August 11, 2025

My Personal Journey with ECC

Neal Koblitz
Dept. of Mathematics
Univ. of Washington, Seattle

My plan of becoming a mathematician was largely set in stone in 1956 at the age of 7, when I returned to the U.S. from India, where I’d spent the “second standard” (2nd grade in American terminology) in a local school that had a more advanced math curriculum than my school back home.

When I entered the 3rd grade, my American teacher was impressed with my facility at multi-digit arithmetic, and she encouraged me.

For me, math was the logical choice — it was the one area, and in fact the only area, where I could do better than my peers. The process of elimination led me to mathematics.

By time I was an undergrad (1965-69), I had settled on my main area of interest in mathematics, namely, number theory and related fields of pure math, such as arithmetic algebraic geometry.

These were all areas in pure, abstract mathematics. I was unfazed by the thought that I would spend my career doing research that was totally useless, barren of any practical applications.

I knew what the famous British number theorist G. H. Hardy had written about number theory in his 1940 book *A Mathematician's Apology*:
"...both Gauss and lesser mathematicians may be justified in rejoicing that there is one science at any rate, and that their own [number theory], whose very remoteness from ordinary human activities should keep it gentle and clean."

G. H. Hardy

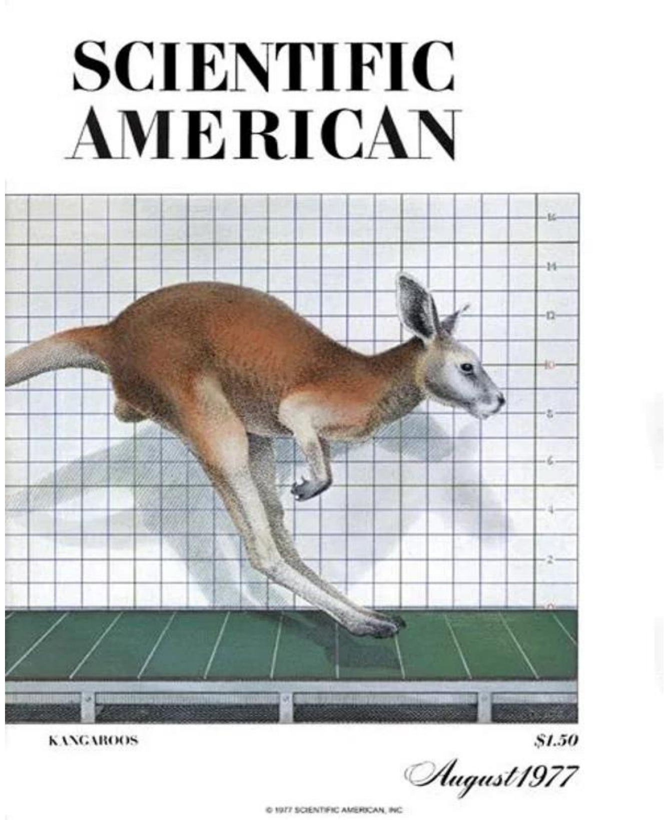
By “gentle and clean,” G. H. Hardy presumably meant “free of applications,” which in his day largely consisted of military applications. Hardy, a pacifist, was expressing a view that seemed reasonable to me 3 decades later. My student years coincided with the height of the genocidal U.S. War in Vietnam, which resulted in the deaths of an estimated 3 million Vietnamese.

(Efforts were made in the Amer. Math. Society to get the AMS to urge mathematicians not to work on military applications, but those efforts failed.)

My 1974 PhD thesis was a study of the *p*-adic properties of the Hasse-Weil zeta function of an elliptic curve, hyperelliptic curve, or more general curve or abelian variety defined over a finite field $\boldsymbol{\mathcal{F}}_q$. That zeta-function codifies the numbers of points on the curve or variety over the extension fields of q^n elements.

Given my thesis topic, there was still no reason to doubt Hardy’s words. The military was not going to come knocking on my door.

