Libraries
Part II

Yung-Hsiang Liu

Department of Computer Science, National Chiao Tung University

ECC 2009
Outline

- MIRACL: Multiprecision Integer and Rational Arithmetic C/C++ Library
  - Schoof & SEA
  - Pairing-friendly Curves & CM
  - Pairings

- A Variety of ECC Libraries
  - libecc, ECC-LIB, PBC Library, LiDIA, Crypto++, ECC and Bilinear Pairing Software Library

- Homework Assignment
MIRACL

- How to Get?
- How to Build?
- How to Use?

- Schoof / SEA
- Pairing-friendly Curves
- Complex Multiplication
- Pairings
How to Get?

- [http://www.shamus.ie/](http://www.shamus.ie/)

Current version 5.4. Get it now — [MIRACL](http://www.shamus.ie/) last updated March 2009, but first check out the [README](http://www.shamus.ie/) file.

Mirror [MIRACL](http://www.shamus.ie/) and [README](http://www.shamus.ie/) in case you need to download.

MIRACL has been under continuous development since 1988. It is currently licensed to many major companies in the US, Brazil, Britain, Germany, France, Switzerland, South Africa and Australia.

MIRACL is [FREE](http://www.shamus.ie/) for non-profit making, educational, or any non-commercial use. In other words if you aren't going to make any money out of it, we don't expect to either. Basically
How to Build?

- Windows...
- Linux...
- Other platforms...
How to Use?

- Windows...
- Linux...
- Other platforms...
Elliptic Curves


MIRACL offers full support for Elliptic Curve Cryptography (ECC) over the prime field $GF(p)$, and the field $GF(2^m)$, including four programs for point-counting. For more information on ECC see "Elliptic Curves in Cryptography", Blake, Seroussi & Smart, London Mathematical Society Lecture Notes Series 265, Cambridge University Press, ISBN 0 521 65374 6. This can be ordered directly from the publisher. Another good source of information is from the IEEE P1363 standards documents.

Elliptic Curves over $GF(p)$ and $GF(2^m)$ offer many advantages over standard methods such as RSA.

- An Elliptic Curve provides an ideal match for the AES (Advanced Encryption Standard) block encipherment (which is implemented within MIRACL). Using a 256 bit prime provides the same security as 128-bit AES. Similarly 384 bit ECC matches 192-bit AES, and 512 bit ECC matches 256-bit AES. Note that to offer the same level of security as 512-bit ECC would require the use of
Schoof / SEA

- schoof - Schoof’s Algorithm for prime fields
- schoof2 - Schoof’s Algorithm for fields of characteristic 2
- SEA - Fast SEA Algorithm for prime fields
Pairing-friendly Curves

- Embedding degree
  - A pairing-friendly curve needs small $k$.
- MNT
  - $k=3, 4, 6$
- Freeman
  - $k=10$
- Cocks-Pinch
  - $k=2, 4, 8$
- Brezing-Weng
  - Bandw - Generate the polynomials
  - $bn$ – Generate curves for ake12t.cpp and ake12.cpp
Complex Multiplication

- cm
  - Given p
  - Optional conditions
  - D cannot be larger

- Demo
Pairings

- pairings.txt - examples from miracl
- ake2sst
- ake2cpt
- ake2nsst
- BLS signature
A Variety of ECC Libraries

