A5/1 is in the Air: Passive Detection of 2G (GSM) Ciphering Algorithms

Matthias Koch*, Christian Nettersheim*‡, Thorsten Horstmann*‡, Michael Rademacher*‡
*Fraunhofer FKIE, Bonn, Germany, firstname.lastname@fkie.fraunhofer.de
‡Hochschule Bonn Rhein-Sieg, Sankt Augustin, Germany, firstname.lastname@h-brs.de

***Abstract*—This paper investigates the ongoing use of the A5/1 ciphering algorithm within 2G GSM networks. Despite its known vulnerabilities and the gradual phasing out of GSM technology by some operators, GSM security remains relevant due to potential downgrade attacks from 4G/5G networks and its use in IoT applications. We present a comprehensive overview of a historical weakness associated with the A5 family of cryptographic algorithms. Building on this, our main contribution is the design of a measurement approach using low-cost, off-the-shelf hardware to passively monitor Cipher Mode Command messages transmitted by base transceiver stations (BTS). We collected over 500,000 samples at 10 different locations, focusing on the three largest mobile network operators in Germany. Our findings reveal significant variations in algorithm usage among these providers. One operator favors A5/3, while another surprisingly retains a high reliance on the compromised A5/1. The third provider shows a marked preference for A5/3 and A5/4, indicating a shift towards more secure ciphering algorithms in GSM networks.**

***Index Terms*—GSM, 2G, Cryptography, Security, A5/1, Measurement**

I. Introduction

While some mobile operators have begun phasing out their Global System for Mobile Communications (GSM) networks, notably in the US [1], GSM remains widely relevant globally due to its extensive coverage, which supports older cellular phones, Internet of Things (IoT) devices like smart meters and Global Positioning System (GPS) trackers. It also serves as a fallback option in areas where newer network generations do not provide sufficient coverage. Moreover, although one might view 2G network security as outdated, it remains important to recognize that downgrade attacks from 4G/5G to 2G still pose a significant threat, making users vulnerable even when modern network generations are present [2].

Among the most prominent security issues is the vulnerability of the A5/1 algorithm. Initially designed as a proprietary ciphering algorithm, A5/1's stream cipher procedure was reverse engineered in the late 1990s [3]. A5/1 is initialized by loading its three Linear-Feedback Shift Registers (LFSRs) with a 64-bit session key (called K_c) and a frame number as Initialization Vector (IV). The resulting state is called the *initial state* from which possible session keys can be extracted by back-clocking. Afterwards, the registers are clocked with a majority rule enabled for 100 additional rounds (the output of which is discarded) and then the subsequent key stream is used for the encryption of the next burst. This led to a variety of research efforts attacking various mathematical weaknesses of its protocol.

While many of these early efforts were constrained by their dependency on large numbers of recorded messages or impractically long attack and pre-computation times, the first practical (near) real-time cracking attack of the session key was presented by Nohl et al. around 2010 [4]. This attack leveraged several earlier insights. For instance, well-known control messages, such as `Cipher Mode Command (CMC)` and `System Information` messages, can be exploited as known-plaintexts to recover the cipher stream sequence, largely due to the presence of numerous constant padding bits. Combining distinguished points to reduce table lookup bottlenecks, substituting traditional Time-Memory Tradeoff (TMTO) lookup tables with rainbow tables to minimize collisions as well as applying optimizing strategies for key space and table compression, the *A5 Cracking Project* [4], [5] was able to make precomputed rainbow tables publicly available at a size of around 2 TB. These allow an attacker to efficiently recover candidates for the aforementioned *initial state* from few known cipher streams yielding a session key with high probability. A detailed report, highlighting the development and the interplay of these various techniques, can be found in [5].

Open source cracking tools like `kraken` [6] combined with today's increased processing power and decreased costs for storage make these attack vectors available to the public - as can be seen by the existence of several YouTube tutorials that detail the whole setup and usage[1]. Possible mitigations exist in the form of padding randomization [4], [7] and a gradual move to the block-cipher-based A5/3 and A5/4 algorithms [4], but in the face of backward compatibility [8], implementation on both network and Mobile Station (MS) side remains an open question.

