A RIDDLE WRAPPED IN AN ENIGMA

NEAL KOBLITZ AND ALFRED J. MENEZES

Abstract. In August 2015 the U.S. National Security Agency (NSA) released a major policy statement on the need for post-quantum cryptography (PQC). This announcement will be a great stimulus to the development, standardization, and commercialization of new quantum-safe algorithms. However, certain peculiarities in the wording and timing of the statement have puzzled many people and given rise to much speculation concerning the NSA, elliptic curve cryptography (ECC), and quantum-safe cryptography. Our purpose is to attempt to evaluate some of the theories that have been proposed.

"It is a riddle wrapped in a mystery inside an enigma; but perhaps there is a key."

—Winston Churchill, 1939 (in reference to the Soviet Union)

1. Introduction

In August 2015, the U.S. government's National Security Agency (NSA) released a major policy statement [40] (see also [14]) on the need to develop standards for post-quantum cryptography (PQC). The NSA, like many others, believes that the time is right to make a major push to design public-key cryptographic protocols whose security depends on hard problems that cannot be solved efficiently by a quantum computer.[1] The NSA announcement will give a tremendous boost to efforts to develop, standardize, and commercialize quantum-safe cryptography. While standards for new post-quantum algorithms are several years away, in the immediate future the NSA is encouraging vendors to add quantum-resistance to existing protocols by means of conventional symmetric-key tools such as AES. Given the NSA's strong interest in PQC, the demand for quantum-safe cryptographic solutions by governments and industry will likely grow dramatically in the coming years.

Most of the NSA statement was unexceptionable. However, one passage was puzzling and unexpected:

> For those partners and vendors that have not yet made the transition to Suite B algorithms [38], we recommend not making a significant expenditure to do so at this point but

Date: 20 October 2015.

[1]For many years it has been known [50] that both the integer factorization problem, upon which RSA is based, and the elliptic curve discrete logarithm problem (ECDLP), upon which ECC is based, can be solved in polynomial time by a quantum computer.

> instead to prepare for the upcoming quantum resistant algorithm transition.... Unfortunately, the growth of elliptic curve use has bumped up against the fact of continued progress in the research on quantum computing, necessitating a re-evaluation of our cryptographic strategy.

The NSA seemed to be suggesting that practical quantum computers were coming so soon that people who had not yet upgraded from RSA to ECC should not bother to do so, and instead should save their money for the future upgrade to post-quantum protocols.

Shortly thereafter, the NSA released a revised version in response to numerous queries and requests for clarification. The new wording was even more explicit in its negative tone on the continuing use of ECC: "...elliptic curve cryptography is not the long term solution many once hoped it would be. Thus, we have been obligated to update our strategy." Although other parts of the statement assured the public that ECC was still recommended during the time before the advent of practical quantum computers, the overall impression was inescapable that the NSA was distancing itself from ECC.

In addition, people at the National Institute of Standards and Technology (NIST) and elsewhere have noticed that the NSA has not been taking an active part in discussions of new curves to replace the NIST curves that were recommended for ECC in 1999. The PQC announcement suggests that NSA has no interest in this topic because it now views ECC as only a stopgap solution. This caught many people by surprise, since it is widely believed that ECC will continue to be used extensively for at least another decade or two.

The purpose of this article is to attempt an evaluation of the various theories, speculations, and interpretations that have been proposed for this sudden change of course by the NSA. We emphasize that this is not an academic paper, and so on occasion we shall give unsourced facts and opinions in circumstances where our sources wish to remain anonymous.

2. History: The NSA and ECC

2.1. **The first decade of ECC (1985-1995).** In the late 1980s and early 1990s, as the Cold War came to an end and as networked computers started to play a major role in the economy, the NSA "came in from the cold" and began to devote resources to advising the private sector on cybersecurity. Almost from the beginning there were indications that, of the available public-key systems, the NSA preferred ECC. (For further commentary on this history, see [30].)

In the early 1990s NIST proposed a Digital Signature Algorithm (DSA) [21] that had been developed by NSA and closely resembled an earlier method invented by C. Schnorr. Although DSA is based on the discrete log problem in the multiplicative group of a finite field — not in an elliptic

curve group — it signaled a dissatisfaction with factorization-based systems within NSA (perhaps in part because of the high licensing fees for the patented RSA algorithm). In fact, shortly after DSA was approved for governmental use in 1994, the analogous ECDSA protocol using elliptic curves was developed.