- **libecc**

### libecc

<table>
<thead>
<tr>
<th>LINKS</th>
<th>NEWS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Page</td>
<td>2 May 2008</td>
</tr>
<tr>
<td>Reference Manual</td>
<td><strong>libecc 0.12.1</strong> has been released. I extended the chapter &quot;Cracking parameter b of the elliptic curve&quot; dramatically. A recursive formula is presented for the number of solutions to ( y^2 + xy = x^3 + 1 ) over ( \mathbb{F}_2^m ) as function of ( m ). This journey is by far not finished, but I thought it was a milestone enough to warrant another release (and update of this chapter).</td>
</tr>
<tr>
<td>Download</td>
<td>9 April 2008</td>
</tr>
<tr>
<td>List of trinomials</td>
<td><strong>libecc 0.12.0</strong> has been released. This is a work of years... Especially since I'm not working on it, for years on a row. However, the fans of libecc will be happy-- I said, they will be HAPPY-- to know that this release adds support for x86_64! I think it's roughly twice as fast as on 32-bit x86 with assembly.</td>
</tr>
<tr>
<td>Mailinglists</td>
<td>Moreover, <strong>Cracking parameter b</strong> has been extended! So, more exciting reading ahead! Unfortunately, it isn't finished; but what I had has been laying around a year, so I thought I'd put it on the net as-is before I continue with it.</td>
</tr>
</tbody>
</table>

| DESCRIPTION | |
| Libecc is an elliptic curve crypto library for C++ developers. It is currently in development. | |

---

Yung-Hsiang Liu
Libraries

ECC 2009
12 / 18
A Variety of ECC Libraries

- **ECC-LIB**

**ECC-LIB**: A Library for Elliptic Curve Cryptography

ECC-LIB is a fully-equipped, portable, and modular library for Elliptic Curve (EC) Cryptography that allows for the easy development of EC cryptographic protocols, and which can be readily tailored to suit different requirements and user needs. The library is implemented in ANSI C using the GNU Multiple Precision library for high-precision floating point arithmetic. The full source code is included, written entirely in standard, portable ANSI C.

- [Version 2.0 of the ECC-LIB](http://www.ceid.upatras.gr/faculty/zaro/software/ecc-lib/)
- [Description and Installation Instructions](http://www.ceid.upatras.gr/faculty/zaro/software/ecc-lib/)

**Copyright**: There is a copyright on most of the implementations (see the corresponding files), but permission is granted to use, copy, modify and distribute this software for non-commercial purpose without fee, provided that the authors’ names are still included. All software in this package is provided and distributed under the [General Public License (GPL)](http://www.gnu.org/licenses/gpl.html).

- [An Example of Using the ECC-LIB](http://www.ceid.upatras.gr/faculty/zaro/software/ecc-lib/)
- [Related Papers](http://www.ceid.upatras.gr/faculty/zaro/software/ecc-lib/)

  - E. Konstantinou, A. Kontogorgis, Y. Stamatiou, and C. Zaroliagis
    "Generating Prime Order Elliptic Curves: Difficulties and Efficiency Considerations"

  - E. Konstantinou, Y. Stamatiou, and C. Zaroliagis
    "On the Use of Weber Polynomials in Elliptic Curve Cryptography"
    in *Public Key Infrastructure - EuroPKI 2004*
A Variety of ECC Libraries

- **PBC Library**

---

**About**

Pairing-based cryptography is a relatively young area of cryptography that revolves around a certain function with special properties.

The PBC (Pairing-Based Cryptography) library is a free C library (released under the [GNU Public License](https://gnu.org/licenses/gpl.html)) built on the [GMP library](http://gmplib.org/) that performs the mathematical operations underlying pairing-based cryptosystems.

The PBC library is designed to be the backbone of implementations of pairing-based cryptosystems, thus speed and portability are important goals. It provides routines such as elliptic curve generation, elliptic curve arithmetic and pairing computation. Thanks to the GMP library, despite being written in C, *pairings times are reasonable*. On a 1GHz Pentium III:

- Fastest pairing: 11ms
- Short pairing: 31ms

The API is abstract enough that the PBC library can be used even if the programmer possesses only an elementary understanding of pairings. There is no need to learn about elliptic curves or much of number theory. (The minimum requirement is some knowledge of cyclic groups and properties of the pairing.)

This [tutorial](http://crypto.stanford.edu/pbc/tutorials/index.html) shows how to implement a pairing-based cryptosystem in a few lines using the PBC library.