Nowadays, in addition to A5/1, A5/3 and A5/4 are in use for encrypting over-the-air messages between a MS and a Base Transceiver Station (BTS). This raises the main research question of this work: **How much A5/1 is actually still used in today's mobile networks in Germany?**

The paper is organized as follows: Section II presents related work regarding the main research question. Section III presents an overview of necessary GSM network mechanics. We designed our own measurement approach based on low-

[1]Due to legal considerations, we do not include a reference to these videos.

cost Commercial Off-the-Shelf (COTS) hardware which we introduce in Section IV. Results are in Section V, limitations in Section VI, and Section VII concludes the paper with future research directions.

II. Related Work

A. A5 Usage Surveys

Morgan [9] proposed two methods for determining the usage and support of different A5 algorithms by Estonian mobile operators: a passive method and an active method. The passive method aligns with our approach, utilizing a heuristic technique that involves counting CMC messages transmitted by the BTS. These messages are received by Software Defined Radio (SDR) devices, such as HackRF or RTL-SDR dongles, and processed using the `gr-gsm` [10] software. Their active method involves connecting a modified phone to the network. In one scenario, the phone is configured to support only one specific A5 algorithm, allowing for determination of whether a BTS is capable of that algorithm. In another scenario, the phone supports all available A5 algorithms, providing insight into which algorithm the BTS prefers. In the first scenario, a successful location update indicated support for the chosen A5 algorithm, while in the second, the algorithm was identified using an SDR. The measurements focused on A5/1, A5/2, and A5/3, excluding A5/4.

The results showed that only one of three Estonian providers used A5/3, with the others exclusively using A5/1. Notably, none used A5/2, adhering to the recommendations of [8] and [11]. Their passive method had limitations, including short measurement periods of 8-17 hours each and a small set of specific BTS per provider, which prevented the observation of long-term effects. In contrast, our work addressed these limitations by conducting long-term measurements across multiple locations, providing a more comprehensive view of the network's encryption activities.

B. SnoopSnitch

The SnoopSnitch app [12] is another notable project that collects statistics on A5 algorithm usage as part of a broader mobile security analysis. Primarily designed to detect threats like IMSI catchers and silent/binary SMS, it also gathers data on mobile networks that users connect to. This data is compiled into yearly automated reports, which are published on the GSMmap website [13]. As noted in the SnoopSnitch FAQ [14], the app has significant limitations. It requires a rooted Android device with a Qualcomm baseband chip to access raw radio messages via the `/dev/diag` interface. Compatibility is limited to Android versions 4.4 to 12, excluding other operating systems and certain custom ROMs or devices lacking the diagnostic interface kernel driver.

The 2023 GSMmap report for Germany [15] indicates A5/1 usage between 13% and 36% across providers, though it notes that these figures are averages from diverse user contributions and "may be influenced by factors like location, Subscriber Identity Module (SIM) type, and network load" [14]. Moreover, as discussed in [9], SnoopSnitch only detects A5 algorithms used by test devices, not other users on the same network. Our passive approach aims to address this limitation by collecting data on all devices within a cell, unlike methods that rely on individual user data. In addition, GSMmap reports on the algorithms A5/0, A5/1, and A5/3. Our approach cannot detect connections with A5/0, as the absence of encryption means that there are no CMCs that we passively eavesdrop on; however, we found many connections utilizing A5/4.

III. Background on GSM

To motivate our approach in Section IV, a brief overview of a typical connection setup in GSM is provided hereafter. The first access point of an MS into the GSM network via the radio interface is the BTS of the respective cell. Each BTS transmits information to the MS on the downlink frequency (network $\rightarrow$ MS) and listens to incoming data from the MS within its coverage area on the uplink frequencies (MS $\rightarrow$ network). The frequencies that a BTS can offer are selected from a subset of those available to the Mobile Network Operator (MNO), ensuring that they do not interfere with frequencies used by neighboring cells. The entire frequency range is typically licensed to various MNOs by national regulatory bodies, such as the *Bundesnetzagentur* in Germany, for use in commercial GSM operations. In our context, we focus on the extended GSM-900 frequency bands (E-GSM) used in Germany, which utilize the downlink frequency range of 925.0 MHz to 960.0 MHz. Before the MS can transmit and receive encrypted data, it must first establish a connection to the subscriber's network. The exchange between the MS and the network varies slightly depending on the MS's state prior to establishing a new connection, such as being idle, switched off, entering a new location area, or requesting an additional connection. The following steps are a condensed high-level overview [16]–[19]:

