Alternative
Elliptic Curve Representations

draft-ietf-lwig-curve-representations-00

René Struik

Struik Security Consultancy

E-mail: rstruik.ext@gmail.com

IETF 103 – Bangkok, Thailand, November 7, 2018

Background

History:

– Initial document presented on March 21, 2018 @ IETF-101
https://datatracker.ietf.org/meeting/101/materials/slides-101-lwig-4-lwig-curve-representations-01

– Adopted as WG doc after IETF-102 meeting Montreal, July 2018

Background:

– NIST curves and CFRG curves use different curve models, thereby *seemingly* precluding code reuse

– Draft shows how curve models are related, by showing how one can switch between curve models via alternative representations

– Draft illustrates how to *reuse existing code* for NIST prime curves to implement CFRG curves (e.g., combine P-256 curve + Curve25519)

– Draft also illustrates how to use this to *reuse existing standards*

– Draft illustrates how to implement Edwards curve via Montgomery ladder, thereby allowing also code reuse amongst just CFRG curves

Current Status (1)

What was in pre-WG version 02?

– Pre-WG draft showed how to reuse *generic* existing ECC code

– Pre-WG draft also showed how to reuse *non-generic* existing implementations, including those that hardcode domain parameter a=-3 with short Weierstrass curves (which NIST*p* and Brainpool do)

– Pre-WG draft still lacked some fine details, since hard to compute

What is new in WG version 00?

– WG draft now provides full details of curve models and mappings, thereby allowing implementation of Curve25519 and Ed25519 with existing short-Weierstrass curve code, whether *generic*, *optimized*, or “Jacobian-friendly” (with hardcoded a=-3 domain parameter)

Current Status (2)

What has been added in WG version 01? (post submission cut-off)

– Some suggestions, e.g., by Nikolas Rösener, Phillip Hallam-Baker

– Incorporates worked-out examples:

- ♦ Implementations:
  - – co-factor Diffie-Hellman (X25519) via Weierstrass curve;
  - – EdDSA signing via Montgomery ladder for Curve25519;
- ♦ Specifications:
  - – reuse NIST SP 800-56a to specify ephemeral key pairs for CFRG curves (e.g., §4.2.2 of draft-selander-ace-cose-ecdhe-10)

Implementations:

[1] N. Rösener, *Evaluating the Performance of Transformations Between Curve Representations in Elliptic Curve Cryptography for Constrained Device Security*, M.Sc., Universität Bremen, August 2018.

[2] H. Liu, "How to Use the Kinets LTC ECC HW to Accelerate Curve25519 (v.7)," NXP, April 27, 2017. See https://community.nxp.com/docs/DOC-330199 **(mentions 10x speed-up with *existing* ECC HW)**

Next Steps?

Main features latest draft:

– Shows how to implement CFRG curves using existing NIST*p* code

– Shows how to implement Edwards curve using Montgomery ladder (thereby, allowing code reuse for different CFRG curve models, [even if one does not care about short-Weierstrass curves])

Do we need more?

– *More feedback on latest draft welcome*!

– Conversions can be implemented using a few field additions and multiplies. Do worked-out examples provide sufficient details?

Question:

– Are there any other ECC implementation mysteries to be disspelled? (and, if so, should this be in this draft or elsewhere?)