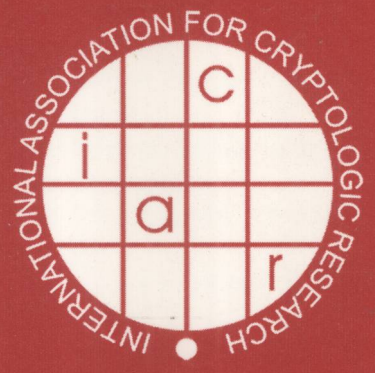
Victor Shoup (Ed.)

LNCS 3621

Advances in Cryptology – CRYPTO 2005

25th Annual International Cryptology Conference
Santa Barbara, California, USA, August 2005
Proceedings



Springer

Victor Shoup (Ed.)

Advances in Cryptology – CRYPTO 2005

25th Annual International Cryptology Conference
Santa Barbara, California, USA, August 14-18, 2005
Proceedings

Springer

Volume Editor

Victor Shoup
New York University, Department of Computer Science
251 Mercer Street, New York, NY 10012, USA
E-mail: shoup@cs.nyu.edu

Library of Congress Control Number: Applied for

CR Subject Classification (1998): E.3, G.2.1, F.2.1-2, D.4.6, K.6.5, C.2, J.1

ISSN 0302-9743
ISBN-10 3-540-28114-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-28114-6 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11535218 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3621

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Preface

These are the proceedings of Crypto 2005, the 25th Annual International Cryptology Conference. The conference was sponsored by the International Association for Cryptologic Research (IACR) in cooperation with the IEEE Computer Science Technical Committee on Security and Privacy and the Computer Science Department of the University of California at Santa Barbara. The conference was held in Santa Barbara, California, August 14–18, 2005.

The conference received 178 submissions, out of which the program committee selected 33 for presentation. The selection process was carried out by the program committee via an "online" meeting. The authors of selected papers had a few weeks to prepare final versions of their papers, aided by comments from the reviewers. However, most of these revisions were not subject to any editorial review.

This year, a "Best Paper Award" was given to Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu, for their paper "Finding Collisions in the Full SHA-1."

The conference program included two invited lectures. Ralph Merkle delivered an IACR Distinguished Lecture, entitled "The Development of Public Key Cryptography: a Personal View; and Thoughts on Nanotechnology." Dan Boneh gave an invited talk, entitled "Bilinear Maps in Cryptography."

We continued the tradition of a "rump session," featuring short, informal presentations (usually serious, sometimes entertaining, and occasionally both). The rump session was chaired this year by Phong Q. Nguyễn.

I would like to thank everyone who contributed to the success of this conference. First, thanks to all the authors who submitted papers: a conference program is no better than the quality of the submissions (and hopefully, no worse). Second, thanks to all the members of the program committee: it was truly an honor to work with a group of such talented and hard working individuals. Third, thanks to all the external reviewers (listed below) for assisting the program committee: their expertise was invaluable. Fourth, thanks to Matt Franklin, Dan Boneh, Jan Camenisch, and Christian Cachin for sharing with me their experiences as previous Crypto and Eurocrypt program chairs. Finally, thanks to my wife, Miriam, and my children, Alec and Nicol, for their love and support, and for putting up with all of this.

June 2005

Victor Shoup

CRYPTO 2005

August 14–18, 2005, Santa Barbara, California, USA

Sponsored by the
International Association for Cryptologic Research (IACR)

in cooperation with
IEEE Computer Society Technical Committee on Security and Privacy,
Computer Science Department, University of California, Santa Barbara