1) The BTS periodically broadcasts `System Information` messages over the Broadcast Common Control Channel (BCCH), providing details about its associated MNO, location area, and cell identity.
2) The MS measures the signal strength of the available frequencies from the GSM range, whereby the BCCHs are monitored.
3) The MS selects the BTS with the strongest signal matching the provider identity stored in its SIM card.
4) A signalling channel is established, allowing the MS to request a service, such as a voice call or SMS.
5) To inform the network of its current location, the MS initiates a location update via a `Location Update Request`. This request includes various identifiers for the MS and a `Mobile Station Classmark` message, which notifies the network of the supported A5 algorithms.
6) The MS and network perform identification and authentication procedures. During this process, a new session key K_c for ciphering is derived on both ends using a 128-bit random challenge `RAND`, chosen by the network, and the master key K_i stored on the SIM card and at the network's Authentication Centre (AuC), respectively.

7) The network selects a ciphering algorithm supported by both the MS and the network for the connection and sends a `Cipher Mode Command (CMC)` message, informing the MS of its choice, to initiate encrypted communication.
8) This command causes the MS to enable (de-)ciphering for all subsequent messages of the connection and responds with an encrypted `Cipher Mode Complete`.
9) If the network deciphers the `Cipher Mode Complete` message, the location update completes and service requests proceed over an encrypted connection.

IV. Methodology

As seen in Section III, one approach to gathering data on the types of encryption algorithms used by a provider is to listen passively to the `Cipher Mode Commands (CMCs)` sent by a BTS on the downlink.

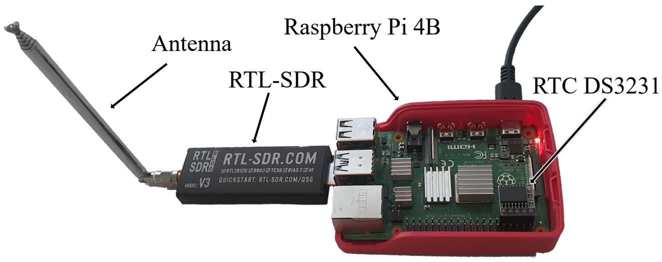

Fig. 1: Components of our sensor

A. Sensor

To capture and log these CMCs over long periods of time, we utilize an SDR to listen to downlink traffic of a BTS and simultaneously filter out instances of CMCs from the incoming packets. For each captured CMC, we extract the cipher algorithm used and log the timestamp of transmission. Our sensors, as depicted in Figure 1, consist of an RTL-SDR dongle connected to either a monopole or dipole antenna and a Raspberry Pi 4 Model B. The data collection is controlled by the `gsm-monitor.service` [20], a custom `systemd` service. This service does the following:

1) To start out, it searches for all available GSM frequencies received by the dongle. This uses the `kalibrate_rtl` [21] tool to find a list of frequencies and their respective signal strength.
2) The `gsm-monitor.service` is configured with a provider's Mobile Network Code (MNC). The frequencies found in the first step are then sorted by signal strength and the according channel frequencies are probed for a few seconds to find the MNC from the `System Information Message Type 3 (SI3)`. To receive GSM packets with the SDR, we use `gr-gsm` [10]. When a frequency for the desired provider is found, the Location Area Code (LAC) and the Cell ID (CID) are additionally recorded. This effectively finds the strongest currently available frequency for the chosen provider.
3) The data points each consist of a timestamp and an algorithm identifier, see [22]. Filtering and extracting is done with `tshark` [23].
4) To ensure continuous sufficient signal strength for data capture, a watchdog runs every 5 minutes, counting SI3 messages over 30 seconds, and restarts the service if the count falls below a set threshold.

For further technical details and setup instructions, refer to the project's git repository [2].

