Fundamental Elliptic Curve Cryptography Algorithms

draft-mcgrew-fundamental-ecc-02

mcgrew@cisco.com

kmigoe@nsa.gov
Elliptic Curve Cryptography

• Alternative to integer-based Key Exchange and Signature algorithms
• Smaller keys and signatures
• More efficient at higher security levels
Diffie Hellman

$g$ is number $< p$

Alice

$x = \text{random}$

$g^x \mod p$

Bob

$y = \text{random}$

$g^y \mod p$

$(g^y)^x \mod p = (g^x)^y \mod p$
EC Diffie Hellman

$g$ is element of EC group $G$

Alice

$x = \text{random}$

Bob

$y = \text{random}$

\[
(g^y)^x = (g^x)^y
\]
Cryptographic Groups

Prime Group
Element is number $x < p$

- Prime modulus $p$
- Generator $g < p$
- Order $n$

EC Group
Element is $(x, y)$ with $x, y < p$
with $y^2 = x^3 + ax + b \mod p$

- Prime modulus $p$
- Parameters $a, b < p$
- Generator $(g_x, g_y)$
- Order $n$

ECC Parameter Set
From RFC3766, *Determining Strengths For Public Keys Used For Exchanging Symmetric Keys*
ECC Efficient at High Security

Computational Cost vs. Security

- Integer
- ECC
fECC

• draft-mcgrew-fundamental-ecc
  – Informational
  – First published 7/09
  – Comments received and incorporated in -02
• Closely based on pre-1994 references
  – Security: survived > 16 years of review
  – IPR: simplifies analysis
Timeline

1985
ECC invented [M1985]

1986

1987
EC ElGamal [K1987]

1988
ECC ElGamal Signatures [A1992]

1989
ECC Implementation [BC1989]

1990
Homogeneous Coordinates [KMOV1991]

1991

1992
Meta ElGamal Signatures [HMP1994]

1993
Abbreviated EC ElGamal Signatures [KT1994]
Layers

- Crypto Algorithms
  - Key Exchange, Signatures
- Elliptic Curve Arithmetic
  - Coordinates, Representation
- Modular Arithmetic
  - +, -, *, /
fECC Diffie-Hellman

- Miller 1985
- Compatible with IKE (RFC 4753)
- Compatible with ECDH (IEEE 1363, ANSI X9.62)
  - Curves over $\text{GF}(p)$ with cofactor=1
  - ECSVDP-DH primitive
  - Key Derivation Function is identity function
fECC Signatures

- Koyama and Tsuruoka, 1994
- Horster, Michels, and Petersen, 1994
- KT-IV Signatures
  - Compatible with ECDSA (IEEE 1363, ANSI X9.62)
- KT-I Signatures
  - Not interoperable with standard
ECC Parameter Sets

• Compatible
  – Suite B
    • USG Cryptographic Interoperability Strategy
    • Uses NIST P256, P384, P521
  – Other NIST curves over GF(p)
  – RFC 5639 *Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation*
  – WAPI ISO/IEC JTC 1/SC 6 Proposal

• Not compatible
  – DJB’s Curve25519 protocol
Not in Scope

- EC Group Parameter Generation
- Identity-based crypto
- Edwards’ coordinates
- $\text{GF}(2^m)$ curves
- Mod $p$ arithmetic optimizations
- Certificate details
- Exotic groups (hyperelliptic, braids, ...)
- ...
Possible Future Drafts

• Optimizations
  – Modular arithmetic
    • Efficient primes
  – Elliptic Curve arithmetic

Priority: preserve interoperability and compatibility with standards
Conclusions

• Draft ready for RFC
• ECC deserves serious consideration
  – fECC is secure and performs well
• Recommendation: IETF work using ECC should explicitly allow fECC
  – … implementations MAY use [fECC] …
Questions?
\((x_3, y_3) = (x_1, y_1) \times (x_2, y_2)\)

\[x_3 = ((y_2-y_1)/(x_2-x_1))^2 - x_1 - x_2\]

\[y_3 = (x_1-x_3)(y_2-y_1)/(x_2-x_1) - y_1\]
A Group

<table>
<thead>
<tr>
<th>$\times$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

$5, \ 5^2=4, \ 5^3=6, \ 5^4=2, \ 5^5=3, \ 5^6=1$

Multiplication modulo 7