Proponents of RSA bitterly opposed DSA, and they claimed that the NSA was promoting DSA because they had inserted a back door in it ("No Back Door" was the slogan of the anti-DSA campaign). However, they gave no evidence of a back door, and in the two decades since that time no one has found a way that a back door could be inserted in DSA (or ECDSA).

The first time NSA publicly and decisively gave support to ECC occurred at a meeting of the American National Standards Institute (ANSI) in December 1995. The backers of RSA at the meeting were casting doubt on the safety of ECC-based protocols; in the mid-1990s a page called "ECC Central" on the RSA website carried statements by leading personalities in cryptography that characterized ECC as untested and based on esoteric mathematics. The nontechnical industry representatives on the ANSI committee were impressed by the RSA argument. As the heated debate continued, the NSA representative left to make a phone call. When he returned, he announced that he was authorized to state that NSA believed that ECC had sufficient security to be used for secure communications among all U.S. government agencies, including the Federal Reserve. People were stunned. In those days the NSA representatives at standards meetings would sit quietly and hardly say a word. No one had expected such a direct and unambiguous statement from NSA. The ECC standards were approved.

2.2. **The second decade (1995-2005).** At Crypto '97 Jerry Solinas gave the first paper [52] ever presented publicly at a major cryptography conference by an NSA member. It contained a procedure he had developed for greatly improved efficiency of ECC using anomalous binary curves, which are the unique family of nonsupersingular curves defined over the field of two elements. In 2000 five curves of this family, each at a different security level, were included in the list of 15 "NIST curves" [22].

Some people regard those curves as risky because of their special nature — for example, they all have complex multiplication by the integers of the quadratic field $Q(\sqrt{-7})$, which has class number 1 (in fact, Solinas' efficient point-multiple algorithms were based on this fact). In addition, in [25, 55] it was shown that for such curves there is a $\sqrt{m}$ speedup in the Pollard-rho algorithm for the ECDLP (where m is the degree of the field extension). For these reasons it was thought that a more conservative choice would be random curves over either a binary field or a prime field.

The other ten NIST curves consist of a set of five randomly generated curves over binary fields and five over prime fields,[2] again each corresponding

[2]The binary fields all have order 2^m with m prime. In the remainder of this paper, we will only be concerned with elliptic curves over such binary fields and over prime fields.

to different security levels. Because of recent progress attacking the ECDLP on curves over binary fields using methods that were inspired by ideas of I. Semaev [49] (see [24]), some experts now have doubts about the long-term security of all elliptic curves over binary fields.[3] Thus, the most conservative choice of NIST curves is the family defined over a prime field. Those curves are denoted P-k, where k is the bitlength of the prime.

NSA's support for ECC became more obvious over the years. In 2003 it licensed 26 ECC-related patents from Certicom for US$25 million, and in 2005 it posted the paper "The Case for Elliptic Curve Cryptography" [37] on its website. This paper described RSA as a "first generation" public key technology that had been superceded by ECC: "Elliptic Curve Cryptography provides greater security and more efficient performance than the first generation public key techniques (RSA and Diffie-Hellman [in the multiplicative group of a prime field]) now in use."

In conjunction with this recommendation, on 16 February 2005 at the RSA Conference the NSA announced its Suite B recommended algorithms [38]. Ironically, in its original form it included no RSA (or DSA) protocols, but only ECC (ECDSA for signatures, ECDH and ECMQV[4] for key agreement) and symmetric key systems (AES and SHA). Two security levels were given, with ECC at 128 bits of security using P-256 and at 192 bits of security using P-384. Since Suite B can be used for classified U.S. government communications up through Top Secret (for higher levels of secrecy the NSA has Suite A), presumably Secret requires 128 bits of security and Top Secret requires 192.

2.3. **The third decade (2005-2015).** In 2010, faced with the slow pace with which both private companies and government agencies were converting to ECC, the NSA updated Suite B [39] so as to allow RSA (and DSA) to be used with a 2048-bit modulus (providing 112 bits of security). The announcement said that "During the transition to the use of elliptic curve cryptography in ECDH and ECDSA, DH [in the multiplicative group of a prime field], DSA and RSA can be used with a 2048-bit modulus to protect classified information up to the Secret level." (There was no mention of RSA/DH/DSA for Top Secret level.)

In 2013 the Edward Snowden revelations had a dramatic impact on public perceptions of the NSA's role in promoting ECC. On September 5 of that year *The New York Times* [43] reported that the Snowden documents showed that the NSA had put a back door in the standardized version [3] of the Dual Elliptic Curve Deterministic Random Bit Generator (Dual

[3]For curves in the range used in ECC, researchers are still unable to solve the ECDLP more rapidly using Semaev-type methods than using Pollard-rho; however, the former methods are asymptotically subexponential, and some researchers are concerned that they will eventually be capable of solving the ECDLP more quickly than Pollard-rho in the cryptographic range.

