Mitsuru Matsui
Robert Zuccherato (Eds.)

LNCS 3006

Selected Areas in Cryptography

10th Annual International Workshop, SAC 2003
Ottawa, Canada, August 2003
Revised Papers

Springer

Mitsuru Matsui Robert Zuccherato (Eds.)

Selected Areas in Cryptography

10th Annual International Workshop, SAC 2003
Ottawa, Canada, August 14-15, 2003
Revised Papers

Springer

Volume Editors

Mitsuru Matsui
Mitsubishi Electric Corporation
5-1-1 Ofuna, Kamakura, Kanagawa, 247-8501 Japan
E-mail: matsui@iss.isl.melco.co.jp

Robert Zuccherato
Entrust Inc.
1000 Innovation Drive, Ottawa, Ontario, Canada K2K 3E7
E-mail: robert.zuccherato@entrust.com

Library of Congress Control Number: 2004102410

CR Subject Classification (1998): E.3, D.4.6, K.6.5, F.2.1-2, C.2.0, H.4.3

ISSN 0302-9743
ISBN 3-540-21370-8 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by DA-TeX Gerd Blumenstein
Printed on acid-free paper SPIN: 10996386 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3006

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Preface

SAC 2003 was the tenth in a series of annual workshops on Selected Areas in Cryptography. This marked the third time that the workshop had been held at Carleton University in Ottawa with previous workshops being held there in 1995 and 1997. The intent of the SAC workshops is to provide a relaxed atmosphere in which researchers in cryptography can present and discuss new work on selected areas of current interest.

The themes for the SAC 2003 workshop were:

- design and analysis of symmetric key cryptosystems,
- primitives for symmetric key cryptography, including block and stream ciphers, hash functions, and MACs,
- efficient implementation of cryptographic systems in public and symmetric key cryptography,
- cryptographic solutions for Web services security,
- cryptography and security of trusted and distributed systems.

A total of 85 papers were submitted to SAC 2003, two of which were subsequently withdrawn. After a review process that had all papers reviewed by at least three referees, 25 papers were accepted for presentation at the workshop. We would like to thank all of the authors who submitted papers, whether or not those papers were accepted, for submitting their high-quality work to this workshop.

As well, we were fortunate to have the following two invited speakers at SAC 2003:

- Nicolas Courtois (Schlumberger Smart Cards)
  Algebraic attacks and design of block ciphers, stream ciphers, and multivariate public key schemes
- Virgil D. Gligor (University of Maryland)
  Cryptolight: Perspective and Status

SAC 2003 was memorable for all those involved, not only because of the quality of the technical program, but also because of the massive power blackout that occurred. On August 14, 2003 much of the eastern part of the United States, and most of the province of Ontario were plunged into darkness. The city of Ottawa was without power from about 4:00 pm on August 14 through most of the day on August 15. Despite the lack of power, the workshop carried on in an "unplugged" format with all remaining talks presented in a makeshift lecture hall using chalk and blackboards. The staff of the Tour and Conference Centre at Carleton University deserve special recognition for helping the chairs make alternate arrangements to deal with the blackout. We would also like to thank all SAC attendees and, in particular, the presenters who persevered and made SAC 2003 a success, despite the trying circumstances.

We appreciate the hard work of the SAC 2003 Program Committee. We are also very grateful to the many others who participated in the review process: Gildas Avoine, Florent Bersani, Alex Biryukov, Eric Brier, Jean-Sebastien Coron, Joan Daemen, Christophe De Canniere, Jean-François Dhem, Zhi (Judy) Fu, Virgil Gligor, Florian Hess, Don Johnson, Pascal Junod, Hans-Joachim Knobloch, Joe Lano, John Malone-Lee, Tom Messerges, Jean Monnerat, Svetla Nikova, Dan Page, Pascal Paillier, Matthew Parker, Holger Petersen, Michael Quisquater, Håvard Raddum, Christophe Tymen, Frederik Vercauteren, and Michael Wiener. We apologize for any unintended errors or omissions in this list.

We are also appreciative of the financial support provided by Carleton University, Cloakware Corporation, Entrust, Inc., Mitsubishi Electric, and Queen's University Kingston.

Special thanks are due to Sandy Dare for providing administrative assistance and to the local arrangements committee consisting of Mike Just, Tao Wan, and Dave Whyte for their help.

On behalf of all those involved in organizing the workshop, we thank all the workshop participants for making SAC 2003 a success!