General Chair

Stuart Haber, HP Labs, USA

Program Chair

Victor Shoup, New York University, USA

Program Committee

Masayuki Abe ..... NTT Information Sharing Platform Laboratories, Japan
Boaz Barak ..... Institute for Advanced Study & Princeton University, USA
Amos Beimel ..... Ben-Gurion University, Israel
Alex Biryukov ..... Katholieke Universiteit Leuven, Belgium
John Black ..... University of Colorado at Boulder, USA
Alexandra Boldyreva ..... Georgia Institute of Technology, USA
Jan Camenisch ..... IBM Zurich Research Laboratory, Switzerland
Jean-Sébastien Coron ..... University of Luxembourg, Luxembourg
Craig Gentry ..... DoCoMo USA Labs, USA
Shai Halevi ..... IBM T. J. Watson Research Center, USA
Stanislaw Jarecki ..... University of California at Irvine, USA
Antoine Joux ..... DGA & Univ. Versailles St-Quentin, France
Jonathan Katz ..... University of Maryland, USA
Arjen Lenstra .. Lucent Technologies, USA & TU Eindhoven, The Netherlands
Yehuda Lindell ..... Bar-Ilan University, Israel
Tal Malkin ..... Columbia University, USA
Ilya Mironov ..... Microsoft Research, USA
David Naccache ..... Gemplus, France & Royal Holloway, UK
Moni Naor ..... Weizmann Institute of Science, Israel
Leonid Reyzin ..... Boston University, USA
Louis Salvail ..... Aarhus Universitet, Denmark
Alice Silverberg ..... University of California at Irvine, USA
Adam Smith ..... Weizmann Institute of Science, Israel
Rebecca Wright ..... Stevens Institute of Technology, USA

Advisory Members

Matt Franklin (Crypto 2004 Program Chair) ..... UC Davis, USA
Cynthia Dwork (Crypto 2006 Program Chair) ..... Microsoft Research, USA

External Reviewers

Luis von Ahn
Jesus F. Almansa
Michael Anshel
Frederik Armknecht
Michael Backes
Endre Bangerter
Paulo Barreto
Donald Beaver
Mihir Bellare
Daniel J. Bernstein
Bhargav Bhatt
Ian F. Blake
Daniel Bleichenbacher
Dan Boneh
Xavier Boyen
An Braeken
Eric Brier
Christian Cachin
Ran Canetti
Pascale Charpin
Melissa Chase
Benoit Chevallier-Mames
Martin Cochran
Nicolas Courtois
Ivan Damgård
Christophe De Cannière
Nenad Dedić
Michael de Mare
Claudia Diaz
Xuhua Ding
Hans Dobbertin
Yevgeniy Dodis
Iwan Duursma
Ariel Elbaz
Michael Engling
Marc Fischlin
Matthias Fitzi
Gerhard Frey
Eiichiro Fujisaki
Steven Galbraith
Juan Garay
Rosario Gennaro
Daniel Gottesman
Louis Goubin
Prateek Gupta
Helena Handschuh
Jonathan Herzog
Susan Hohenberger
Omer Horvitz
Nick Howgrave-Graham
Jim Hughes
Dae Hyun Yum
Yuval Ishai
Geetha Jagannathan
Marc Joye
Charanjit Jutla
Yael Tauman Kalai
Alexander Kholosha
Chiu-Yuen Koo
Hugo Krawczyk
Kaoru Kurosawa
Eyal Kushilevitz
Tanja Lange
Joseph Lano
Gregor Leander
Homin Lee
Wen-Ching Winnie Li
Anna Lysyanskaya
David M'Raihi
Phil Mackenzie
John Malone-Lee
Alexander May
Daniele Micciancio
Sara Miner More
Tal Moran
Shiho Moriai
Ryan Moriarty
Frédéric Muller
Kumar Murty
Steven Myers
Anderson Nascimento
Antonio Nicolosi
Jesper Buus Nielsen
Kobbi Nissim
Kazuo Ohta
Tatsuaki Okamoto
Siddika Berna Örs
Pascal Paillier
Matthew Parker
Rafael Pass
Thomas B. Pedersen
Krzysztof Pietrzak
Benny Pinkas
David Pointcheval
Joern-Mueller Quade
Tal Rabin
Zulfikar Ramzan
Omer Reingold
Pankaj Rohatgi
Guy Rothblum
Karl Rubin
Andreas Ruttor
Christian Schaffner
Berry Schoenmakers
Hovav Shacham
abhi shelat
Vitaly Shmatikov
Thomas Shrimpton
Hervé Sibert
Andrey Sidorenko
Nigel Smart
Dieter Sommer
Martijn Stam
Douglas R. Stinson
Koutarou Suzuki
Emmanuel Thomé
Eran Tromer
Fréderik Vercauteren
Eric Verheul
Emanuele Viola
Andrew Wan
Bogdan Warinschi
Hoeteck Wee
Benne de Weger
Enav Weinreb
Stephen Weis
Susanne Wetzel
Claire Whelan
Christopher Wolf
Nikolai Yakovenko
Shoko Yonezawa
Moti Yung
Sheng Zhong

Lecture Notes in Computer Science

For information about Vols. 1–3508

please contact your bookseller or Springer

Vol. 3632: R. Nieuwenhuis (Ed.), Automated Deduction – CADE-20. XIII, 459 pages. 2005. (Subseries LNAI).

