Improving SHA-2 Collisions

Using Satisfiability

Modulo Theory (SMT) Solvers

BSc Computer Science  
Marcel Barlik  
[marcel.barlik@city.ac.uk](mailto:marcel.barlik@city.ac.uk)

# Table of Contents

[Table of Contents 1](#_Toc7)

[Abstract 2](#_Toc8)

[Introduction 2](#_Toc9)

[Research Questions 2](#_Toc10)

[Background Knowledge (8.2.4) 2](#_Toc11)

[Related Work 2](#_Toc12)

[Glossary 2](#_Toc13)

[Theoretical Framework 2](#_Toc14)

[Methodology (8.2.5) 3](#_Toc15)

[Cryptanalytic Techniques 3](#_Toc16)

[Experimental Setup 3](#_Toc17)

[Results & Analysis (8.2.6) 3](#_Toc18)

[Security Implications 3](#_Toc19)

[Limitations and Future Work 3](#_Toc20)

[Conclusion 4](#_Toc21)

[References 4](#_Toc22)

[Appendices 4](#_Toc23)

# Abstract

* Short findings from “Analysis” part

# Introduction

* Background & Motivation
* Problem Statement
* RQs
* Contributions
* Paper Structure

## Research Questions

**RQ1:** Does using a more effective SMT solver yield better SHA-256 collision results?

**RQ2**: Can encodings provided in the research (Li, Y. Liu, F. And Wang, G, 2024) be improved upon, aiming for better practical SHA-256 collisions?

# Background Knowledge (8.2.4)

## Related Work

* Previous cryptanalysis approaches
* Theoretical foundations
* References to base paper
* Gaps in existing literature

## Glossary

* Glossary of words and terms (8.2.8)

# Theoretical Framework

* (Math Equations for RQ2)
* Mathematical foundations
* Cryptographic primitives
* Attack models

# Methodology (8.2.5)

* Research design
* Tools and Technologies

## Benchmark Methodology

* Data collection
* Design
* Evaluation

Slide 4 (https://moodle4.city.ac.uk/pluginfile.php/1093089/mod\_resource/content/6/UGP\_Session-Methods\_2024-25-v10.pptx.pdf):

What are the content and organisation of the Method Chapter?

What you need to report and in how much detail

– How do you report methods for your software development and

assessment?

– How do you report methods for experiments, measurements etc.?

– How do you report role of supervisor / client / others?

– How do you report use of AI aids?

– How do you report on Legal, Social, Ethical, Professional Issues in this

chapter?

– What goes into the Method chapter, what elsewhere?

Definition of how you ran your project: both your overall plan and choice of steps to go though, and how it turned out in practice

A justification of why you chose to do it this way: explaining your decisions, both for your initial planning and in response to events during the project

– product development lifecycle followed

– technologies for building your product

– methods for assessing it

– methods for any experiments or measurements

– how the work to be done was subdivided into phases

– timeline of the phases of work, specific incidents causing plan changes

# Cryptanalytic Techniques

* Detailed info of the approach taken
* Algorithms and methods developed
* GitHub reference (impl details/overview)

# Experimental Setup

* Hardware/Software env
* Reproducibility considerations
* Param settings
* Characteristics

# Results & Analysis (8.2.6)

* RQ1 - Benchmark results (graphs)
* RQ1 Discussion
* Performance analysis
* Comparison with previous approaches

# Security Implications

* Vulnerabilities identified (if any)
* Impact assessment

# Limitations and Future Work

* Difficulties while developing research
* Personal limitations (for e.g. Limited knowledge)
* Potential improvements
* Future research directions
* Pain points during the project

# Conclusion

* Summary of findings
* Significance of results
* Final remarks

# References

# Appendices

* Detailed proofs
* Code snippets

(8.2.10) The following must be included:

* Appendix A: Project Definition Document;
* Appendix B: Reuse Summary (this must always be Appendix B; see Section 9.6, "Reusing software and how to document re-use");
* (mandatory for MSci students in Cybersecurity, good practice for all MSci students) Appendix C: Minutes of Supervisory Meetings (this must always be Appendix C);
* Complete records of each interview; questionnaires and complete questionnaire replies;
* Requirements;
* Routine design documentation;
* Source code, with instructions for building an executable version (as part of the “product package” submission; see next section);
* Test plans and test results; output listings; displays etc.;
* Any software installation guides and user guides produced (as part of the “product package” submission; see next section);
* An executable version of your software with installation instructions, or a URL if it is a web-based application that can be tested online (as part of the “product package” submission; see next section);
* All the reports produced for a client;
* Additional screen shots of your product running that are useful for understanding the product.

Other stuff to do:

* Reference PDD and what was delivered/wasn’t, what was improved on, etc
* must not exceed 12,000 words
* Same stuff as in 4.5.1 for cover sheet
* 8.2.2
* 8.2.3
* Read into what the video submission thing is about?