[4]In 2008 ECMQV was dropped from Suite B, presumably for patent reasons.

EC_DRBG) and that at Crypto 2007 two Microsoft researchers (unnamed in the article — they were Dan Shumow and Niels Ferguson) had called attention to the possibility of such a back door.[5] (See [29] for an analysis by John Kelsey of NIST of how a back-doored random bit generator came to be included in the standards.)

The basic assumption in any security claim for the Elliptic Curve DRBG is that the dual points P and Q are generated randomly and independently of one another. From the beginning it was understood that knowledge of a relationship $Q = kP$ completely negates that security (see [11]). In fact, the original Certicom patent application [54] describes how the DRBG can be used for key escrow — namely, the value of k could be in the hands of a court that could release it to the government when issuing a search warrant.

At first it was a mystery why NSA would have bothered to get the EC_DRBG standardized with the back door, since it seemed that hardly anyone (except possibly some U.S. government agencies) would ever use that random bit generator. It was roughly 1000 times slower than DRBGs based on symmetric-key constructions, and three such generators were included with the Elliptic Curve DRBG in the same standards. The only possible advantage of a DRBG based on elliptic curves was that it had a "proof of security" (which was also presented at Crypto 2007, see [11]). But it seemed doubtful that many people would opt for a much slower protocol simply because the standard symmetric primitives such as AES lacked a proof of security.

Then on 20 December 2013 Reuters [36] reported that the RSA company had received a secret ten million dollar payment from the NSA so that they would make the Dual EC_DRBG the default in their BSAFE toolkit.[6] Now it was clear how the back door would have enabled the NSA to get access to many users' decryption keys.

The Dual EC_DRBG is atypical, in that no other standardized ECC protocol has any known way to insert a back door. Nevertheless, public perception of all of ECC took a big hit. Some prominent researchers, such as Bruce Schneier [47],[7] noted the NSA's role over the years in promoting ECC

[5]NIST's Dual EC_DRBG standard included a specific pair of points (P, Q) whose source was not explained, leaving open the possibility that NSA selected P and k first, and then set $Q = kP$. The standard included an option for users to select their own P and Q by means of a seeded hash; that is, users were not required to use the back-doored pair (P, Q). However, in order to get FIPS 140-2 validation of one's implementation one had to use NSA's suggested values of P and Q.

[6]RSA never denied receiving the payment, although they said that under no circumstances would they take a bribe to weaken their customers' cryptographic protection. Rather, the payment was negotiated without the knowledge of the company's cryptographers by business people who were perhaps naive about the NSA's motives. Unimpressed by this explanation, several cryptographers boycotted the 2014 RSA Conference in protest.

[7]In addition, in Scott Aaronson's blog (see http://www.scottaaronson.com/blog/?p=1517 he comments that "as Schneier has emphasized, the fact that NSA has been aggressively pushing elliptic-curve cryptography in recent years invites the obvious speculation

and suggested that that alone might be sufficient reason for people to stop using ECC. But despite widespread anger over the deliberate weakening of standards by NSA, in practice there was no noticeable decline in ECC use. Rather, the main reaction in the cryptographic community was heightened interest in revising the ECC standards and proposing new recommended curves.

Finally, in August 2015 the NSA released the statement on post-quantum cryptography that was mentioned in the Introduction. In it they hinted that they would soon have their own proposals for post-quantum cryptosystems, and stated that the "move... to a quantum resistant algorithm suite" will occur "in the not distant future." In the meantime, people should continue using Suite B, which still relied primarily on ECC — although P-256 had mysteriously vanished from Suite B, leaving just P-384 (and RSA was included with a minimum 3072-bit modulus).

3. Can the NSA Break ECC?

Some people have been suspicious of ECC precisely because the NSA energetically promoted it. Those suspicions seemed to be confirmed, or at least given a new life, when the Snowden documents revealed the back door in Dual EC_DRBG. However, there are several reasons to doubt this speculation.

In the first place, the Snowden documents are fascinating in part for what they do *not* contain. Judging by all of the published and informal reports by journalists and experts who have seen the Snowden documents, there is no evidence that the NSA is ahead of outside researchers in attacking either integer factorization or the ECDLP.

In the second place, ECC has been around for three decades, and NSA has been promoting it for over two decades. If NSA had discovered an efficient general-purpose ECDLP algorithm in the early 1990s, it strains credulity that no one in the outside world has thought of it, despite all the effort that has been put into attacking the ECDLP.