Vol. 3626: B. Ganter, G. Stumme, R. Wille (Eds.), Formal Concept Analysis. X, 349 pages. 2005. (Subseries LNAI).

Vol. 3621: V. Shoup (Ed.), Advances in Cryptology – CRYPTO 2005. XI, 568 pages. 2005.

Vol. 3615: B. Ludäscher, L. Raschid (Eds.), Data Integration in the Life Sciences. XII, 344 pages. 2005. (Subseries LNBI).

Vol. 3607: J.-D. Zucker, L. Saitta (Eds.), Abstraction, Reformulation and Approximation. XII, 376 pages. 2005. (Subseries LNAI).

Vol. 3598: H. Murakami, H. Nakashima, H. Tokuda, M. Yasumura, Ubiquitous Computing Systems. XIII, 275 pages. 2005.

Vol. 3597: S. Shimojo, S. Ichii, T.W. Ling, K.-H. Song (Eds.), Web and Communication Technologies and Internet-Related Social Issues - HSI 2005. XIX, 368 pages. 2005.

Vol. 3596: F. Dau, M.-L. Mugnier, G. Stumme (Eds.), Conceptual Structures: Common Semantics for Sharing Knowledge. XI, 467 pages. 2005. (Subseries LNAI).

Vol. 3594: J.C. Setubal, S. Verjovski-Almeida (Eds.), Advances in Bioinformatics and Computational Biology. XIV, 258 pages. 2005. (Subseries LNBI).

Vol. 3587: P. Perner, A. Imiya (Eds.), Machine Learning and Data Mining in Pattern Recognition. XVII, 695 pages. 2005. (Subseries LNAI).

Vol. 3586: A.P. Black (Ed.), ECOOP 2005 - Object-Oriented Programming. XVII, 631 pages. 2005.

Vol. 3584: X. Li, S. Wang, Z.Y. Dong (Eds.), Advanced Data Mining and Applications. XIX, 835 pages. 2005. (Subseries LNAI).

Vol. 3583: R.W. H. Lau, Q. Li, R. Cheung, W. Liu (Eds.), Advances in Web-Based Learning – ICWL 2005. XIV, 420 pages. 2005.

Vol. 3582: J. Fitzgerald, I.J. Hayes, A. Tarlecki (Eds.), FM 2005: Formal Methods. XIV, 558 pages. 2005.

Vol. 3581: S. Miksch, J. Hunter, E. Keravnou (Eds.), Artificial Intelligence in Medicine. XVII, 547 pages. 2005. (Subseries LNAI).

Vol. 3580: L. Caires, G.F. Italiano, L. Monteiro, C. Palamidessi, M. Yung (Eds.), Automata, Languages and Programming. XXV, 1477 pages. 2005.

Vol. 3579: D. Lowe, M. Gaedke (Eds.), Web Engineering. XXII, 633 pages. 2005.

Vol. 3578: M. Gallagher, J. Hogan, F. Maire (Eds.), Intelligent Data Engineering and Automated Learning - IDEAL 2005. XVI, 599 pages. 2005.

Vol. 3577: R. Falcone, S. Barber, J. Sabater-Mir, M.P. Singh (Eds.), Trusting Agents for Trusting Electronic Societies. VIII, 235 pages. 2005. (Subseries LNAI).

Vol. 3576: K. Etessami, S.K. Rajamani (Eds.), Computer Aided Verification. XV, 564 pages. 2005.

Vol. 3575: S. Wermter, G. Palm, M. Elshaw (Eds.), Biomimetic Neural Learning for Intelligent Robots. IX, 383 pages. 2005. (Subseries LNAI).

Vol. 3574: C. Boyd, J.M. González Nieto (Eds.), Information Security and Privacy. XIII, 586 pages. 2005.