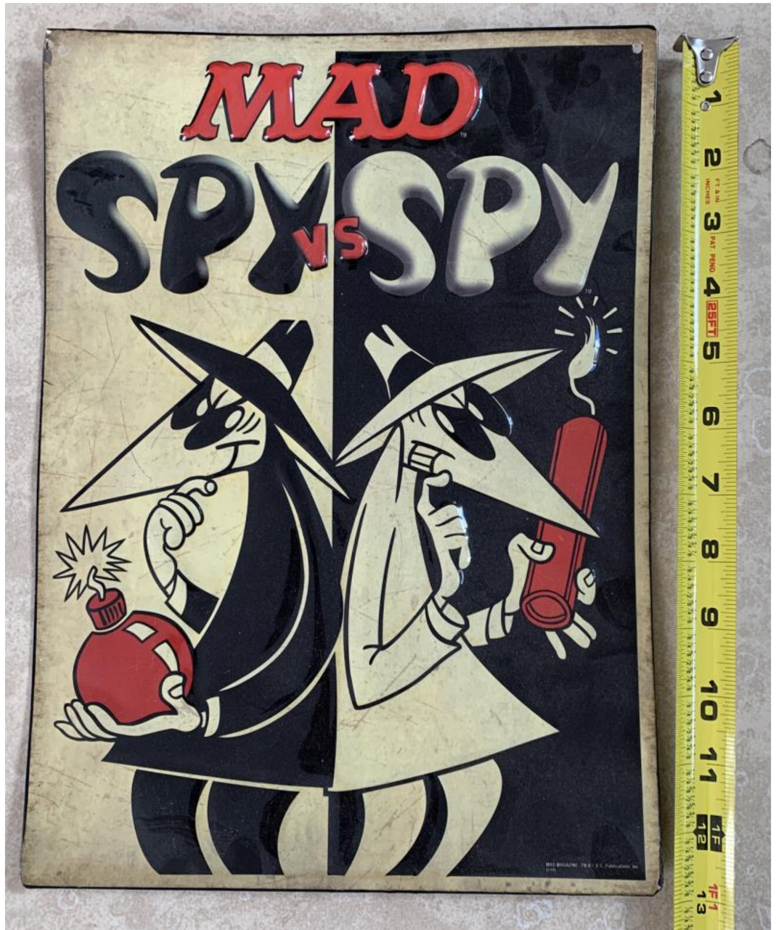
My interest in cryptography started shortly after I moved to the University of Washington in 1979 for my first (and last) tenure-track job. Although I had read and (like everyone else) been intrigued by Martin Gardner’s 1977 article about RSA in *Scientific American*, it was teaching, not research, that caused me to want to learn a little cryptography.



Martin Gardner’s article was in the Mathematical Games section of the August 1977 issue.

One of the courses I taught was our introduction to number theory. It had traditionally been taught as an introduction to notions of rigorous proof, an approach that I thought most students would find boring and pointless.

Almost none of our students intended to become theorem-proving pure mathematicians. After reading Martin Gardner’s article, I was convinced that focusing the course on the mathematics of RSA — that is, connecting number theory to the spy-vs-spy world of cryptography — would be a much better way to motivate students.



At about the same time I taught a graduate course on elliptic curves and modular forms, and turned my lecture notes for the course into a textbook that was published in 1984.

In December of that year Hendrik Lenstra sent me a 2-page outline of his new elliptic curve factoring algorithm. His algorithm was easy to read and understand — and it hit me like a thunderbolt. Lenstra’s algorithm suddenly escalated the level of sophistication of the mathematics that could be applied to cryptography.

Hendrik Lenstra, Jr.

Inevitably I started wondering if the rich structure of the objects of arithmetic algebraic geometry — such as the group laws on elliptic curves and on the Jacobians of higher-genus curves — could be useful not only for attacking RSA as Lenstra had done, but also for constructing secure cryptographic primitives.

The next month, in January of 1985, I went to Moscow to participate in the exchange program between the Academies of Sciences of the U.S. and the Soviet Union.

Unlike in earlier visits to Moscow, when I took part in Yuri Manin's research group, I worked alone. In the first place, Manin and his students had largely changed their field of interest from number theory to mathematical physics.

Yuri I. Manin

My interests had also shifted — to cryptography, which was a field in which, as far as I was aware, nobody did open research in the Soviet Union.

The Soviet authorities would have been no more enthusiastic about open academic research on cryptography than was the NSA in the U.S.