B. Deployment

We evaluated two distinct deployment options: one offline and the other online. For offline deployment, a real-time clock such as a *DS3231* needs to be installed on the Raspberry Pi. For online deployments, time synchronization can be achieved using the Network Time Protocol (NTP).

Online deployments offer greater flexibility, but necessitate an active internet connection. This connection can be preconfigured using the NetworkManager service before deployment. To enhance accessibility, we opted for remote management of the monitoring stations via Tailscale, which provides `ssh` access from virtually any network [24]. This setup allows us to manually administer changes to the chosen provider and retrieve collected data at any time.

C. Measuring campaign

	Provider A		Provider B		Provider C	
Location	**Days**	**#CMCs**	**Days**	**#CMCs**	**Days**	**#CMCs**
1/u	2.67	418	0	0	3.02	6131
2/u	3.8	872	3.75	5669	3.77	8576
3/u	58.08	13852	42.95	212567	15.07	40438
4/s	35.76	34302	33.87	54477	18.29	54914
5/s	1.07	966	1.06	1945	1.06	2914
6/s	2.05	964	1.99	2238	3.45	9611
7/u	7.15	1756	7.46	11780	1.99	7454
8/r	4.75	1778	2.87	3999	2.87	50247
9/s	1.42	946	2.23	5186	2.27	9742
10/r	2.13	1189	3.72	20182	0	0

TABLE I: Overview of the measurement campaign. Duration of the measurements and the number of captured CMCs per provider. Settlement type of the location given by u = urban, s = suburban, r = rural.

In total, we constructed five sensors as outlined in Section IV-A. A total of 565,115 CMCs packets were analyzed at 10 distinct locations. Measurements were conducted over a period of 88 days, commencing on 23 December 2024 and concluding on 19 March 2025. Furthermore, the measurement duration differs at each location, as seen in Table I. The locations at which we measured were distributed in and around Bonn, Germany. There were 10 locations in a total of 6 distinct municipalities. An assessment of whether the locations are urban, suburban or rural is also given in Table I. Altogether, we measured in four urban, four suburban and two rural locations. The results later showed that location did not have a major influence on the outcomes.

[2]https://github.com/mclab-hbrs/GSM-Cipher-Sensor

At least one day was measured per location and per provider, but usually more. As we were limited by the number of sensors, in most locations the provider to be measured was replaced after a period of time, so that the providers were not measured at the same time, but one after the other.

Long-term measurements were carried out at locations 3 and 4, which lasted approx. 73 days and approx. 54 days respectively. This was intended to generate a good initial pool of measured values. No measurements were taken for Provider B at location 1 and Provider C at location 10. At location 1, the signal for Provider B experienced excessive attenuation, presumably due to the indoor measurement environment, resulting in an insufficient signal strength for a valid measurement to be obtained. At Location 10, Provider C could also not be measured due to insufficient signal strength.

V. Evaluation & Results

In this section, we will present and discuss the results. Our results are examined across the three largest providers in Germany which are pseudonymized as Provider A, Provider B and Provider C.

Because we took measurements at the locations for different durations, the absolute values were not included in the overall result. Otherwise, the results would be skewed by the long-term measurements from locations 3 and 4. As we cannot rule out the possibility of site-specific anomalies at the locations, we weighted the measurement. For example, we measured all (10) locations for Provider A, but only 9 locations for provider B and C. Therefore, for the overall results, Provider A has weighting of $\frac{1}{10}$ and Provider B and C of $\frac{1}{9}$ respectively.

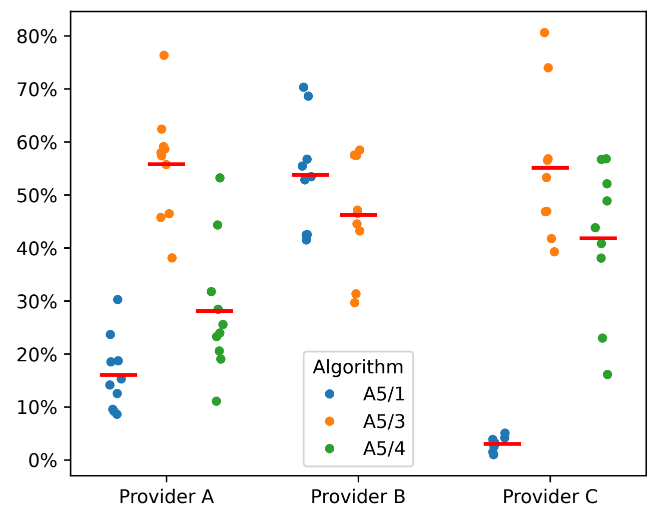

Fig. 2: Distribution of algorithm usage for the different providers across the various locations. Mean values are marked with red line.