Vol. 3573: S. Etalle (Ed.), Logic Based Program Synthesis and Transformation. VIII, 279 pages. 2005.

Vol. 3572: C. De Felice, A. Restivo (Eds.), Developments in Language Theory. XI, 409 pages. 2005.

Vol. 3571: L. Godo (Ed.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty. XVI, 1028 pages. 2005. (Subseries LNAI).

Vol. 3570: A. S. Patrick, M. Yung (Eds.), Financial Cryptography and Data Security. XII, 376 pages. 2005.

Vol. 3569: F. Bacchus, T. Walsh (Eds.), Theory and Applications of Satisfiability Testing. XII, 492 pages. 2005.

Vol. 3568: W.-K. Leow, M.S. Lew, T.-S. Chua, W.-Y. Ma, L. Chaisorn, E.M. Bakker (Eds.), Image and Video Retrieval. XVII, 672 pages. 2005.

Vol. 3567: M. Jackson, D. Nelson, S. Stirk (Eds.), Database: Enterprise, Skills and Innovation. XII, 185 pages. 2005.

Vol. 3566: J.-P. Banâtre, P. Fradet, J.-L. Giavitto, O. Michel (Eds.), Unconventional Programming Paradigms. XI, 367 pages. 2005.

Vol. 3565: G.E. Christensen, M. Sonka (Eds.), Information Processing in Medical Imaging. XXI, 777 pages. 2005.

Vol. 3564: N. Eisinger, J. Małuszyński (Eds.), Reasoning Web. IX, 319 pages. 2005.

Vol. 3562: J. Mira, J.R. Álvarez (Eds.), Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach, Part II. XXIV, 636 pages. 2005.

Vol. 3561: J. Mira, J.R. Álvarez (Eds.), Mechanisms, Symbols, and Models Underlying Cognition, Part I. XXIV, 532 pages. 2005.

Vol. 3560: V.K. Prasanna, S. Iyengar, P.G. Spirakis, M. Welsh (Eds.), Distributed Computing in Sensor Systems. XV, 423 pages. 2005.

Vol. 3559: P. Auer, R. Meir (Eds.), Learning Theory. XI, 692 pages. 2005. (Subseries LNAI).

Vol. 3558: V. Torra, Y. Narukawa, S. Miyamoto (Eds.), Modeling Decisions for Artificial Intelligence. XII, 470 pages. 2005. (Subseries LNAI).

Vol. 3557: H. Gilbert, H. Handschuh (Eds.), Fast Software Encryption. XI, 443 pages. 2005.

Vol. 3556: H. Baumeister, M. Marchesi, M. Holcombe (Eds.), Extreme Programming and Agile Processes in Software Engineering. XIV, 332 pages. 2005.

Vol. 3555: T. Vardanega, A.J. Wellings (Eds.), Reliable Software Technology – Ada-Europe 2005. XV, 273 pages. 2005.

Vol. 3554: A. Dey, B. Kokinov, D. Leake, R. Turner (Eds.), Modeling and Using Context. XIV, 572 pages. 2005. (Subseries LNAI).

Vol. 3553: T.D. Hämäläinen, A.D. Pimentel, J. Takala, S. Vassiliadis (Eds.), Embedded Computer Systems: Architectures, Modeling, and Simulation. XV, 476 pages. 2005.

Vol. 3552: H. de Meer, N. Bhatti (Eds.), Quality of Service – IWQoS 2005. XVIII, 400 pages. 2005.

Vol. 3551: T. Härder, W. Lehner (Eds.), Data Management in a Connected World. XIX, 371 pages. 2005.

Vol. 3548: K. Julisch, C. Kruegel (Eds.), Intrusion and Malware Detection and Vulnerability Assessment. X, 241 pages. 2005.

Vol. 3547: F. Bomarius, S. Komi-Sirviö (Eds.), Product Focused Software Process Improvement. XIII, 588 pages. 2005.

Vol. 3546: T. Kanade, A. Jain, N.K. Ratha (Eds.), Audio- and Video-Based Biometric Person Authentication. XX, 1134 pages. 2005.

Vol. 3544: T. Higashino (Ed.), Principles of Distributed Systems. XII, 460 pages. 2005.