I soon saw that the Diffie-Hellman key exchange, based on the multiplicative group of a finite field, could easily be carried over to an elliptic curve group; and I convinced myself that the index-calculus algorithms that provide the most efficient attack on the original Diffie-Hellman protocol did not carry over in any obvious way to the elliptic curve group.

As a newcomer to crypto, I was not very confident about all this, so I wrote a letter to Andrew Odlyzko describing the idea and asking him what he thought.

Andrew Odlyzko

The turnaround for letters between the U.S. and the U.S.S.R. averaged a month or more. Finally, I heard from Odlyzko that he thought it was a good approach, and that Victor Miller had independently come up with the same idea.

The high point of my semester in Moscow was my invited talk at the Moscow Math Society. My friend Tolya Fomenko set up the invitation, largely because he thought it would be cool to have the first lecture about cryptography ever given at the Moscow Math Society given by an American.

Some of the most famous Soviet mathematicians — Arnol'd, Gel'fand, Shafarevich, Manin, and others — would attend, and I, a newcomer to cryptography, was very nervous about making a bad impression.

What saved me was that most Soviet mathematicians knew even less about cryptography than I did. I got good feedback after the lecture; it seems that much of what I described was new to them.

But I didn’t mention ECC, since I had not yet heard from Odlyzko.

Back in the U.S., I started attending crypto conferences, especially the August Crypto conference in Santa Barbara.

I liked the interdisciplinary nature of it. For the first time I was learning from people whose background was in computer science or engineering, rather than mathematics, and who worked in industry or government research teams rather than in academia.

I met some of the famous names in crypto, such as the RSA trio — Ron Rivest, Adi Shamir, and Len Adleman — and Whit Diffie.

Diffie had a special role at Crypto running the rump sessions, which got wild at times with audience participation and heckling. The rump sessions became a little less rowdy after restrictions were introduced on what you were permitted to throw at speakers if you didn’t like what they were saying.

Whit Diffie

In those days Crypto was a lively, free-spirited event, perhaps partly because there was an aspect of “forbitten fruit” in the very existence of the conference.

Just a few years before, the NSA had made a heavy-handed attempt to get the power of prior restraint to prevent open publication of cryptographic research. They wanted to preview papers in number theory, for example, and block publication when they contained results of cryptographic significance.

Fortunately, that attempt at suppressing open research was derailed.

But it led to great mistrust of the NSA, along with a feeling that having open meetings like Crypto was an act of defiance.

The atmosphere at Crypto changed in the 1990s, as the field got more professionalized, with intense competition to get large numbers of papers published and get contributed talks for oneself and one's students accepted at the major conferences. Crypto started to resemble the major computer science conferences, such as FOCS and STOC.

In addition, in the 1990s other conference series were started with a narrower focus on a subfield — for example, the ECC conference series. Gradually, many of the mathematicians, including me, stopped attending Crypto and became more interested in some of the conferences that had more mathematical content.

Mathematicians can sometimes be very naïve about practical matters. In my case this led to some serious misjudgments in my early crypto papers.

First, for pedagogical reasons I thought it would be nice to illustrate ECC using a curve for which the group order was easy to calculate — the example of the curve $y^2 = x^3 - x$ over the field of p elements, with $p \equiv 3 \pmod 4$ has order $p+1$, as a simple exercise will show, with no need for Schoof's algorithm.

At that time I thought that if $p+1$ has a large prime factor, say, if $(p+1)/4$ is prime, then we'd have a nice, self-contained description of a secure choice of parameters.

But soon thereafter, Menezes, Okamoto, and Vanstone (MOV) showed how to imbed that elliptic curve group into the subgroup of elements of norm 1 in the multiplicative group of the field of p^2 elements.

This MOV-attack was made possible by the same special structure that allowed us to make a quick count of the group order.

This reduces ECC for this curve to a traditional Diffie-Hellman protocol in a finite field. So my “nice” ECC example does not illustrate ECC at all.

A second example of flawed reasoning in my early papers occurred when I proposed using the Jacobian groups of hyperelliptic curves of genus $g > 1$ (elliptic curves are of genus g=1).

I gave examples of curves of high genus g over a small prime field F_p whose simple equation led to an easy computation of the group order, which has magnitude p^g.

