Trustless Construction of Groups of Unknown Order with Hyperelliptic Curves

Samuel Dobson, Steven D. Galbraith, and Benjamin Smith

June 20, 2020

Interest in groups of unknown order has been fuelled in recent years by applications such as delay functions [Wes19], accumulators [BBF19], and zero-knowledge arguments of knowledge [BFS19]. Previously, there were two proposals for groups of unknown order:

1. RSA groups $(\mathbb{Z}/pq\mathbb{Z})^*$ [RSW96] - requires **trusted** setup
2. Ideal class groups of imaginary quadratic fields [BW88] - **trustless**

Many interesting protocols [BBF19, BH01] *require* trustless setup, so the latter is especially interesting. We propose a third construction for trustless groups of unknown order:

3. Jacobian groups of genus 3 hyperelliptic curves - **trustless**

We also re-analyse suggested parameters for secure groups of unknown order, compare efficiencies of class groups and Jacobians, and thirdly provide a method of compressing elements of ideal class groups.

Table 1: Summary of constructions' security, parameter sizes and storage sizes

Construction	Parameters	Security	Element size (kilobytes) uncompressed	Element size (kilobytes) compressed
Class groups	$\Delta \sim 1665$-bit	55-bit (previously thought to be 128-bit)	0.21 kB	0.16 kB
Class groups	$\Delta \sim 6656$-bit	128-bit	0.83 kB	0.62 kB
Genus 3 Jacobians	$q \sim 1100$-bit	128-bit	0.83 kB	0.41 kB

Secure group orders

We first discuss the security of ideal (equivalently, binary quadratic form) class groups of unknown order. Buchmann and Hamdy [BH01] suggest the use of a 1665-bit (negative prime) discriminant for 128-bit security. This estimate is used by Boneh et al. [BBF19, BFS19] in their instantiations of various protocols.

Sutherland presents a generic algorithm in his thesis [Sut07, Algorithm 4.2], which he names the Primorial-Steps algorithm, for finding group element orders. Let G be the semismoothness probability function. Let the order of a group element be uniformly distributed in $[1, M]$. Sutherland's algorithm [Sut07, Proposition 4.7] computes the order in time and space $O(M^{1/u})$ with probability $P \geq G(1/u, 2/u)$. The algorithm is designed for groups whose order is a random integer, which is essentially the case with the groups of unknown order we study here. Intrinsically, we cannot check if a randomly generated group is vulnerable to this algorithm. Suppose a group is vulnerable with probability p and, if vulnerable, there is an attack in time T. Then, we claim that the security level of the system is at most $\max\{T, 1/p\}$.

Using previously calculated values of $G(1/u, 2/u)$ [BP96] we show that $\Delta \sim 1665$-bits only provides 55-bit security by the above definition. For 128-bit security, we tentatively extrapolated the trend of u vs $G(1/u, 2/u)$, and claim that a group order of at least $M = 2^{128*26} = 2^{3328}$ is required. Because Sutherland's algorithm is generic, this applies to *all* trustless groups of unknown order, regardless of construction. In the case of ideal/form class groups, this requires $\Delta \sim 6656$-bits, which is much larger than desirable.

Compression of class group elements

An element of an ideal/form class group of discriminant Δ corresponds to a triple of integers (a, b, c) such that $b^2 - 4ac = \Delta$. Since Δ is a fixed and known constant, it suffices to store the pair (a, b). A reduced quadratic form satisfies $|b| \leq a < \sqrt{|\Delta|}$. It follows that the pair (a, b) can be represented in approximately $\log_2(|\Delta|)$ bits.