Vol. 3543: L. Kutvonen, N. Alonistioti (Eds.), Distributed Applications and Interoperable Systems. XI, 235 pages. 2005.

Vol. 3542: H.H. Hoos, D.G. Mitchell (Eds.), Theory and Applications of Satisfiability Testing. XIII, 393 pages. 2005.

Vol. 3541: N.C. Oza, R. Polikar, J. Kittler, F. Roli (Eds.), Multiple Classifier Systems. XII, 430 pages. 2005.

Vol. 3540: H. Kalviainen, J. Parkkinen, A. Kaarna (Eds.), Image Analysis. XXII, 1270 pages. 2005.

Vol. 3539: K. Morik, J.-F. Boulicaut, A. Siebes (Eds.), Local Pattern Detection. XI, 233 pages. 2005. (Subseries LNAI).

Vol. 3538: L. Ardissono, P. Brna, A. Mitrovic (Eds.), User Modeling 2005. XVI, 533 pages. 2005. (Subseries LNAI).

Vol. 3537: A. Apostolico, M. Crochemore, K. Park (Eds.), Combinatorial Pattern Matching. XI, 444 pages. 2005.

Vol. 3536: G. Ciardo, P. Darondeau (Eds.), Applications and Theory of Petri Nets 2005. XI, 470 pages. 2005.

Vol. 3535: M. Steffen, G. Zavattaro (Eds.), Formal Methods for Open Object-Based Distributed Systems. X, 323 pages. 2005.

Vol. 3534: S. Spaccapietra, E. Zimányi (Eds.), Journal on Data Semantics III. XI, 213 pages. 2005.

Vol. 3533: M. Ali, F. Esposito (Eds.), Innovations in Applied Artificial Intelligence. XX, 858 pages. 2005. (Subseries LNAI).

Vol. 3532: A. Gómez-Pérez, J. Euzenat (Eds.), The Semantic Web: Research and Applications. XV, 728 pages. 2005.

Vol. 3531: J. Ioannidis, A. Keromytis, M. Yung (Eds.), Applied Cryptography and Network Security. XI, 530 pages. 2005.

Vol. 3530: A. Prinz, R. Reed, J. Reed (Eds.), SDL 2005: Model Driven. XI, 361 pages. 2005.

Vol. 3528: P.S. Szczepaniak, J. Kacprzyk, A. Niewiadomski (Eds.), Advances in Web Intelligence. XVII, 513 pages. 2005. (Subseries LNAI).

Vol. 3527: R. Morrison, F. Oquendo (Eds.), Software Architecture. XII, 263 pages. 2005.

Vol. 3526: S. B. Cooper, B. Löwe, L. Torenvliet (Eds.), New Computational Paradigms. XVII, 574 pages. 2005.

Vol. 3525: A.E. Abdallah, C.B. Jones, J.W. Sanders (Eds.), Communicating Sequential Processes. XIV, 321 pages. 2005.

Vol. 3524: R. Barták, M. Milano (Eds.), Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. XI, 320 pages. 2005.

Vol. 3523: J.S. Marques, N. Pérez de la Blanca, P. Pina (Eds.), Pattern Recognition and Image Analysis, Part II. XXVI, 733 pages. 2005.

Vol. 3522: J.S. Marques, N. Pérez de la Blanca, P. Pina (Eds.), Pattern Recognition and Image Analysis, Part I. XXVI, 703 pages. 2005.

Vol. 3521: N. Megiddo, Y. Xu, B. Zhu (Eds.), Algorithmic Applications in Management. XIII, 484 pages. 2005.

Vol. 3520: O. Pastor, J. Falcão e Cunha (Eds.), Advanced Information Systems Engineering. XVI, 584 pages. 2005.

Vol. 3519: H. Li, P. J. Olver, G. Sommer (Eds.), Computer Algebra and Geometric Algebra with Applications. IX, 449 pages. 2005.

Vol. 3518: T.B. Ho, D. Cheung, H. Liu (Eds.), Advances in Knowledge Discovery and Data Mining. XXI, 864 pages. 2005. (Subseries LNAI).

Vol. 3517: H.S. Baird, D.P. Lopresti (Eds.), Human Interactive Proofs. IX, 143 pages. 2005.