I further suggested that the Diffie-Hellman protocol might be even more secure when g is large, because high-genus curves have greater complexity than small-genus ones.

Well, “greater complexity” is true in a topological sense when one works over the complex numbers, where the compactification of a genus-g curve can be represented as a “donut” with 1 hole when g=1 and a surface with g holes when $g > 1$.

But topological complexity by no means implies computational complexity. Before long an attack on the DLP for a hyperelliptic Jacobian group was found that becomes much more efficient for large g.

I hope that no one took my suggestion of basing their crypto on hyperelliptic curves of high genus over a small prime field.

It was at Crypto that I met Scott Vanstone and his star graduate student at the time, Alfred Menezes. That was the start of a long and fruitful collaboration with them. In 1998-99 I spent a year at Waterloo, on leave from the University of Washington.

<— Scott Vanstone

Alfred Menezes—>

In the 1990s public key cryptography and computer security went from being a niche interest of a relatively small number of industry, government, and academic researchers to being a major concern in a computerized economy.

There was fierce competition and conflict, but it was not of the “Spy-vs-Spy” variety as in the *Mad Magazine* cartoons of the 1960s and 70s.

Rather, it was company-vs-company. The RSA company, in particular, started to feel threatened by the upstart company Certicom, the Canadian crypto company started by the Waterloo people working with Scott Vanstone. It looked like ECC, which could achieve similar security levels to RSA with shorter keys, would expand its market share, especially in recently created constrained environments, such as smart cards.

In 1996, when Alfred Menezes and I were jointly teaching a short course for grad students at MSRI in Berkeley, Alfred showed me the part of the RSA website called “ECC Central,” which featured quotations from leading cryptographers connected with RSA that expressed skepticism about ECC.

I was particularly impressed by Ron Rivest’s statement. He said that elliptic curves are a difficult and relatively obscure topic of study, and that “trying to get an evaluation of the security of an elliptic curve cryptosystem is a bit like trying to get an evaluation of some recently discovered Chaldean poetry.”

I didn’t know who the Chaldeans were, so I asked my wife Ann, who was trained as a historian, and of course she knew. They were a people living in Babylonia in Biblical times. I found that the U.C. Berkeley library had a volume of Chaldean poetry, and even in English translation it was readable and evocative.

It was customary at the end of the MSRI short courses to have souvenir T-shirts made with an appropriate slogan. I proposed that the swag have a picture of an elliptic curve with the words “I Love Chaldean Poetry.”

I also got some of the T-shirts, and am wearing one of them today.

Two years later, in early September 1998, I was visiting Ann in Phoenix (where she was a Professor of Women and Gender Studies at Arizona State University) before going to Waterloo for the year.

I received an urgent message that Scott Vanstone had called and wanted me to call back as soon as possible.

I was surprised by this, since I'd never before got a telephone message from Scott, and I hadn't even thought that he knew how to reach me in Phoenix.

It must be truly urgent if he couldn't wait until I arrived in Waterloo in a few days.

A few hours later I received an email from Joe Silverman at Brown University that included a new algorithm for attacking ECC, and I knew why Scott wanted to talk with me.

Joe Silverman

Joe called his algorithm “xedni”, which is “index” spelled backwards, because in some sense its steps are similar to those in index calculus but in the reverse order. Also, xedni uses elliptic curves defined over the rational numbers in an important way.

Joe was “agnostic” about whether xedni would be practical for curves over finite fields in the range used in cryptography. He wasn’t ready to go public with his algorithm. He was only giving a few people advance notice, and presumably hoped that we would help him estimate the running time of xedni for our curves.

I badly wanted to have something useful to say to my friends and "allies" in the RSA-vs-ECC conflict. I understood the reason for Scott's sense of urgency. Even though Joe had not yet informed the RSA people about xedni, they'd get wind of it before long and would then start proclaiming from the rooftops that ECC was broken, and that this shows the danger in replacing RSA with ECC in smartcards and other constrained environments.

I saw that a reasonably accurate estimate of xedni's running time would likely depend on the heuristics of the Birch and Swinnerton-Dyer Conjecture for elliptic curves defined over the rational numbers. And I had no idea how to estimate those heuristics.

I was stymied.

The night before my flight from Phoenix to Toronto, I had trouble sleeping, and returned to thinking about xedni. Suddenly it occurred to me that, even though I'd have nothing to say that could rule out practicality of xedni in finding discrete logs on elliptic curves, I could do something almost as good. I could show that a minor modification of xedni could be used to factor large integers.

So if ECC was broken, so was RSA! This would buy us some time. The RSA people would no longer be eager to advertise the possible consequences of xedni!

(The reason why xedni can be carried over from the DLP to the Integer Factorization Problem is very similar to the reason why Shor’s post-quantum algorithm, which is basically for finding discrete logs, carries over to factoring integers.)

When I got to Waterloo, we immediately formed a small group of grad students and faculty at the Centre for Applied Cryptographic Research that would study the practicality of xedni.

Within a couple of weeks I had a strong argument, based on the heights of points on an elliptic curve over the rationals, that xedni would be incredibly slow asymptotically, that is, as the size of the finite field of definition approaches infinity. I presented this to the group, and everyone was convinced.

However, in cryptography it’s a common mistake to be satisfied with asymptotic assurances. We really needed to test xedni on curves in the range used in crypto, and this turned out to be a much longer project than finding my argument for asymptotic impracticality.

Our group collaborated closely with Joe Silverman, who advised us on which elliptic curves to test and how best to incorporate the properties of elliptic curves defined over the rational numbers.

Finally, some time in the winter, we had sufficient evidence for curves in the range used in ECC. Already in that range xedni was extremely slow, even slower than exhaustive search algorithms. Joe conceded that xedni was “dead in the water.”

The xedni drama was over, and (for us) it had a happy ending.

A decade and a half later we went through another drama with ECC, and this one did not end happily..
In 2013 Wikileaks released NSA documents that had been obtained by Edward Snowden. For the cryptographic community the most shocking revelation was that the NSA had inserted a back door in the Dual Elliptic Curve Deterministic Random Bit Generator (Dual_EC_DRBG). That made the front page of the *New York Times*.

Then in December 2013 we learned that the NSA had basically bribed the RSA company $10M to install Dual_EC_DRBG as the default random number generator in their BSAFE computer security kit.

The back door danger was known to the designers of EC_DRBG, who urged users to publicly verify that the points P and Q in their algorithm were chosen independently of one another.

(Here a back door, such as the one put in by the NSA, would consist of choosing $Q = kP$ with k known only to the NSA.)

But NIST failed to carry out this verification, and a pseudorandom bit generator was standardized with a back door.

In an article in the IEEE journal *Security & Privacy*, Alfred and I discussed possible explanations for the radical change in 2015 in the NSA's stand on ECC — from strong support for ECC to a new policy urging everyone to abandon both RSA and ECC and transition to post-quantum crypto as rapidly as possible:

(with A. J. Menezes) A riddle wrapped in an enigma, IEEE Security & Privacy, Vol. 14, No. 6 (Nov.-Dec. 2016), 34-42; available at https://eprint.iacr.org/2015/1018.pdf

My own preferred explanation is #4:

4. The NSA wanted to get away from ECC as quickly as possible after the EC_DRBG scandal, since ECC & NSA together are associated in people's minds with deception and lack of trust.

But the rush to adopt lattice-based crypto has some dangers.

A critique of claims of "provable security" for LWE systems: (with P. Sarkar, S. Samajder, and S. Singha) Concrete analysis of approximate ideal-SIVP to decision ring-LWE reduction, Advances in Mathematics of Communications, Vol. 18, 2024, 1216-1258, available at https://eprint.iacr.org/2022/275

Others have also expressed doubt that lattice-based systems are ready for prime time: Dan Bernstein, MATZOV (the crypto arm of the Israeli Defense Forces).

Questions: 1. Why was NIST in such a hurry to approve and standardize lattice-based post-quantum crypto?

Was it pressure from the NSA, or some other reason?

2. Why not standardize and recommend a hybrid system, where an adversary needs to break both ECDLP and LWE in order to succeed?

Wouldn’t this be a prudent step, to allow for possible breaks in the standardized lattice-based systems during the years while ECC remains secure?

Thank you