January 2004 Mitsuru Matsui and Robert Zuccherato

Organization

Program Committee

Carlisle Adams	University of Ottawa, Canada
Steve Babbage	Vodafone, UK
Josh Benaloh	Microsoft, USA
Lily Chen	Motorola, USA
Henri Gilbert	France Telecom, France
Helena Handschuh	Gemplus, France
Lars Knudsen	Technical University of Denmark, Denmark
Mitsuru Matsui	Mitsubishi Electric, Japan (Co-chair)
Alfred Menezes	University of Waterloo, Canada
Markus Michels	Secorvo, Germany
Kaisa Nyberg	Nokia, Finland
Bart Preneel	Katholieke Universiteit Leuven, Belgium
Nigel Smart	University of Bristol, UK
Doug Stinson	University of Waterloo, Canada
Paul Van Oorschot	Carleton University, Canada
Serge Vaudenay	EPFL, Switzerland
Robert Zuccherato	Entrust, Canada (Co-chair)

Local Arrangements Committee

Mike Just, Tao Wan, Dave Whyte, Robert Zuccherato

Sponsoring Institutions

Carleton University
Cloakware Corporation
Entrust, Inc.
Mitsubishi Electric
Queen's University Kingston

Lecture Notes in Computer Science

For information about Vols. 1–2888

please contact your bookseller or Springer-Verlag

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004. (Subseries LNAI).

Vol. 2952: N. Guelfi, E. Astesiano, G. Reggio (Eds.), Scientific Engineering of Distributed Java Applications. X, 157 pages. 2004.

Vol. 2951: M. Naor (Ed.), Theory of Cryptography. XI, 523 pages. 2004.

Vol. 2949: R. De Nicola, G. Ferrari, G. Meredith (Eds.), Coordination Models and Languages. X, 323 pages. 2004.

Vol. 2948: G.L. Mullen, A. Poli, H. Stichtenoth (Eds.), Finite Fields and Applications. VIII, 263 pages. 2004.

Vol. 2947: F. Bao, R. Deng, J. Zhou (Eds.), Public Key Cryptography – PKC 2004. XI, 455 pages. 2004.

Vol. 2946: R. Focardi, R. Gorrieri (Eds.), Foundations of Security Analysis and Design II. VII, 267 pages. 2004.

Vol. 2943: J. Chen, J. Reif (Eds.), DNA Computing. X, 225 pages. 2004.

Vol. 2941: M. Wirsing, A. Knapp, S. Balsamo (Eds.), Radical Innovations of Software and Systems Engineering in the Future. X, 359 pages. 2004.

Vol. 2940: C. Lucena, A. Garcia, A. Romanovsky, J. Castro, P.S. Alencar (Eds.), Software Engineering for Multi-Agent Systems II. XII, 279 pages. 2004.

Vol. 2939: T. Kalker, I.J. Cox, Y.M. Ro (Eds.), Digital Watermarking. XII, 602 pages. 2004.

Vol. 2937: B. Steffen, G. Levi (Eds.), Verification, Model Checking, and Abstract Interpretation. XI, 325 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M. Paolucci (Eds.), Regulated Agent-Based Social Systems. X, 301 pages. 2004. (Subseries LNAI).

Vol. 2930: F. Winkler (Ed.), Automated Deduction in Geometry. VII, 231 pages. 2004. (Subseries LNAI).

Vol. 2929: H. de Swart, E. Orlowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments. VII, 273 pages. 2003.

Vol. 2926: L. van Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. XI, 428 pages. 2004. (Subseries LNAI).

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 365 pages. 2004. (Subseries LNAI).

Vol. 2919: E. Giunchiglia, A. Tacchella (Eds.), Theory and Applications of Satisfiability Testing. XI, 530 pages. 2004.

Vol. 2917: E. Quintarelli, Model-Checking Based Data Retrieval. XVI, 134 pages. 2004.

Vol. 2916: C. Palamidessi (Ed.), Logic Programming. XII, 520 pages. 2003.

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gesture-Based Communication in Human-Computer Interaction. XIII, 558 pages. 2004. (Subseries LNAI).

Vol. 2914: P.K. Pandya, J. Radhakrishnan (Eds.), FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science. XIII, 446 pages. 2003.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2003. XX, 512 pages. 2003. (Subseries LNAI).

Vol. 2911: T.M.T. Sembok, H.B. Zaman, H. Chen, S.R. Urs, S.H. Myaeng (Eds.), Digital Libraries: Technology and Management of Indigenous Knowledge for Global Access. XX, 703 pages. 2003.

Vol. 2910: M.E. Orlowska, S. Weerawarana, M.M.P. Papazoglou, J. Yang (Eds.), Service-Oriented Computing - ICSOC 2003. XIV, 576 pages. 2003.