In the analysis presented in Figure 2, we examine the distribution of algorithm usage (A5/1, A5/3, A5/4) across the providers. Each point represents the usage percentage of an algorithm at a specific location, while the red horizontal lines indicate the mean usage for each algorithm per provider.

- The result for **Provider A** shows a strong use of A5/3 with an average of 55.8%, whereby A5/4 is at around 28.1%. Algorithm A5/1 is still frequently used with approx. 16.1% on average.
- With **Provider B** it is even 53.8% for algorithm A5/1, whereas A5/3 is at 46.2%. We were unable to detect any communication for algorithm A5/4 at Provider B. We cannot say whether our observations occurred by chance, but it is statistically reasonable to assume that Provider B does not support A5/4 and even favors A5/1.
- **Provider C** has the lowest proportion of A5/1 traffic at around 3%. The remaining share is distributed between A5/3 with 55.1% and A5/4 with 41.8%.

As discussed in Section I, algorithm A5/1 is deemed broken. That is reflected in its low utilization rate at both Provider C and Provider A. In contrast, the usage at Provider B is surprisingly the highest, which raises questions about its operational context and the reasons behind this choice, despite the known issues associated with this algorithm. The complete absence of A5/4 at Provider B also raises concerns about the security standards of the network.

One theory is that since MS and BTS negotiate the algorithm and older MS tend to choose the older algorithm. Newer equipment would also prefer newer mobile technologies such as 4G or 5G. Another possibility is that the BTS does not implement, for example, the A5/4 algorithm. With our passive approach, we are not able to further test this.

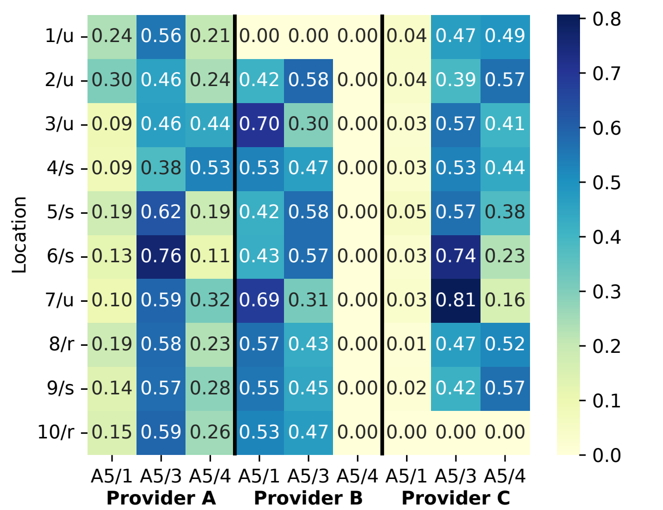

Fig. 3: Heatmap showing the usage rates of the algorithms per provider and per location. The color intensity represents the rate of algorithm usage. Settlement type of the location given by u = urban, s = suburban, r = rural.

Figure 3 shows a heatmap indicating the proportions of the algorithms for each location and provider. Darker fields indicate a more frequent occurrence than lighter fields. As already mentioned in Section IV-C we would like to point out the missing measurements from Provider B in Location 1 and Provider C in Location 10.

As the stripplot in Figure 2 has already shown, it is especially the high utilization of A5/1 and the lack of A5/4 at Provider B that stands out. A particularly large amount of A5/1 was measured during the long-term measurement in location 3. This may be due to a few MSs negotiating this algorithm. With our method, it is not possible to make more detailed statements about this. Usage varies significantly between locations, but the core findings hold. The consistently low usage of A5/1 at Provider C is also noteworthy.

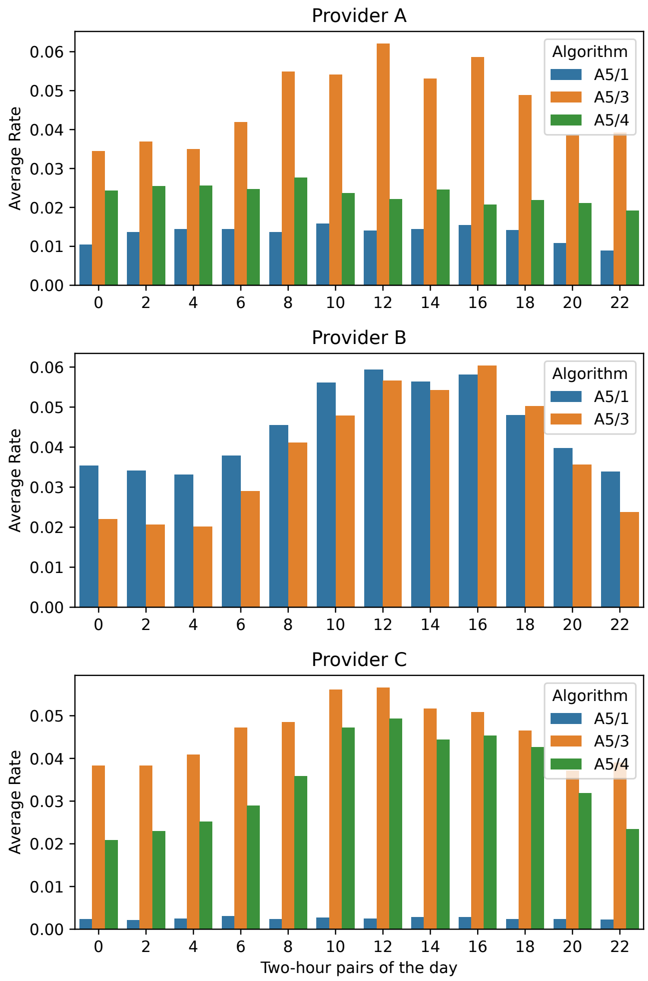

Fig. 4: Normalized hourly usage rates of different encryption algorithms for each provider. The rates are calculated as the proportion of the total measurements for each location and hour. The graph shows the average rate across all locations for each hour of the day.

Figure 4 shows the algorithm usage by time of day. In this graph, each location contributes 1/9 or respectively 1/10 to the graph. Two-hour pairs are listed on the x-axis, i.e. 0 represents hours 0-1, 2 represents hours 2-3, etc. The values of the graph, i.e. the y-axis, were formed as follows: First, for each location we calculated how much of the total traffic each algorithm has at each hour. Then the average was taken for each algorithm at each hour across the locations. This means that for each diagram all the bars add up to 1. By doing so we preserve the distribution of usage throughout the day.

- At **Provider B**, the use of the two algorithms is similar to each other over the course of the day. The graph corresponds roughly to what you would expect if you assume that the traffic is largely produced by humans or machines operated by humans. Comparatively little traffic in the nighttime hours, slowly increasing at the peak traffic times of midday and afternoon and then decreasing again towards the evening and at night.
- The graph for **Provider C** follows a similar pattern to that of Provider B for algorithms A5/3 and A5/4. For algorithm A5/1, the traffic is at a relatively constant low level. This could be a sign that the traffic does not come from humans, but from machines such as sensors or actuators. These machines may be older, meaning that the newer algorithms could be not supported.
- By contrast, only algorithm A5/3 follows the pattern for **Provider A**. Algorithms A5/1 and A5/4 behave rather consistently over the course of the day, with some upward and downward fluctuations. While the behavior of algorithm A5/1 at provider C could still be explained by machine traffic, this explanation is more difficult here.

Our evaluation of algorithm usage across the three major providers in Germany reveals significant variations in the adoption of encryption algorithms. Provider A demonstrates a notable preference for A5/3, while Provider B's unexpectedly high usage of the broken A5/1 raises concerns about its security practices. Provider C, on the other hand, displays a predominantly low utilization of A5/1, indicating a shift towards more secure alternatives like A5/3 and A5/4.