The PBC library can also be used to build conventional cryptosystems.
A Variety of ECC Libraries

- LiDIA
  - http://www.cdc.informatik.tu-darmstadt.de/TI/LiDIA/

LiDIA
A C++ Library For Computational Number Theory

MAIN PAGE

- News
- Introduction
- Online Documentation
- System Requirements
- License
- Download
- Mailing Lists
- Support, Bug Reports

Darmstadt University of Technology
A Variety of ECC Libraries

- **Crypto++**
  - [http://www.cryptopp.com/](http://www.cryptopp.com/)

### What is it?

Crypto++ Library is a free C++ class library of cryptographic schemes. Currently the library contains the following algorithms:

<table>
<thead>
<tr>
<th>Algorithm Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticated encryption schemes</td>
<td>GCM, CCM, EAX</td>
</tr>
<tr>
<td>high speed stream ciphers</td>
<td>Panama, Sosemanuk, Salsa20, XSalsa20</td>
</tr>
<tr>
<td>AES and AES candidates</td>
<td>AES (Rijndael), RC6, MARS, Twofish, Serpent, CAST-256</td>
</tr>
<tr>
<td>other block ciphers</td>
<td>IDEA, Triple-DES (DES-EDE2 and DES-EDE3), Camellia, SEED, RC5, Blowfish, TEA, XTEA, Skipjack, SHACAL-2</td>
</tr>
<tr>
<td>block cipher modes of operation</td>
<td>ECB, CBC, CBC ciphertext stealing (CTS), CFB, OFB, counter mode (CTR)</td>
</tr>
<tr>
<td>message authentication codes</td>
<td>VMAC, HMAC, CMAC, CBC-MAC, DMAC, Two-Track-MAC</td>
</tr>
<tr>
<td>hash functions</td>
<td>SHA-1, SHA-2 (SHA-224, SHA-256, SHA-384, and SHA-512), Tiger, WHIRLPOOL, RIPEMD-128, RIPEMD-256, RIPEMD-160, RIPEMD-320</td>
</tr>
<tr>
<td>public-key cryptography</td>
<td>RSA, DSA, ElGamal, Nyberg-Rueppel (NR), Rabin-Williams (RW), LUC, LUCELG, DLIES (variants of DHAES), EIGN</td>
</tr>
<tr>
<td>padding schemes for public-key systems</td>
<td>PKCS#1 v2.0, OAEP, PSS, PSSR, IEEE P1363 EMSA2 and EMSA5</td>
</tr>
<tr>
<td>key agreement schemes</td>
<td>Diffie-Hellman (DH), Unified Diffie-Hellman (DH2), Menezes-Qu-Vanstone (MQV), LUCDIF, XTR-DH</td>
</tr>
<tr>
<td>elliptic curve cryptography</td>
<td>ECDSA, ECNR, ECIES, ECDH, ECMQV</td>
</tr>
<tr>
<td>insecure or obsolescent algorithms</td>
<td>MD2, MD4, MD5, Panama Hash, DES, ARC4, SEAL 3.0, WAKE, WAKE-OFB, DESX (DES-XEX3), RC2, SAFER, 3-WAY, GOST, SHARK, CAST-128, Square</td>
</tr>
<tr>
<td>backwards compatibility and historical value</td>
<td></td>
</tr>
</tbody>
</table>
ECC and Bilinear Pairing Software Library

http://www.cs.cityu.edu.hk/~ecc/software/lib_about.htm

About

The ECC and Bilinear Pairing Software Library is a portable library written in C designed to offer functions related to ECC and Bilinear Pairing.

One can do the primitive operations in the underlying fields, ecc operations, and pairing functions with the help of the library.

The library can also be used to build practical cryptosystems.

Examples

In the Crypto Systems directory, there are demonstrations of many cryptosystems built with the library, although there are still works to be done before turning them into practical applications. Examples include:
Homework


- 20%
  - Run a program, and find an elliptic curve suitable for cryptography use.
  - Introduce the curve, and sell the curve in your report, it might include (but not necessary)
    - The security strength.
    - The way to use.
    - Any other you like.
  - Hand in via E-mail: randy.cs94g@nctu.edu.tw