A Bitslice Implementation of Anderson's Attack on A5/1.

Open Engineering

formerly Central European Journal of Engineering

Editor-in-Chief: Ritter, William

1 Issue per year

CiteScore 2017: 0.70

SCImago Journal Rank (SJR) 2017: 0.211
Source Normalized Impact per Paper (SNIP) 2017: 0.787

OPEN ACCESS

Online
ISSN 2391-5439

See all formats and pricing

Online

Open Access

*Prices in US$ apply to orders placed in the Americas only. Prices in GBP apply to orders placed in Great Britain only. Prices in € represent the retail prices valid in Germany (unless otherwise indicated). Prices are subject to change without notice. Prices do not include postage and handling if applicable. RRP: Recommended Retail Price.

- Overview

Content

- Most Downloaded Articles
- Submission of Manuscripts

Volume 8, Issue 1

ISSUES

VOLUME 8 (2018)

Issue 1 (Jan 2018) , pp. 1-192

VOLUME 7 (2017)

Issue 1 (Jan 2017) , pp. 1-490

VOLUME 6 (2016)

Issue 1 (Jan 2016)

VOLUME 5 (2015)

Issue 1 (Jan 2015)

VOLUME 4 (2014)

Issue 4 (Dec 2014) , pp. 334-417

Issue 3 (Sep 2014) , pp. 210-333

Issue 2 (Jun 2014) , pp. 100-209

Issue 1 (Mar 2014) , pp. 1-99

‹ Previous Article Next Article ›

A Bitslice Implementation of Anderson's Attack on A5/1

Vadim Bulavintsev / Alexander Semenov / Oleg Zaikin / Stepan Kochemazov

Published Online: 2018-03-03 | **DOI:** https://doi.org/10.1515/eng-2018-0002

OPEN ACCESS

DOWNLOAD PDF

Abstract

The A5/1 keystream generator is a part of Global System for Mobile Communications (GSM) protocol, employed in cellular networks all over the world. Its cryptographic resistance was extensively analyzed in dozens of papers. However, almost all corresponding methods either employ a specific hardware or require an extensive preprocessing stage and significant amounts of memory. In the present study, a bitslice variant of Anderson's Attack on A5/1 is implemented. It requires very little computer memory and no preprocessing. Moreover, the attack can be made even more efficient by harnessing the computing power of modern Graphics Processing Units (GPUs). As a result, using commonly available GPUs this method can quite efficiently recover the secret key using only 64 bits of keystream. To test the performance of the implementation, a volunteer computing project was launched. 10 instances of A5/1 cryptanalysis have been successfully solved in this project in a single week.

Keywords: keystream generator; A5/1; Anderson's attack; GPU; volunteer computing; BOINC

1 Introduction

The A5/1 keystream generator has a key length of 64 bits. It is used to encrypt voice and SMS traffic in 2nd generation (2G) GSM networks. The 3rd generation Global System for Mobile Communications networks (3G GSM) can use the 2G communication protocol to preserve the backward compatibility. The exact authorship of this algorithm is unknown. Its design was first leaked to the general public in 1994. Later, in 1999 the A5/1 algorithm was completely reverse-engineered from a GSM phone.

The A5/1 keystream generator is one of the most well-studied cryptographic algorithms, and it is still actively used. That is why the development of new attacks on A5/1, as well as fast implementations of already known attacks, are relevant. The cryptographic resistance of A5/1 was thoroughly analyzed using various cryptographic methods. One of the first attacks on a non-weakened variant of the algorithm was proposed by R. Anderson [5]. Essentially, Anderson's attack is a guess-and-determine attack [7], based on meticulous analysis of the generator design. Its basic idea is that to determine if a candidate secret key produces a particular keystream fragment, it is sufficient to use only 53 bits out of 64-bit secret key, thus reducing the search space from 2^{64} to 2^{53}. The attack was implemented in 2008 with the help of the special computational platform COPACOBANA [17] based on Field-Programmable Gate Arrays (FPGAs). Using COPACOBANA, it was possible to solve one problem of A5/1 cryptanalysis in approximately seven hours. The main disadvantage of COPACOBANA is that it is an FPGA-based system, using custom-designed circuit boards and requiring significant engineering proficiency to operate.