Vol. 3516: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part III. LXIII, 1143 pages. 2005.

Vol. 3515: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part II. LXIII, 1101 pages. 2005.

Vol. 3514: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part I. LXIII, 1089 pages. 2005.

Vol. 3513: A. Montoyo, R. Muñoz, E. Métais (Eds.), Natural Language Processing and Information Systems. XII, 408 pages. 2005.

Vol. 3512: J. Cabestany, A. Prieto, F. Sandoval (Eds.), Computational Intelligence and Bioinspired Systems. XXV, 1260 pages. 2005.

Vol. 3511: U.K. Wiil (Ed.), Metainformatics. VIII, 221 pages. 2005.

Vol. 3510: T. Braun, G. Carle, Y. Koucheryavy, V. Tsaoussidis (Eds.), Wired/Wireless Internet Communications. XIV, 366 pages. 2005.

Vol. 3509: M. Jünger, V. Kaibel (Eds.), Integer Programming and Combinatorial Optimization. XI, 484 pages. 2005.

Table of Contents

Efficient Collision Search Attacks on SHA-0

Xiaoyun Wang[1,*], Hongbo Yu[2], and Yiqun Lisa Yin[3]

[1] Shandong University, China
xywang@sdu.edu.cn
[2] Shandong University, China
yhb@mail.sdu.edu.cn
[3] Independent Security Consultant, Greenwich CT, US
yyin@princeton.edu

Abstract. In this paper, we present new techniques for collision search in the hash function SHA-0. Using the new techniques, we can find collisions of the full 80-step SHA-0 with complexity less than 2^{39} hash operations.

Keywords: Hash functions, Collision search attacks, SHA-0, SHA-1.

1 Introduction

The hash function SHA-0 was issued in 1993 as a federal standard by NIST. A revised version called SHA-1 was later issued in 1995 as a replacement for SHA-0. The only difference between the two hash functions is the additional rotation operation in the message expansion of SHA-1, which is supposed to provide more security. Both hash functions are based on the design principles of MD4.

In 1997, Wang found an attack on SHA-0 [14] which produces a collision with probability 2^{-58} by utilizing algebraic methods to derive a collision differential path. In 1998, Chabaud and Joux [6] independently found the same differential path through computer search. In August 2004, Joux [7] announced the first real collision of SHA-0, which consists of four message blocks (a pair of 2048-bit input messages). The collision search took about 80,000 hours of CPU time (three weeks of real time) and is estimated to have a complexity of about 2^{51} hash operations. To our knowledge, this is the best existing attack on the full 80-step SHA-0 prior to the work reported here.

The attacks in [14,6] found a differential path which is composed of certain 6-step local collisions. There is an obstacle to further improve these attacks, as finding a differential characteristic for two consecutive local collisions corresponding to two consecutive disturbances in the first round turns out to be impossible. This phenomenon makes it difficult to find a differential path which has a smaller number of local collisions in rounds 2-4 and no consecutive local collisions in the first round.

* Supported by the National Natural Science Foundation of China (NSFC Grant No.90304009) and Program for New Century Excellent Talents in University.

V. Shoup (Ed.): Crypto 2005, LNCS 3621, pp. 1–16, 2005.

In this paper, we introduce a new cryptanalytic method to cope with this difficulty. Our analysis includes the following techniques: Firstly, we identify an "impossible" differential path with few local collisions in rounds 2-4 and some consecutive local collisions in round 1. Secondly, we transform the impossible differential path into a possible one. Thirdly, we derive a set of conditions which guarantee that the modified differential path holds. Finally, we design message modifications to correct all the unfulfilled conditions in the first round as well as some such conditions in the second round. With these techniques, we can find collisions of the full SHA-0 with at most 2^{39} hash operations, which is a major improvement over existing attacks. The same techniques can be used to find near collisions of SHA-0 with complexity about 2^{33} hash operations.

We note that the new techniques have also been proven to be effective in the analysis of SHA-1[16].

The rest of the paper is organized as follows. In Section 2, we give a description of SHA-0. In Section 3, we provide an overview of the original attack on SHA-0 [14] and subsequent improvements [15,1,2,7,3]. In Section 4, we review the "message modification techniques" presented in [11,12,13] to break HAVA-128, MD5, MD4 and RIPEMD, and consider their effectiveness in improving existing attacks on SHA-0. In Section 5, we present our new collision search attacks on SHA-0. In Section 6, we give an example of real collision of SHA-0 found by computer search using the new techniques. We conclude the paper in Section 7.