Vol. 2909: R. Solis-Oba, K. Jansen (Eds.), Approximation and Online Algorithms. VIII, 269 pages. 2004.

Vol. 2908: K. Chae, M. Yung (Eds.), Information Security Applications. XII, 506 pages. 2004.

Vol. 2907: I. Lirkov, S. Margenov, J. Wasniewski, P. Yalamov (Eds.), Large-Scale Scientific Computing. XI, 490 pages. 2004.

Vol. 2906: T. Ibaraki, N. Katoh, H. Ono (Eds.), Algorithms and Computation. XVII, 748 pages. 2003.

Vol. 2905: A. Sanfeliu, J. Ruiz-Shulcloper (Eds.), Progress in Pattern Recognition, Speech and Image Analysis. XVII, 693 pages. 2003.

Vol. 2904: T. Johansson, S. Maitra (Eds.), Progress in Cryptology - INDOCRYPT 2003. XI, 431 pages. 2003.

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2003: Advances in Artificial Intelligence. XVI, 1075 pages. 2003. (Subseries LNAI).

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress in Artificial Intelligence. XV, 504 pages. 2003. (Subseries LNAI).

Vol. 2901: F. Bry, N. Henze, J. Ma luszyński (Eds.), Principles and Practice of Semantic Web Reasoning. X, 209 pages. 2003.

Vol. 2900: M. Bidoit, P.D. Mosses (Eds.), Casl User Manual. XIII, 240 pages. 2004.

Vol. 2899: G. Ventre, R. Canonico (Eds.), Interactive Multimedia on Next Generation Networks. XIV, 420 pages. 2003.

Vol. 2898: K.G. Paterson (Ed.), Cryptography and Coding. IX, 385 pages. 2003.

Vol. 2897: O. Balet, G. Subsol, P. Torguet (Eds.), Virtual Storytelling. XI, 240 pages. 2003.

Vol. 2896: V.A. Saraswat (Ed.), Advances in Computing Science – ASIAN 2003. VIII, 305 pages. 2003.

Vol. 2895: A. Ohori (Ed.), Programming Languages and Systems. XIII, 427 pages. 2003.

Vol. 2894: C.S. Laih (Ed.), Advances in Cryptology - ASIACRYPT 2003. XIII, 543 pages. 2003.

Vol. 2893: J.-B. Stefani, I. Demeure, D. Hagimont (Eds.), Distributed Applications and Interoperable Systems. XIII, 311 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept Graphs with Negation. XI, 213 pages. 2003. (Subseries LNAI).

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agents and Multi-Agent Systems. X, 215 pages. 2003. (Subseries LNAI).

Vol. 2890: M. Broy, A.V. Zamulin (Eds.), Perspectives of System Informatics. XV, 572 pages. 2003.

Vol. 2889: R. Meersman, Z. Tari (Eds.), On The Move to Meaningful Internet Systems 2003: OTM 2003 Workshops. XIX, 1071 pages. 2003.

Table of Contents

Cryptanalysis II

Cryptographic Primitives

Stream Ciphers

Efficient Implementation

Low Cost Security: Explicit Formulae for Genus-4 Hyperelliptic Curves

Jan Pelzl, Thomas Wollinger, and Christof Paar

Department of Electrical Engineering and Information Sciences
Communication Security Group (COSY)
Ruhr-Universität Bochum, Germany
{pelzl,wollinger,cpaar}@crypto.rub.de

Abstract. It is widely believed that genus four hyperelliptic curve cryptosystems (HECC) are not attractive for practical applications because of their complexity compared to systems based on lower genera, especially elliptic curves. Our contribution shows that for low cost security applications genus-4 hyperelliptic curves (HEC) can outperform genus-2 HEC and that we can achieve a performance similar to genus-3 HEC. Furthermore our implementation results show that a genus-4 HECC is an alternative cryptosystem to systems based on elliptic curves.
In the work at hand we present for the first time explicit formulae for genus-4 HEC, resulting in a 60% speed-up compared to the best published results. In addition we implemented genus-4 HECC on a Pentium4 and an ARM microprocessor. Our implementations on the ARM show that for genus four HECC are only a factor of 1.66 slower than genus-2 curves considering group order $\approx 2^{190}$. For the same group order ECC and genus-3 HECC are about a factor of 2 faster than genus-4 curves on the ARM. The two most surprising results are: 1) for low cost security application, namely considering an underlying group of order 2^{128}, HECC with genus 4 outperform genus-2 curves by a factor of 1.46 and has similar performance to genus-3 curves on the ARM and 2) when compared to genus-2 and genus-3, genus-4 HECC are better suited to embedded microprocessors than to general purpose processors.