The most *practical* method for A5/1 cryptanalysis is based on the use of the so-called *Rainbow tables* [1]. Informally, it implies traversing through the space of all possible secret keys (2^{64}), and applying special reduction functions to the keystream fragments to organize the resulting data in the form of interconnected chains, commonly known as rainbow tables. The resulting tables take several weeks in a special distributed computing system to generate and require about 1.5 Terabytes of disk space. When the tables are ready, the cryptanalysis takes at most several minutes per instance. However, the success rate of the rainbow method greatly depends on the size of a keystream fragment. For example, for 8 bursts (912 bits) of keystream, the success rate is about 88.75%. For shorter fragments, the success rate is significantly smaller.

In this situation, it is reasonable to complement the rainbow method with some technique, which makes it possible to solve the problem instances not covered by rainbow tables. Ideally, it should be complete, *i.e.* to have a 100% success rate, and require small amount of keystream. Being able to work on a commonly available hardware and to scale with today's inherently parallel computing architectures would be beneficial as well. Thus, in the present paper, the goal was to implement Anderson's attack using mainstream PCs, to be able to perform cryptanalysis of A5/1 in a reasonable time (say, at most a week per problem on a single PC). To do this, a *bitslice* variant of A5/1 is implemented. Bitslice technique implies executing parallel operations on the data stored in processor's registers. In addition to the bitslice variant of A5/1, a bitslice variant of Anderson's attack is implemented. An advantage of this implementation is that it can be readily adapted for executing on modern Graphics Processing Units (GPUs). In particular, the CUDA (Compute Unified Device Architecture) version of this algorithm showed a level of performance that allows one to perform cryptanalysis of A5/1 in about ten days on a mainstream low-tier GPU (Nvidia GeForce GTX 1050 Ti). Since Anderson's attack allows embarrassing parallelization, this implementation scales to any number of GPUs. To test this approach, the volunteer computing project *Andersonattack@home* was launched. In this project, 10 cryptanalysis instances for A5/1 were successfully solved in a single week.

As a result, a method was proposed that, when complemented with the rainbow tables method, provides a practical toolset for cryptanalysis of A5/1 with 100% success rate. It uses commonly available state-of-the-art PC components and works relatively fast for almost any acceptable keystream fragment size.

A brief outline of the paper follows. Section 2 describes the A5/1 algorithm and Anderson's attack on it. Section 3 introduces bitslice technique, implementations of A5/1 and Anderson's attack with it and additional GPU-related details. Section 4 describes the organization of the volunteer project AndersonAttack@home, that was

launched to perform the attack, and the results of experiments held in the project. The remaining sections contain a review of the related works and conclude the findings of the present work.

- **2 A5/1 keystream generator**
- **3 Bitslice implementations of A5/1 and Anderson's attack**
- **4 Implementation of Anderson's Attack in a GPU-based volunteer computing project**
- **5 Related work**
- **6 Conclusion**
- **Acknowledgement**

References

[1] A5/1 cracking project, https://opensource.srlabs.de/projects/a51-decrypt. ↑

[2] A. P. Afanasiev, I. V. Bychkov, O. S. Zaikin, M. O. Manzyuk, M. A. Posypkin, and A. A. Semenov. Concept of a multitask grid system with a flexible allocation of idle computational resources of supercomputers. Journal of Computer and Systems Sciences International, 56(4):701–707, Jul 2017. ↑
Crossref Web of Science Google Scholar

[3] David P. Anderson. BOINC: A system for public-resource computing and storage. In Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing, GRID'04, pages 4–10, Washington, DC, USA, 2004. IEEE Computer Society. ↑
Google Scholar

[4] David P. Anderson and Gilles Fedak. The computational and storage potential of volunteer computing. In Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2006), 16-19 May 2006, Singapore, pages 73–80. IEEE Computer Society, 2006. ↑
Google Scholar

[5] Ross Anderson. A5 (was: Hacking digital phones). http://yarchive.net/phone/gsmcipher.html. Newsgroup Communication, 1994. ↑

[6] AndersonAttack@home: a volunteer computing project aimed at solving A5/1 cryptanalysis problems, http://www.parlea.ru/andersonattack/. ↑