2 Description of SHA-0

The hash function SHA-0 takes a message of length less than 2^{64} bits and produces a 160-bit hash value. The input message is padded and then processed in 512-bit blocks in the Damgård/Merkle iterative structure. Each iteration invokes a so-called compression function which takes a 160-bit chaining value and a 512-bit message block and outputs another 160-bit chaining value. The initial chaining value (called IV) is a set of fixed constants, and the final chaining value is the hash of the message.

In what follows, we describe the compression function of SHA-0. For each 512-bit block of the padded message, divide it into 16 32-bit words, $(m_0, m_1, ..., m_{15})$. The message words are first expanded as follows: for $i = 16, ..., 79$,

$$m_i = m_{i-3} \oplus m_{i-8} \oplus m_{i-14} \oplus m_{i-16}.$$

The expanded message words are then processed in four rounds, each consisting of 20 steps. The step function is defined as follows.

For $i = 1, 2, ..., 80$,

$$\begin{aligned} a_i &= (a_{i-1} << 5) + f_i(b_{i-1}, c_{i-1}, d_{i-1}) + e_{i-1} + m_{i-1} + k_i \\ b_i &= a_{i-1} \\ c_i &= b_{i-1} << 30 \\ d_i &= c_{i-1} \\ e_i &= d_{i-1} \end{aligned}$$

The initial chaining value $IV = (a_0, b_0, c_0, d_0, e_0)$ is defined as:

$$(0x67452301,\ 0xefcdab89,\ 0x98badcfe,\ 0x10325476,\ 0xc3d2e1f0)$$

Each round employs a different Boolean function f_i and constant k_i, which is summarized in Table 1.

Table 1. Boolean functions and constants in SHA-0

rounds	steps	Boolean function f_i	constant k_i
1	$1-20$	IF: $(x \wedge y) \vee (\neg x \wedge z)$	0x5a827999
2	$21-40$	XOR: $x \oplus y \oplus z$	0x6ed6eba1
3	$41-60$	MAJ: $(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$	0x8fabbcdc
4	$61-80$	XOR: $x \oplus y \oplus z$	0xca62c1d6

3 Previous Attacks on SHA-0

In this section, we first describe the original collision attack on SHA-0 given by Wang in 1997 [14]. This sets up the basic framework for introducing our new techniques later on. For other independent attacks on SHA-0 the reader may wish to refer to [15,6,1,7,3].

3.1 Local Collisions of SHA-0

Informally, a local collision is a collision within a few steps of the hash function. A simple yet very important observation is that SHA-0 has a 6-step local collision that can start at any step i, and this type of local collision is the basic component in constructing full collisions.

Suppose a message difference in bit j first occurs in Step i (e.g., $\Delta m_{i-1,j} = 1$.) The difference will affect the chaining variables a, b, c, d, e consecutively in the next five steps. In order to offset these differences and reach a local collision, more message differences are introduced in subsequent message words. In Table 2, we illustrate the differential path of such a local collision. The chaining variable conditions under which the local collisions hold were given in [14,15].

The probability associated with the above local collision depends on the Boolean function, the bit position j, and some conditions on the message bits. The differential attack in [14] and [6] chooses $j = 2$ so that $j + 30$ becomes the MSB[1] to eliminate the carry effect in the last three steps. In addition, the following condition

$$m_{i,2} = \neg m_{i+1,7}$$

[1] Throughout this paper, we label the bit positions in a 32-bit word as $32, 31, 30, ..., 3, 2, 1$, where bit 32 is the most significant bit and bit 1 is the least significant bit. Please note that this is different from the convention of labelling bit positions from 31 to 0.

