

Alternative Elliptic Curve Representations

draft-struik-lwig-curve-representations-01

René Struik

Struik Security Consultancy

E-mail: rstruik.ext@gmail.com

Status

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>

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 P256 curve + Curve25519)
- Draft also illustrates how to use this to *reuse existing standards*

Status

What is new in version 01?

- Old draft showed how to reuse *generic* existing ECC code
- New draft shows how this also works for *non-generic* existing implementations:
 - ◆ implementation that hardcodes specific domain parameters (e.g., code uses Jacobian coordinates and hardcodes $a=-3$)
 - ◆ implementation that allows speed-up if domain parm a is small (draft shows how to end up with short-Weierstrass curve with domain parameter $a=2$ [thereby, improving speed])

What's next?

- Draft still needs detailed mappings for short-Weierstrass curve with $a=-3$ (once computations finished) [NOTE: *this is one para...*]

Implementation:

- Being implemented by Nikolas Rösener (Bremen University)

Next Steps?

Questions:

- Is this useful to LWIG?
- Should we make this a WG draft (intended status: informational)?
- Are there any other ECC implementation mysteries to be dispelled?

Conclusions

1. Different curve models can be implemented using the same code if one uses the short-Weierstrass model.
2. One can thereby reuse not just code, but also existing standards, thus significantly reducing standards development cycles.
3. Encoding format issues may negatively impact code reuse and reuse of existing standards, since can be used as artificial “moat around a solution”, making code reuse or algorithm agility economically less viable than these could/should be.
4. Representation conventions require more careful considerations by IETF in the future than has happened so far (in TLS1.3, CFRG).