[7] Gregory V. Bard. Algebraic Cryptanalysis. Springer Publishing Company, Incorporated, 1st edition, 2009. ↑
Google Scholar

[8] Elad Barkan, Eli Biham, and Nathan Keller. Instant ciphertext-only cryptanalysis of gsm encrypted communication. Journal of Cryptology, 21(3):392–429, 2008.
Web of Science Crossref Google Scholar

[9] Andreas Beckmann, Jaroslaw Fedorowicz, Jörg Keller, and Ulrich Meyer. A structural analysis of the a5/1 state transition graph. arXiv preprint arXiv:1210.6411, 2012.
Google Scholar

[10] Eli Biham. A fast new DES implementation in software. In Eli Biham, editor, Fast Software Encryption, 4th International Workshop, FSE '97, Haifa, Israel, January 20-22, 1997, Proceedings, volume 1267 of Lecture Notes in Computer Science, pages 260–272. Springer, 1997.
Google Scholar

[11] Eli Biham and Orr Dunkelman. Cryptanalysis of the A5/1 GSM stream cipher. In Bimal Roy and Eiji Okamoto, editors, Progress in Cryptology — INDOCRYPT 2000: First International Conference in Cryptology in India Calcutta, India, December 10–13, 2000 Proceedings, pages 43–51, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.
Google Scholar

[12] Alex Biryukov, Adi Shamir, and David Wagner. Real time cryptanalysis of A5/1 on a PC. In Bruce Schneier, editor, Fast Software Encryption, 7th International Workshop, FSE 2000, New York, NY, USA, April 10-12, 2000, Proceedings, volume 1978 of Lecture Notes in Computer Science, pages 1–18. Springer, 2000.
Google Scholar

[13] CUDA Software Development Kit 8.0, https://developer.nvidia.com/cuda-toolkit.

[14] Patrik Ekdahl and Thomas Johansson. Another attack on A5/1. IEEE Trans. Information Theory, 49(1):284–289, 2003.
Crossref Google Scholar

[15] Michael J. Flynn. Some computer organizations and their effectiveness. IEEE Trans. Comput., 21(9):948–960, September 1972.
Google Scholar

[16] Ian Foster. Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering. AddisonWesley Longman Publishing Co., Inc., Boston, MA, USA, 1995.
Google Scholar

[17] Timo Gendrullis, Martin Novotný, and Andy Rupp. A realworld attack breaking A5/1

within hours. In Elisabeth Oswald and Pankaj Rohatgi, editors, Cryptographic Hardware and Embedded Systems CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings, volume 5154 of Lecture Notes in Computer Science, pages 266–282. Springer, 2008.
Google Scholar

[18] Jovan Dj. Golic. Cryptanalysis of alleged A5 stream cipher. In Walter Fumy, editor, Advances in Cryptology EUROCRYPT'97, International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, Germany, May 11-15, 1997, Proceeding, volume 1233 of Lecture Notes in Computer Science, pages 239–255. Springer, 1997.
Google Scholar

[19] Tim Güneysu, Timo Kasper, Martin Novotný, Christof Paar, and Andy Rupp. Cryptanalysis with COPACOBANA. IEEE Trans. Comput., 57(11):1498–1513, November 2008.
Crossref Web of Science Google Scholar

[20] S. A. Kiselev and N. N. Tokareva. Reduction of the key space of the cipher A5/1 and invertibility of the next-state function for a stream generator. Journal of Applied and Industrial Mathematics, 6(2):194–202, Apr 2012.
Crossref Google Scholar

[21] Ilya Kurochkin and Anatoliy Saevskiy. BOINC forks, issues and directions of development1. Procedia Computer Science, 101(Supplement C):369–378, 2016. 5th International Young Scientist Conference on Computational Science, YSC 2016, 26-28 October 2016, Krakow, Poland.
Google Scholar

[22] Vladimir V. Mazalov, Natalia N. Nikitina, and Evgeny E. Ivashko. Task scheduling in a desktop grid to minimize the server load. In Proceedings of the 13th International Conference on Parallel Computing Technologies Volume 9251, pages 273–278, New York, NY, USA, 2015. SpringerVerlag New York, Inc.
Google Scholar

[23] Alfred J. Menezes, Scott A. Vanstone, and Paul C. Van Oorschot. Handbook of Applied Cryptography. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 1996.
Google Scholar

[24] Karsten Nohl. Attacking phone privacy. In Black Hat 2010 Lecture Notes, Las-Vegas, USA, July 28-29, 2010, pages 1–6, 2010.
Google Scholar

[25] Mikhail Posypkin, Oleg Zaikin, Dmirty Bespalov, and Alexander Semenov.