Table 2. A 6-step local collision of SHA-0 starting at step i. The measure of difference is $\oplus$. Addition in the exponents is modulo 32. "nc" stands for no carry. Δf is the output difference of the Boolean function

step	Δm	Δa	Δb	Δc	Δd	Δe	Conditions
i	2^j	2^j					nc
$i+1$	2^{j+5}		2^j				
$i+2$	2^j			2^{j+30}			nc, $\Delta f = 2^j$
$i+3$	2^{j+30}				2^{j+30}		nc, $\Delta f = 2^{j+30}$
$i+4$	2^{j+30}					2^{j+30}	nc, $\Delta f = 2^{j+30}$
$i+5$	2^{j+30}						nc

helps to offset completely the chaining variable difference in the second step of the local collision, where $x_{i,j}$ $(x = m)$ denotes the j-th bit of message word x_i.

The message condition in round 3

$$m_{i,2} = \neg m_{i+2,2}$$

helps to offset the difference caused by the non-linear function in the third step of the local collision.

3.2 Differential Paths of SHA-0

At a high level, the differential path used in [14] is a sequence of local collisions joined together with possible overlaps. To construct such a path, we need to find a set of appropriate starting step for each local collision. We can use an 80-bit 0-1 vector $x = (x_0, ..., x_{79})$ to specify these starting steps, and the vector is called a *disturbance vector*. It is easy to show that the disturbance vector satisfies the same recursion defined by the message expansion. That is, for $i = 16, ...79$,

$$x_i = x_{i-3} \oplus x_{i-8} \oplus x_{i-14} \oplus x_{i-16}.$$

For the 80 variables x_i, any 16 consecutive ones determine the rest. So there are 16 free variables to be set for a total of 2^{16} possibilities.

In order for the disturbance vector to lead to a possible collision, several conditions on the disturbance vectors need to be imposed, and they are discussed in details in [14]. These conditions are summarized in Table 3.

From [15], we know condition 1 in Table 3 holds if and only if the following equations hold:

$$\begin{aligned}
x_{11} &= x_3 + x_8 \\
x_{12} &= x_4 + x_9 \\
x_{13} &= x_5 + x_{10} \\
x_{14} &= x_0 + x_3 + x_6 + x_8 \\
x_{15} &= x_1 + x_4 + x_7 + x_9
\end{aligned}$$

Table 3. Conditions on disturbance vectors for SHA-0 with t steps

	Condition	Purpose
1	$x_i = 0$ for $i = 75, 76, 77, 78, 79$	to produce a collision in the last step 5
2	$x_i = 0$ for $i = -5, ..., -1$	to avoid truncated local collisions in first few steps
3	no consecutive ones in the first 17 variables	to avoid an impossible collision path due to a property of IF

Condition 2 in Table 3 holds if and only if

$$\begin{aligned} x_6 &= x_0 + x_1 + x_2 + x_4 \\ x_7 &= x_0 + x_4 \\ x_8 &= x_0 + x_1 + x_5 \\ x_9 &= x_4 \\ x_{10} &= x_0 + x_5 \end{aligned}$$

We can also search for a disturbance vector using $(x_0, ..., x_{15})$ as the 16 variables. After imposing Conditions 1 and 2, there are 6 free variables remaining: $(x_0, ..., x_5)$. With Condition 3, only 3 choices are left for the 6 free variables, namely (001000) and (000100) and (000101), the first of which corresponds to the differential path given in [14].

We remark that the Hamming weight of the disturbance vector is closely related to the complexity of the attack. Given a disturbance vector x, we define $hw_{r+}(x)$ as the Hamming weight of x from step r to 80. To minimize the complexity, the Hamming weight $hw_{17+}(x)$ should as small as possible (although there are other more subtle conditions). The corresponding vector used in [14] have $hw_{17+} = 27$, and the complexity of collision search attack is about 2^{58}.

3.3 Existing Techniques for Improving the Attack

In the past year, there have been some major advances in the analysis of SHA-0. These latest attacks are built upon the differential attack by Chaband and Joux, while introducing new ideas for significant improvements. We summarize these techniques below.

- *Neutral bit techniques* [1]. This allows the collision search to start at a step $i > 17$. [2] Biham and Chen showed how to start the collision search of SHA-0 at step $i = 22$ [1] and reduce complexity of finding full collisions to 2^{56}.

  More interestingly, they were able to find near collisions of SHA-0 with complexity 2^{40}, and this provides a basis for finding multi-block collisions.

[2] Since the first 16 message words are independent, in general one can bypass the first 16 steps and start the search at $i = 17$.