VI. Known Limitations

By design, our approach is limited to monitoring packets transmitted over the air when communicating with unknown parties. We rely on the traffic generated between MS and BTS, as our passive monitoring method can only capture what is already in the air. However, by conducting long-term measurements across multiple locations, we can provide strong indicators of the cipher algorithm usage patterns within the monitored region.

Our current sensor setup has limitations due to the use of a single-frequency SDR, which prevents simultaneous monitoring of multiple providers with a single sensor. Additionally, GSM-1800 networks could be investigated with the present method, but as the deployed RTL-SDR dongles only receive frequencies up to 1.75 GHz and as German MNOs gradually move their 2G operations into the GSM-900 band to free up frequencies for LTE usage, we decided to limit both the scope of the survey and hardware cost. These points could be addressed by either deploying multiple sensors or using more advanced SDRs capable of handling multiple frequencies and a higher frequency range in exchange for higher deployment costs. The setup also requires a nearby power supply and indoor placement, limiting deployment options. While battery-assisted, weatherproof designs could expand placement possibilities, this may not be necessary in areas sufficiently covered by 2G and densely populated like Germany.

Power consumption is another consideration. The Raspberry Pi's continuous processing of GSM packets creates a constant CPU load, which could quickly drain a battery. This makes it challenging to use in areas without easy access to power supplies and likely precludes the use of more power-efficient IoT devices, as they typically rely on idle times or low-power operation periods that are not adequate in this application.

The limited CPU and USB bus speed of the Raspberry Pi may occasionally lead to randomly dropped GSM packets. However, since CMCs are sparsely distributed, compared to the rest of the packets, this doesn't significantly affect the overall observed patterns in cipher algorithm distribution over extended periods. Computers with higher CPU and bus speeds could be used to avoid dropping, but in exchange for higher costs and less deployment flexibility. Factors such as receiver placement, signal strength, and SDR hardware quality may also influence data gathering. Despite these limitations, the method provides valuable insights into long-term trends in cipher algorithm usage.

VII. Conclusion & Open Questions

In conclusion, this study provides critical insights into the ongoing use of the A5/1 ciphering algorithm within 2G (GSM) networks, despite its well-documented vulnerabilities. Our findings reveal a significant variance in the adoption of cryptographic algorithms among major mobile network operators in Germany. Notably, Provider B's high reliance on the compromised A5/1 algorithm raises serious concerns regarding its security practices, especially when juxtaposed with Providers A and C, which demonstrate a transition towards more secure alternatives like A5/3 and A5/4. The persistence of A5/1 highlights the challenges posed by legacy systems and the potential risks of downgrade attacks as 4G/5G users switch between different generations of mobile networks. Our methodology, employing low-cost hardware for passive monitoring, effectively captures the algorithm usage patterns, underscoring the importance of continued vigilance and assessment of network security practices.

Future research should focus on investigating the underlying reasons for the discrepancies in algorithm adoption among operators, the implications for user security, and the potential for upgrading legacy systems. Additionally, there is a need to explore the feasibility of implementing more robust encryption standards across all operators to mitigate risks associated with outdated protocols.

Open questions remain regarding the long-term impacts of maintaining outdated encryption standards in a rapidly evolving technological landscape and how best to balance legacy support with enhanced security measures for users.

References

[1] M. Dano, "T-Mobile begins dismantling 2G network," https://www.lightreading.com/2g-3g-4g/t-mobile-begins-dismantling-2g-network, February 2025, [Online; accessed 2025-03-21].

[2] M. S. Wani, M. Rademacher, T. Horstmann, and M. Kretschmer, "Security Vulnerabilities in 5G Non-Stand-Alone Networks: A Systematic Analysis and Attack Taxonomy," *Journal of Cybersecurity and Privacy*, vol. 4, no. 1, pp. 23–40, 2024.