Keywords: Hyperelliptic curves, genus four, explicit formulae, efficient implementation, low cost security, embedded application, comparison HECC vs. ECC

1 Introduction

It is widely recognized that data security will play a central role in the design of future IT systems. One of the major tools to provide information security is public-key cryptography. Additionally, one notices that more and more IT applications are realized as embedded systems. In fact, 98% of all microprocessors sold today are embedded in household appliances, vehicles, and machines on factory floors [9, 3], whereas only 2% are used in PCs and workstations. Embedded

M. Matsui and R. Zuccherato (Eds.): SAC 2003, LNCS 3006, pp. 1–16, 2004.

processors have a 100 – 1000 times lower computational power than conventional PCs. In addition to many other challenges, the integration of security and privacy in the existing and new embedded applications will be a major one.

Since the invention of public-key (PK) cryptography in 1976, three different variants of PK cryptosystems of practical relevance have been introduced, namely cryptosystems based on the difficulty of integer factorization (e.g. RSA [36]), solving the discrete logarithm problem in finite fields (e.g. Diffie-Hellman [6]), and the discrete logarithm problem (DLP) in the group of points of an elliptic curve (EC) over a finite field [29, 17]. Hyperelliptic curve cryptosystems (HECC) are a generalization of elliptic curve cryptosystems (ECC) that were suggested in 1988 for cryptographic applications [18].

Considering the implementation aspects of the three public-key variants, one notices that a major difference is the bit-length of the operands. It is widely accepted that for commercial applications one needs 1024-bit operands for RSA or Diffie-Hellman. In the case of ECC or HECC applications, a group order of size $\approx 2^{160}$ is believed to be sufficient for moderate long-term security. In this contribution we consider genus-4 HECC over $\mathbb{F}_q$ and therefore we will need at least $4 \cdot \log_2 q \approx 2^{160}$. In particular, for these curves, we will need a field $\mathbb{F}_q$ with $|\mathbb{F}_q| \approx 2^{40}$, i.e., 40-bit long operands. However, in many low cost and embedded applications lower security margins are adequate. In practice, if a group order of 2^{128} is sufficient, the operations can be performed with an operand length of 32-bit. Thus, the underlying field operations can be implemented very efficiently if working with 32-bit microprocessors (e.g. ARM). It is important to point out that the small field sizes and the resulting short operand size of HECC compared to other cryptosystems makes HECC specially promising for the use in embedded environments. We discuss the security of such curves in Section 4.2.

Our Contributions

The work at hand presents for the first time explicit formulae for genus-4 curves. Genus-4 HECC did not draw a lot of attention in the past because they seem to be far less efficient than genus-2 HECC, genus-3 HECC, and ECC. Our contribution is a major step in accelerating this kind of cryptosystem and contrary to common belief we were able to develop explicit formulae that perform the scalar multiplication 72% and 60% faster than previous work by Cantor [5] and Nagao [32], respectively.

Genus-4 HECC are well suited for the implementation of public-key cryptosystems in constrained environments because the underlying arithmetic is performed with relatively small operand bit-lengths. In this contribution, we present our implementation of this cryptosystem on an ARM and a Pentium microprocessor. We were able to perform a 160bit scalar multiplication in 172 msec on the ARM@80MHz and in 6.9 msec on the Pentium4@1.8GHz. In addition, our implementations show, that genus-4 HECC are only a factor of 1.66 and 2.08 slower than genus-2 and genus-3 curves considering group order of $\approx 2^{190}$, respectively. Compared to ECC, the genus-4 HECC are a factor of 2 slower for the same group order .

Genus-4 HEC are well suited, especially for cryptographic applications with short term security. Performing arithmetic with 32-bit operands only, genus-4 HECC allow for a security comparable to of 128-bit ECC. We implemented genus-4 HECC with underlying field arithmetic for 32-bit. In this case one is able to perform arithmetic with only one word. Contrary to the general case, the implementation of genus-4 curves in groups of order $\approx 2^{128}$ outperform genus-2 curves by a factor of about 1.5. Furthermore, our implementation shows that, HECC with genus three and four have similar performance considering the group order $\approx 2^{128}$.

The remainder of the paper is organized as follows. Section 2 summarizes contributions dealing with previous implementations and efficient formulae of genus-4 HECC. Section 3 gives a brief overview of the mathematical background related to HECC and Section 4 considers the security of the implemented HECCs. Sections 5 and 6 present our new explicit formulae for genus-4 curves and methodology used for our implementation. Finally, we end this contribution with a discussion of our results and some conclusions.