This can be compressed using a similar method as that of Bleichenbacher [Ble04] for Rabin signatures. Observe that $b^2 \equiv \Delta \pmod{a}$. The continued fraction algorithm (i.e., Euclid's algorithm) computes integers s, t such that $b \equiv s/t \pmod{a}$ and $|s|, |t| \leq \sqrt{a}$. Publishing (a, t) allows recovery of s and thus b (up to sign). Since $|t| < \sqrt{a}$ we have a representation for the ideal class that uses approximately $\frac{3}{2}\log_2(\sqrt{|\Delta|})$ bits, which is 3/4 the size of the standard representation. Some extra care is needed to ensure that decompression is possible in all cases, but we work through these details in the full paper and have a short PoC implementation of the code in Python.

Hyperelliptic curves

We suggest that hyperelliptic curves of genus 3 can be used as a more efficient alternative to the ideal/form class group in situations where a group of unknown order is required. To construct a group of unknown order, simply take a random (nothing-up-my-sleeve) curve over a sufficiently large finite field. If we have a hyperelliptic curve $y^2 = f(x)$ ($\deg f = 2g+1$) of genus g over the finite field of cardinality q, the $\mathbb{F}_q$-rational points of the Jacobian form a finite group of order bounded by $(\sqrt{q} - 1)^{2g} \leq \#\mathrm{Jac}(C) \leq (\sqrt{q} + 1)^{2g}$, by the Hasse-Weil bound. Thus to achieve the required 3300-bit group order above with a genus 3 curve, we should choose $q \sim 1100$-bit.

Elements of the group are given uniquely in Mumford representation. Hess, Seroussi, and Smart [HSS01] gave a method to compress elements in genus g that requires g field elements and g extra bits, which is more efficient (essentially optimal) than the proposed compression of class group elements above. Hence Jacobian elements are more compactly represented for the same level of security (at 128-bit security, this uses 0.41kB vs 0.62kB for class groups, as shown in Table 1).

We review known point counting and discrete logarithm solving algorithms for hyperelliptic curves in the paper. We claim that all existing algorithms for order computation are infeasible at the group sizes recommended above, including the current state of a "polynomial-time" Schoof-Pila type algorithm - whose hidden constants make such an algorithm impractical in genus 3. There are some unusual properties of hyperelliptic curves which class groups do not have, such as division ideals, which we discuss in the paper.

References

[BBF19] Dan Boneh, Benedikt Bünz, and Ben Fisch. Batching techniques for accumulators with applications to IOPs and stateless blockchains. In *Advances in Cryptology – CRYPTO 2019*, pages 561–586, 2019.

[BFS19] Benedikt Bünz, Ben Fisch, and Alan Szepieniec. Transparent SNARKs from DARK compilers. Cryptology ePrint Archive, Report 2019/1229, 2019. `https://eprint.iacr.org/2019/1229`.

[BH01] Johannes Buchmann and Safuat Hamdy. A survey on IQ cryptography. In *In Proceedings of Public Key Cryptography and Computational Number Theory*, pages 1–15, 2001.

[Ble04] Daniel Bleichenbacher. Compressing Rabin signatures. In *CT-RSA 2004*, volume 2964 of *LNCS*, pages 126–128, 2004.

[BP96] Eric Bach and René Peralta. Asymptotic semismoothness probabilities. *Mathematics of computation*, 65(216):1701–1715, 1996.

[BW88] Johannes Buchmann and H. C. Williams. A key-exchange system based on imaginary quadratic fields. *Journal of Cryptology*, 1(2):107–118, Jun 1988.

[HSS01] Florian Hess, Gadiel Seroussi, and Nigel P. Smart. Two topics in hyperelliptic cryptography. In *International Workshop on Selected Areas in Cryptography*, pages 181–189, 2001.

[RSW96] Ronald L. Rivest, Adi Shamir, and David A. Wagner. Time-lock puzzles and timed-release crypto. 1996.

[Sut07] Andrew V Sutherland. *Order computations in generic groups*. PhD thesis, Massachusetts Institute of Technology, 2007.

[Wes19] Benjamin Wesolowski. Efficient verifiable delay functions. In *Advances in Cryptology – EUROCRYPT 2019*, pages 379–407, 2019.