[3] M. Briceno, I. Goldberg, and D. Wagner, "A pedagogical implementation of the GSM A5/1 and A5/2 "voice privacy" encryption algorithms," May 1999, [Online; accessed 2025-03-24]. [Online]. Available: https://cryptome.org/jya/a51-pi.htm

[4] K. Nohl, "Attacking phone privacy," *Black Hat USA*, pp. 1–6, 2010. [Online]. Available: https://media.blackhat.com/bh-ad-10/Nohl/BlackHat-AD-2010-Nohl-Attacking-Phone-Privacy-wp.pdf

[5] J. Lu, Z. Li, and M. Henricksen, *Time–Memory Trade-Off Attack on the GSM A5/1 Stream Cipher Using Commodity GPGPU*. Springer Berlin Heidelberg, 2016, p. 350–369. [Online]. Available: https://doi.org/10.1007/978-3-319-28166-7_17

[6] F. A. Stevenson. kraken. [Accessed: 24-03-2025]. [Online]. Available: https://github.com/joswr1ght/kraken

[7] 3GPP, "Mobile Station - Base Station System (MS - BSS) interface; Data Link (DL) layer specification," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 44.006, 05 2024, version 18.0.0 Release 18. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/144000_144099/144006/18.00.00_60/ts_144006v180000p.pdf

[8] 3GPP, "Security related network functions," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 43.020, 05 2024, version 18.0.0 Release 18. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/143000_143099/143020/09.04.00_60/ts_143020v090400p.pdf

[9] D. Morgan, "Determining the A5 Encryption Algorithms Used in 2G (GSM) Networks," in *NordSec 2024*, vol. LNCS 15396, 2025, p. 45–61.

[10] P. Krysik and B. Kerler. gr-gsm. [Accessed: 25-03-2025]. [Online]. Available: https://github.com/bkerler/gr-gsm

[11] GSMA, "Security Algorithm Deployment Guidance," GSM Association, Official Document FS.35, 04 2022, Version 3.0. [Online]. Available: https://www.gsma.com/solutions-and-impact/technologies/security/wp-content/uploads/2022/09/FS.35-v3.0.pdf

[12] Security Research Labs. SnoopSnitch. [Accessed: 20-03-2025]. [Online]. Available: https://github.com/srlabs/snoopsnitch

[13] Security Research Labs. GSM Security Map. [Accessed: 20-03-2025]. [Online]. Available: https://gsmmap.org/

[14] Security Research Labs. FAQ - SnoopSnitch. [Accessed: 20-03-2025]. [Online]. Available: https://web.archive.org/web/20240918170514/https://opensource.srlabs.de/projects/snoopsnitch/wiki/FAQ

[15] GSM Map Project, "Mobile network security report: Germany," Security Research Labs, Berlin, Tech. Rep., 05 2023. [Online]. Available: https://gsmmap.org/assets/pdfs/gsmmap.org-country_report-Germany-2023-05.pdf

[16] M. Sauter, *From GSM to LTE-advanced Pro and 5G: An introduction to mobile networks and mobile broadband*. John Wiley & Sons, 2017.

[17] J. Eberspächer, H.-J. Vögel, C. Bettstetter, and C. Hartmann, *GSM – Architecture, Protocols and Services*, 3rd ed. John Wiley & Sons, 2008.

[18] G. Heine, *GSM Networks: Protocols, Terminology, and Implementation*. Artech House, Inc., 1998.

[19] 3GPP, "Mobile radio interface Layer 3 specification; Core network protocols; Stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 24.008, 09 2024, version 18.7.0 Release 18. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/124000_124099/124008/18.07.00_60/ts_124008v180700p.pdf

[20] Repo included in final Version of the paper.

[21] S. Markgraf. kalibrate-rtl. [Accessed: 25-03-2025]. [Online]. Available: https://github.com/steve-m/kalibrate-rtl

[22] 3GPP, "Mobile radio interface layer 3 specification; GSM/EDGE Radio Resource Control (RRC) protocol," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 44.018, 05 2024, version 18.0.0 Release 18. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/144000_144099/144018/18.00.00_60/ts_144018v180000p.pdf

[23] The Wireshark Team. tshark. [Accessed: 25-03-2025]. [Online]. Available: https://tshark.dev/

[24] Tailscale Inc. Tailscale. [Accessed: 25-03-2025]. [Online]. Available: https://tailscale.com/