2 Previous Work

We will first summarize previous improvements on genus-4 HEC group operations and second introduce implementations published in earlier contributions.

Improvements to HECC Group Operations of Genus-4 HECC Cantor [5] presented algorithms to perform the group operations on HEC in 1987. In recent years, there has been extensive research being performed to speed up the group operations on genus two HECC [32, 16, 27, 30] [43, 23, 24, 25] and genus three [32, 22, 34].
Only Nagao [32] tried to improve Cantor's algorithm for higher genera.

Nagao evaluated the computational cost of the group operations by applying the stated improvements for genus $2 \leq g \leq 10$. The most efficient group addition for genus-4 curves needs 2I + 289M/S or 3I + 286M/S (depending on the cost of the field inversion compared to multiplications, one or the other is more efficient). I refers to field inversion, M to field multiplication, S to field squaring, and M/S to field multiplications or squarings, since squarings are assumed to be of the same complexity as multiplications in these publications. For the computation of a group doubling in genus-4 curves one has to perform 2I + 268M/S or 3I + 260M/S. Notice that the ideas proposed by [32] are used to improve polynomial arithmetic.

Genus-4 HECC Implementations Since HECC were proposed, there have been several software implementations on general purpose machines [21, 38] [42, 39, 27, 30, 22, 23] and publications dealing with hardware implementations of HECC [46, 4]. Only very recently work dealing with the implementation of HECC on embedded systems was published in [33, 34].

Table 1. Execution times of recent HEC implementations in software

reference	processor	genus	field	$t_{scalarmult.}$ in ms
[21]	Pentium@100MHz	4	$\mathbb{F}_{2^{31}}$	1100
[38]	Alpha@467MHz	4	$\mathbb{F}_{2^{41}}$	96.6
	Pentium-II@300MHz	4	$\mathbb{F}_{2^{41}}$	10900
[39]	Alpha21164A@600MHz	4	$\mathbb{F}_{2^{41}}$	43

The results of previous genus-4 HECC software implementations are summarized in Table 1. All implementations use Cantor's algorithm with polynomial arithmetic. We remark that the contribution at hand is the first genus-4 HECC implementation based on explicit formulae.

3 Mathematical Background

The mathematical background described in this section is limited to the material that is required in our contribution. The interested reader is referred to [19, 28, 20] for more details.

3.1 HECC and the Jacobian

Let $\mathbb{F}$ be a finite field, and let $\overline{\mathbb{F}}$ be the algebraic closure of $\mathbb{F}$. A hyperelliptic curve C of genus $g \geq 1$ over $\mathbb{F}$ is the set of solutions $(u, v) \in \mathbb{F} \times \mathbb{F}$ to the equation

$$C : v^2 + h(u)v = f(u)$$

The polynomial $h(u) \in \mathbb{F}[u]$ is of degree at most g and $f(u) \in \mathbb{F}[u]$ is a monic polynomial of degree $2g + 1$. For odd characteristic it suffices to let $h(u) = 0$ and to have $f(u)$ square free.

A divisor $D = \sum m_i P_i$, $m_i \in \mathbb{Z}$, is a finite formal sum of $\overline{\mathbb{F}}$-points. The set of divisors of degree zero will be denoted by $\mathbb{D}^0$. Every rational function on the curve gives rise to a divisor of degree zero and is called principal. The the set of all principal divisors is denoted by $\mathbb{P}$. We can define the Jacobian of C over $\mathbb{F}$, denoted by $\mathbb{J}_C(\mathbb{F})$ as the quotient group $\mathbb{D}^0/\mathbb{P}$.

In [5] it is shown that the divisors of the Jacobian can be represented as a pair of polynomials $a(u)$ and $b(u)$ with $\deg b(u) < \deg a(u) \leq g$, with $a(u)$ dividing $b(u)^2 + h(u)b(u) - f(u)$ and where the coefficients of $a(u)$ and $b(u)$ are elements of $\mathbb{F}$ [31]. In the remainder of this paper, a divisor D represented by polynomials will be denoted by $div(a, b)$.

3.2 Group Operations in the Jacobian

This section gives a brief description of the algorithms used for adding and doubling divisors on $\mathbb{J}_C(\mathbb{F})$. Algorithm 1 describes the group addition. Doubling a divisor is easier than general addition and therefore, Steps 1,2, and 3 of Algorithm 1 can be simplified as follows: