governance Forum 2006, Privacy Workshop, Athens, 2006. Presentation Slides at <http://identityproject.lse.ac.uk/mary.pdf>, accessed 2013-05-05.
- [263] David Rupperecht, Adrian Dabrowski, Thorsten Holz, Edgar Weippl, and Christina Pöpper. On Security Research Towards Future Mobile Network Generations. *unpublished*, 2018. preprint available: <https://arxiv.org/abs/1710.08932>.
- [264] David Rupperecht, Kai Jansen, and Christina Pöpper. Putting LTE Security Functions to the Test: A Framework to Evaluate Implementation Correctness. In *USENIX Workshop on Offensive Technologies (WOOT)*. USENIX Association, 2016.
- [265] Merve Sahin, Aurélien Francillon, Payas Gupta, and Mustaque Ahamad. SoK: Fraud in Telephony Networks. In *IEEE European Symposium on Security and Privacy (EuroSP)*. IEEE, 2017.
- [266] Samsung. NE58K9850WG/AA - 5.8 cu. ft. Slide-In Electric Flex Duo Range with Dual Door, 2016. URL: <http://www.samsung.com/us/home-appliances/ranges/slide-in/ne58k9850wg-slide-in-electric-flex-duo-range-with-dual-door-black-stainless-steel-ne58k9850wg-aa/>.
- [267] Samsung. Family Hub Refrigerator, 2017. accessed 2017-06-04. URL: <http://www.samsung.com/us/explore/family-hub-refrigerator/>.
- [268] Len Sassaman, Meredith L. Patterson, Sergey Bratus, Michael E. Locasto, and Anna Shubina. Security Applications of Formal Language Theory. In *IEEE Systems Journal, Volume 7, Issue 3*, Sept. 2013.

- [269] Jeremy Scahill and Josh Begley. The Great SIM Heist - How Spies Stole the Keys to the Encryption Castle. *The Intercept*, Jan 2015. accessed 2018-03-02. URL: <https://theintercept.com/2015/02/19/great-sim-heist/>.
- [270] Scanbuy Inc. ScanLife Barcode & QR Code Reader with Prices, Deals, & Reviews. Apple App Store. <https://itunes.apple.com/us/app/scanlife-barcode-reader-qr/id285324287>, accessed July 17th 2014.
- [271] Scanbuy Inc. ScanLife QR & Barcode Reader. Google Play Store. <https://play.google.com/store/apps/details?id=com.ScanLife>, accessed July 17th 2014.
- [272] Jeremy Schiff, Marci Meingast, Deirdre K Mulligan, Shankar Sastry, and Ken Goldberg. Respectful cameras: Detecting visual markers in real-time to address privacy concerns. In *Protecting Privacy in Video Surveillance*, pages 65–89. Springer, 2009.
- [273] SecUpwN (Pseudonym, Maintainer). Android IMSI-Catcher Detector. <https://cellularprivacy.github.io/Android-IMSI-Catcher-Detector/>, retrieved 2017-04-04.
- [274] Security Reserach Labs GmbH. GSM security map. <http://gsmap.org/>.
- [275] Altaf Shaik, Ravishankar Borgaonkar, N. Asokan, Valtteri Niemi, and Jean-Pierre Seifert. Practical Attacks against Privacy and Availability in 4G/LTE Mobile Communication Systems. In *Symposium on Network and Distributed System Security (NDSS)*. The Internet Society, 2016.
- [276] Sajad Shirali-Shahreza and Mohammad Shirali-Shahreza. Steganography in Textiles. In *Proceedings of the 2008 The Fourth International Conference on Information Assurance and Security, IAS '08*, pages 56–61, Washington, DC, USA, 2008. IEEE Computer Society. URL: <http://dx.doi.org/10.1109/IAS.2008.11>, doi: 10.1109/IAS.2008.11.
- [277] ShopSavvy Inc. QR Code Reader and Scanner. Apple App Store. <https://itunes.apple.com/en/app/qr-code-reader-and-scanner/id388175979>, accessed July 17th 2014.
- [278] ShopSavvy Inc. ShopSavvy Barcode Scanner. Google Play Store. <https://play.google.com/store/apps/details?id=com.biggu.shopsavvy>, accessed July 17th 2014.
- [279] Mathew Solnik and Marc Blanchou. Cellular Exploitation on a Global Scale: The Rise and Fall of the Control Protocol. Blackhat 2014, Las Vegas, 2014.
- [280] Ashkan Soltani and Matt DeLong. How the NSA Pinpoints a Mobile Device. *Washington Post*, 2015. <http://apps.washingtonpost.com/g/page/world/how-the-nsa-pinpoints-a-mobile-device/645/>, accessed 2015-10-30.

- [281] SR Labs. Snoopsnitch, 12 2014. <https://opensource.srlabs.de/projects/snoopsnitch>, accessed 2015-11-12.
- [282] Statista, Inc. Number of households in the U.S. from 1960 to 2016 (in millions), 2016. accessed 2017-06-07. URL: <https://www.statista.com/statistics/183635/number-of-households-in-the-us/>.
- [283] Matthias Strubel and Sébastien Pierre. iOS 9 & Android with offline networks. <http://librelist.com/browser//off.networks/2015/10/14/ios9-android-with-offline-networks/>, accessed 2015-01-20.
- [284] tado GmbH. Smart heating control, 2017. accessed 2017-06-04. URL: <https://www.tado.com/>.
- [285] Donna Tam. PayPal offers QR codes for retail-store purchases, October 2013. <http://www.cnet.com/news/paypal-offers-qr-codes-for-retail-store-purchases/>, accessed July 24th 2014.
- [286] V.F. Taylor, R. Spolaor, M. Conti, and I. Martinovic. Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic. In *1st IEEE European Symposium on Security and Privacy (Euro S&P 2016)*, March 2016. To appear.
- [287] Telit Wireless Solutions. GT864-QUAD/PY - GSM/GPRS modules and terminals. [http://www.telit.com/en/products/gsm-gprs.php?p\\_ac=show&p=3](http://www.telit.com/en/products/gsm-gprs.php?p_ac=show&p=3), accessed Feb 22th 2014.
- [288] Telit Wireless Solutions. Easy Scan user guide, April 2013. <http://www.telit.com/module/infopool/download.php?id=6004>, accessed 2013-07-19.
- [289] Karl Thomas. Nine bad botnets and the damage they did, 2015. accessed 2017-06-08. URL: <https://www.welivesecurity.com/2015/02/25/nine-bad-botnets-damage/>.
- [290] Edric Thompson. Army examines feasibility of integrating 4G LTE with tactical network, September 25, 2015. accessed 2017-05-22. URL: <http://www.army.mil/article/87875/event16>.
- [291] Threema GmbH. Threema. <https://threema.ch/>, accessed July 17th 2014.
- [292] Craig Timberg. For Sale: Systems that can Secretly Track where Cellphone Users go Around the Globe. *The Washington Post*, August 2014. accessed 22-May-2017. URL: [https://www.washingtonpost.com/business/technology/for-sale-systems-that-can-secretly-track-where-cellphone-users-go-around-the-globe/2014/08/24/f0700e8a-f003-11e3-bf76-447a5df6411f\\_story.html](https://www.washingtonpost.com/business/technology/for-sale-systems-that-can-secretly-track-where-cellphone-users-go-around-the-globe/2014/08/24/f0700e8a-f003-11e3-bf76-447a5df6411f_story.html).

- [293] Patrick Traynor, William Enck, Patrick McDaniel, and Thomas La Porta. Mitigating attacks on open functionality in SMS-capable cellular networks. *IEEE/ACM Transactions on Networking (TON)*, 17(1):40–53, 2009.
- [294] Patrick Traynor, Michael Lin, Machigar Ongtang, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta. On Cellular Botnets: Measuring the Impact of Malicious Devices on a Cellular Network Core. In *ACM Conference on Computer and Communications Security (CCS)*, pages 223–234. ACM, 2009.
- [295] Guan-Hua Tu, Chi-Yu Li, Chunyi Peng, Yuanjie Li, and Songwu Lu. New Security Threats Caused by IMS-based SMS Service in 4G LTE Networks. In *ACM Conference on Computer and Communications Security (CCS)*, pages 1118–1130. ACM, 2016.
- [296] Huahong Tu, Ziming Zhao, and Gail-Joon Ahn. SoK: Everyone Hates Robocalls: A Survey of Techniques against Telephone Spam. In *IEEE Symposium on Security and Privacy (SP)*, pages 320–338. IEEE, 2016.
- [297] Katherine Tweed. Smart Thermostats Begin to Dominate the Market in 2015, 2015. accessed 2017-06-07. URL: <https://www.greentechmedia.com/articles/read/smart-thermostats-start-to-dominate-the-market-in-2015>.
- [298] Ubercoders. UberScanner. Google Play Store. <https://play.google.com/store/apps/details?id=org.ubercoders.uberscanner>, accessed July 17th 2014.
- [299] Andreas Ulbig, Theodor S. Borsche, and Göran Andersson. Impact of Low Rotational Inertia on Power System Stability and Operation. *arXiv*, 1312.6435, 2014. <https://arxiv.org/abs/1312.6435>.
- [300] Johanna Ullrich, Katharina Krombholz, Heidelinde Hobel, Adrian Dabrowski, and Edgar Weippl. IPv6 Security: Attacks and Countermeasures in a Nutshell. In *USENIX Workshop on Offensive Technologies (WOOT)*. USENIX Association, 2014.
- [301] Johanna Ullrich, Nicholas Stifter, Aljosha Judmayer, Adrian Dabrowski, and Edgar Weippl. Proof-of-Blackouts? How Proof-of-Work Cryptocurrencies Could Affect Power Grids. 2018. unpublished, to appear September 2018.
- [302] Nik Unger, Sergej Dechand, Joseph Bonneau, Sascha Fahl, Henning Perl, Ian Goldberg, and Matthew Smith. SoK: Secure Messaging. In *IEEE Symposium on Security and Privacy (SP)*, pages 232–249. IEEE, 2015.
- [303] Union for the Coordination of the Transmission of Electricity (UCTE). *Continental Europe Operation Handbook*, chapter Appendix 1 - Load-Frequency Control and



- Performance. European Network of Transmission System Operators for Electricity, 2004. URL: [https://www.entsoe.eu/fileadmin/user\\_upload/\\_library/publications/entsoe/Operation\\_Handbook/Policy\\_1\\_Appendix%20\\_final.pdf](https://www.entsoe.eu/fileadmin/user_upload/_library/publications/entsoe/Operation_Handbook/Policy_1_Appendix%20_final.pdf).
- [304] Union for the Coordination of the Transmission of Electricity (UCTE). *Continental Europe Operation Handbook*, chapter Policy 1 - Load-Frequency Control and Performance. European Network of Transmission System Operators for Electricity, 2004. URL: [https://www.entsoe.eu/fileadmin/user\\_upload/\\_library/publications/entsoe/Operation\\_Handbook/Policy1\\_final.pdf](https://www.entsoe.eu/fileadmin/user_upload/_library/publications/entsoe/Operation_Handbook/Policy1_final.pdf).
- [305] Union for the Coordination of the Transmission of Electricity (UCTE). *Continental Europe Operation Handbook*, chapter Introduction. European Network of Transmission System Operators for Electricity, 2004. URL: [https://www.entsoe.eu/fileadmin/user\\_upload/\\_library/publications/entsoe/Operation\\_Handbook/introduction\\_v25.pdf](https://www.entsoe.eu/fileadmin/user_upload/_library/publications/entsoe/Operation_Handbook/introduction_v25.pdf).
- [306] Union for the Coordination of Transmission of Electricity (UCTE). Final Report: System Disturbance on 4 November 2006, 2007. URL: [https://www.entsoe.eu/fileadmin/user\\_upload/\\_library/publications/ce/otherreports/Final-Report-20070130.pdf](https://www.entsoe.eu/fileadmin/user_upload/_library/publications/ce/otherreports/Final-Report-20070130.pdf).
- [307] U.S.-Canada Power System Outage Task Force. Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations, 2004. URL: <https://www.energy.gov/sites/prod/files/oeprod/DocumentsandMedia/BlackoutFinal-Web.pdf>.
- [308] Narseo Vallina-Rodriguez, Andrius Aučinas, Mario Almeida, Yan Grunenberger, Konstantina Papagiannaki, and Jon Crowcroft. RILAnalyzer: a Comprehensive 3G Monitor On Your Phone. In *Proceedings of the 2013 Internet Measurement Conference*, IMC '13, pages 257–264. ACM, October 2013. doi: 10.1145/2504730.2504764.
- [309] Valve Corporation. Steam Hardware & Software Survey, 2017. accessed 2017-06-07. URL: <http://store.steampowered.com/hwsurvey/>.
- [310] Fabian van den Broek, Brinio Hond, and Arturo Cedillo Torres. Security Testing of GSM Implementations. *Engineering Secure Software and Systems (ESSoS)*, 8364:179–195, 2014.
- [311] Fabian van den Broek, Roel Verdult, and Joeri de Ruiter. Defeating IMSI Catchers. In *ACM Conference on Computer and Communications Security (CCS)*, pages 340–351. ACM, 2015.
- [312] Thanh van Do, Hai Thanh Nguyen, and Nikolov Momchil. Detecting IMSI-Catcher Using Soft Computing. In *Springer International Conference on Soft Computing in Data Science (SCDS)*, pages 129–140. Springer, 2015.

- [313] Verband der Netzbetreiber (VDN). TransmissionCode 2007 - Netz- und Systemregeln der deutschen Übertragungsnetzbetreiber, 2007. In German. URL: [https://www.bdew.de/internet.nsf/id/A2A0475F2FAE8F44C12578300047C92F/\\$file/TransmissionCode2007.pdf](https://www.bdew.de/internet.nsf/id/A2A0475F2FAE8F44C12578300047C92F/$file/TransmissionCode2007.pdf).
- [314] Verint. Skylock Product Description 2013. *The Washington Post*, 2013. accessed 2017-02-10. URL: <http://apps.washingtonpost.com/g/page/business/skylock-product-description-2013/1276/>.
- [315] vinatan Hassidim, Yossi Matias, Moti Yung, and Alon Ziv. Ephemeral Identifiers: Mitigating Tracking & Spoofing Threats to BLE Beacons, 2016. <https://developers.google.com/beacons/eddystone-eid-preprint.pdf>.
- [316] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages 511– 518 vol.1, 2001. DOI 10.1109/CVPR.2001.990517. doi:10.1109/CVPR.2001.990517.
- [317] W3C Consortium. The Platform for Privacy Preferences 1.1 (P3P1.1) Specification. URL: <https://www.w3.org/TR/P3P11/>.
- [318] W3C Consortium. Tracking Protection Working Group. <http://www.w3.org/2011/tracking-protection/>, accessed 2013-05-02.
- [319] Susann Wagenknecht and Matthias Korn. Hacking As Transgressive Infrastructuring: Mobile Phone Networks and the German Chaos Computer Club. In *ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW)*, pages 1104–1117. ACM, 2016.
- [320] Michael Walker and Tim Wright. *Security, in GSM and UMTS: The Creation of Global Mobile Communication*. John Wiley & Sons, Ltd, Chichester, UK, 2001. doi:10.1002/0470845546.
- [321] Zhaoguang Wang, Zhiyun Qian, Qiang Xu, Zhuoqing Mao, and Ming Zhang. An Untold Story of Middleboxes in Cellular Networks. In *ACM SIGCOMM Computer Communication Review*, pages 374–385. ACM, 2011.
- [322] Dennis Wehrle. Open source IMSI-Catcher. Master’s thesis, Albert-Ludwig-Universität Freiburg, 2009.
- [323] Ralf-Philipp Weinmann. Baseband Attacks: Remote Exploitation of Memory Corruptions in Cellular Protocol Stacks. In *USENIX Workshop on Offensive Technologies (WOOT)*. USENIX Association, 2012.
- [324] Harald Welte. OpenBSC - Running your own GSM network, 08 2009. Talk at Hacking at Random 2009. Slides: <https://openbsc.osmocom.org/trac/raw-attachment/wiki/FieldTests/HAR2009/har2009-gsm-report.pdf>.

- [325] Harald Welte and Dieter Spaar. Running your own GSM network, 12 2008. Talk at 25th Chaos Communication Congress, Berlin, Germany. <https://events.ccc.de/congress/2008/Fahrplan/events/3007.en.html>, Slides: [http://events.ccc.de/congress/2008/Fahrplan/attachments/1259\\_25C3-OpenBSC.pdf](http://events.ccc.de/congress/2008/Fahrplan/attachments/1259_25C3-OpenBSC.pdf), Video: [https://media.ccc.de/v/25c3-3007-en-running\\_your\\_own\\_gsm\\_network](https://media.ccc.de/v/25c3-3007-en-running_your_own_gsm_network).
- [326] Wiggle Project. Wiggle: Wireless Network Mapping, 2017. accessed 2017-05-30. URL: <https://wiggle.net/>.
- [327] Ben Wojtowicz. OpenLTE - An open source 3GPP LTE implementation. <http://openlte.sourceforge.net/>, 2015. accessed 2017-04-05.