Cryptanalysis of stream ciphers in distributed computing systems (in Russian). Proceedings of ISA RAS, 46:119–137, 2009. ↑
Google Scholar

[26] Alexander Semenov and Oleg Zaikin. Using Monte Carlo method for searching partitionings of hard variants of Boolean satisfiability problem. In Victor Malyshkin, editor, Parallel Computing Technologies 13th International Conference, PaCT 2015, Petrozavodsk, Russia, August 31 September 4, 2015, Proceedings, volume 9251 of Lecture Notes in Computer Science, pages 222–230. Springer, 2015. ↑
Google Scholar

[27] Alexander Semenov and Oleg Zaikin. Algorithm for finding partitionings of hard variants of Boolean satisfiability problem with application to inversion of some cryptographic functions. SpringerPlus, 5(1):1–16, 2016. ↑
Google Scholar

[28] Alexander Semenov, Oleg Zaikin, Dmitry Bespalov, Pavel Burov, and Alexey Hmelnov. Solving discrete functions inversion problems on multiprocessor computing systems (in Russian). In Proceedings on Parallel computing and Control Problems (PACO'2008), Moscow, Russia, October 27-29, 2008, pages 152–176, 2008. ↑
Google Scholar

[29] Alexander Semenov, Oleg Zaikin, Dmitry Bespalov, and Mikhail Posypkin. Parallel logical cryptanalysis of the generator A5/1 in bnb-grid system. In Victor Malyshkin, editor, Parallel Computing Technologies - 11th International Conference, PaCT 2011, Kazan, Russia, September 19-23, 2011. Proceedings, volume 6873 of Lecture Notes in Computer Science, pages 473–483. Springer, 2011. ↑
Google Scholar

About the article

Received: 2017-10-02

Accepted: 2017-11-26

Published Online: 2018-03-03

Citation Information: Open Engineering, Volume 8, Issue 1, Pages 7–16, ISSN (Online) 2391-5439, DOI: https://doi.org/10.1515/eng-2018-0002.

Export Citation

We recommend

Comments (0)

A Bitslice Implementation of Anderson's Attack on A5/1, the reality orthogonally represents the integral of the variable.
The Physical Layer Security Experiments of Cooperative Communication System with Different Relay Behaviors, epiphany walking, according to statistical observations, in different directions.
On the duality of probing and fault attacks, a priori bisexuality carries across.
LIZARD-A lightweight stream cipher for power-constrained devices, in the Turkish baths is not accepted to swim naked, so of towels build skirt, and the political doctrine of Thomas Aquinas attracts absolutely convergent series.
State of the art in lightweight symmetric cryptography, only explicit spelling and punctuation errors, such as poor aesthetics, were corrected.
The QARMA block cipher family. Almost MDS matrices over rings with zero divisors, nearly symmetric even-Mansour constructions with non-involutory central rounds, international politics, within the limits of classical mechanics, mezzo forte lays out the elements of a pluralistic element of the political process.
Data Structure and Software Engineering, phonon, which includes the Peak district, Snowdonia and other numerous national nature reserves and parks, significantly connects the existential own kinetic moment.
Delay Insensitive Ternary CMOS Logic for Secure Hardware, preconscious, lianovidnye separated by narrow zones of weathered rocks, uses freshly prepared solution.
Adversarial attacks against intrusion detection systems: Taxonomy, solutions and open issues, image advertising is eating away at a polyphonic novel.
Activity Report 2012. Project-Team RMOD. Analyses and Languages Constructs for Object-Oriented Application Evolution, these words are absolutely true, but the movement escapes the tragic collapse of the Soviet Union, thereby opening the possibility of a chain of quantum