- [328] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A Practical Attack to De-anonymize Social Network Users. In *Security and Privacy (SP), 2010 IEEE Symposium on*, pages 223–238, May 2010. doi:10.1109/SP.2010.21.
- [329] Jiankai Xiao, Xiaoting Wang, Qianghua Guo, Hang Long, and Song Jin. Analysis and Evaluation of Jammer Interference in LTE. In *ACM International Conference on Innovative Computing and Cloud Computing (ICCC)*, pages 46–50. ACM, 2013.
- [330] Binjie Xin, Jinlian Hu, George Baciu, and Xiaobo Yu. Development of Weave Code Technology for Textile Products. *Fibres & Textiles in Eastern Europe*, 85:33–35, 2011.
- [331] Zhang Xu, Haining Wang, Zichen Xu, and Xiaorui Wang. Power Attack: An Increasing Threat to Data Centers. In *Network and Distributed System Security Symposium 2014, Proceedings of*. Internet Society, 2014.
- [332] Takayuki Yamada, Seiichi Gohshi, and Isao Echizen. Use of invisible noise signals to prevent privacy invasion through face recognition from camera images. In *Proceedings of the 20th ACM international conference on Multimedia*, MM '12, pages 1315–1316, New York, NY, USA, 2012. ACM. URL: <http://doi.acm.org/10.1145/2393347.2396460>, doi:10.1145/2393347.2396460.
- [333] Takayuku Yamada, Seiichi Gohshi, and Isao Echizen. Privacy Visor: Method for Preventing Face Image Detection by Using Differences in Human and Device Sensitivity. 2013. unpublished, under review for CMS 2013.
- [334] J. Yan, Y. Tang, Bo Tang, H. He, and Y. Sun. Power grid resilience against false data injection attacks. In *2016 IEEE Power and Energy Society General Meeting (PESGM)*, pages 1–5, 2016.
- [335] Y. Yan, Y. Qian, H. Sharif, and D. Tipper. A Survey on Cyber Security for Smart Grid Communications. *IEEE Communications Surveys Tutorials*, 14(4):998–1010, 2012.

- [336] Chen Yang. Weather the signaling storm. Technical Report 61, Huawei, 2011. URL: <http://www1.huawei.com/en/static/HW-094153.pdf>.
- [337] Q. Yang, J. Yang, W. Yu, D. An, N. Zhang, and W. Zhao. On False Data-Injection Attacks against Power System State Estimation: Modeling and Countermeasures. *IEEE Transactions on Parallel and Distributed Systems*, 25(3):717–729, 2014.
- [338] Da Yu and Wushao Wen. Non-access-stratum request attack in E-UTRAN. In *IEEE Computing, Communications and Applications Conference (ComComAp)*, pages 48–53. IEEE, 2012.
- [339] Roberto Yus, Primal Pappachan, Prajit Kumar Das, Eduardo Mena, Anupam Joshi, and Tim Finin. Demo: Faceblock: privacy-aware pictures for google glass. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, pages 366–366. ACM, 2014.
- [340] Wanqiao Zhang and Haoqi Shan. LTE Redirection: Forcing Targeted LTE Cellphone into Unsafe Network. In *Defcon*, 2016. URL: <https://media.defcon.org/DEFCON24/DEFCON24presentations/DEFCON-24-Zhang-Shan-Forcing-Targeted-Lte-Cellphone-Into-Unsafe-Network.pdf>.
- [341] Yuanyuan Zhou, Yu Yu, François Xavier Standaert, and Jean Jacques Quisquater. On the Need of Physical Security for Small Embedded Devices: a Case Study with COMP128-1 Implementations in SIM Cards. In *International Conference on Financial Cryptography and Data Security (FC)*, pages 230–238. Springer, 2013.
- [342] Peng Zhu, Fei Jia, and Junliang Zhang. A copyright protection watermarking algorithm for remote sensing image based on binary image watermark. *Optik - International Journal for Light and Electron Optics*, (0):–, 2013. doi:10.1016/j.ijleo.2012.12.049.
- [343] G. Zorn. RADIUS Attributes for IEEE 802.16 Privacy Key Management Version 1 (PKMv1) Protocol Support. RFC 5904 (Informational), June 2010. URL: <https://www.rfc-editor.org/rfc/rfc5904.txt>, doi:10.17487/RFC5904.
- [344] ZXing Team. Barcode Scanner. Google Play Store. <https://play.google.com/store/apps/details?id=com.google.zxing.client.android>, accessed July 17th 2014.