In the third place, ECC started to get strong support from the Information Assurance Directorate (IAD) of NSA during the time when Brian Snow was the Technical Director and Mike Jacobs was the head of IAD. There has never been any evidence — in the reports on the Snowden documents or anywhere else — of any actions by Snow and Jacobs or their researchers that would weaken or undermine cryptographic standards. On the contrary, during that period IAD cooperated with other sectors in pushing for strong security. This was consistent with IAD's mission as the defensive arm of NSA.[8]

---

that they know something about ECC that the rest of us don't." Later in the blog entry Aaronson approvingly quotes Schneier's remark in *The Guardian* that ECC has "constants that the NSA influences when they can."

[8]The offensive arm, called Signals Intelligence (SIGINT), is another matter.

Even before the Snowden leaks it was well known that after the September 11 attacks and the passage of the Patriot Act by the U.S. Congress in October 2001, the balance of power between IAD and SIGINT shifted abruptly.[9] Almost all of the dirty deeds revealed by Snowden are post-2001.

A final reason to doubt that the NSA could break ECC is that it is not in NSA's interest to support a cryptosystem based on a conjecturally hard mathematical problem that the NSA knows to be weak. The reason is that the weakness is likely to be discovered soon by outside critics of NSA and also by adversaries. In the former case the NSA ends up losing credibility, and in the latter case American users (including U.S. government users) can be attacked by cybercriminals and hostile nation-states.

The beauty of the back door into Dual EC_DRBG from the NSA's point of view was that only the NSA would know the discrete log value k that was used to generate Q from P. To the rest of the world — including the cleverest mathematicians and hackers in all of Russia and China — the random bit generator was as impregnable as if the P and Q had been properly chosen. The NSA and no one else could read encrypted messages whose security relied upon pseudo-randomness of the DRBG. And if it weren't for Snowden, most likely no one would have ever known that the NSA knew k.

3.1. **Are the NIST curves weak?** There are both historical and technical reasons why it is unlikely that the NIST curves are back-doored, although this in no way means that the NIST list of recommended curves, which are more than 15 years old, should not be replaced. In this subsection we first recall some of the central issues in curve selection and the circumstances when the NIST curves were generated.

Let $\mathbb{F}_q$ be a finite field of order q, and let E be an elliptic curve defined over $\mathbb{F}_q$. By Hasse's Theorem, we know that the group $E(\mathbb{F}_q)$ of $\mathbb{F}_q$-rational points on E has order $\#E(\mathbb{F}_q) = q + 1 - t$, where $|t| \leq 2\sqrt{q}$; observe that $\#E(\mathbb{F}_q) \approx q$. Suppose that $\#E(\mathbb{F}_q) = hn$, where n is prime and h is a small cofactor. Then $n \approx q$. Let P be a fixed point of order n in $E(\mathbb{F}_q)$. Given a point $Q \in \langle P \rangle$, the *elliptic curve discrete logarithm problem* (ECDLP) is the problem of determining the integer $\ell \in [0, n-1]$ such that $Q = \ell P$. The integer ℓ is called the *discrete logarithm* of Q to the base P.

The main premise for using ECC is that the fastest general-purpose algorithm known for solving the ECDLP is Pollard's rho algorithm, which has *fully-exponential* running time $\approx \sqrt{n} \approx \sqrt{q}$. This is in contrast with the RSA public-key cryptosystem, where *subexponential-time* algorithms are known for solving the underlying integer factorization problem.

Of course, the ECDLP could be easier for specific elliptic curves. The first class of *weak* elliptic curves was discovered in 1990 [35] (see also [23]). In particular, it was shown that the ECDLP for "supersingular" elliptic curves

[9] In 2002 Brian Snow was moved from the technical directorship of IAD to a less influential position within NSA; Mike Jacobs retired from the NSA the same year.

can be efficiently reduced to the discrete logarithm problem in small extensions of $\mathbb{F}_q$ for which subexponential-time algorithms are known. These attacks, known as the *Weil and Tate pairing attacks*, are not a concern in practice since supersingular elliptic curves (and other elliptic curves vulnerable to this attack) can easily be identified by a simple divisibility check and thus avoided.

In the mid-1990s, standards bodies including IEEE P1363, ANSI and ISO began considering ECC. There was still considerable doubt about the security of ECC since the ECDLP was viewed by many as not having received enough scrutiny by cryptanalysts and mathematicians. Nevertheless, ECC standards were drafted by IEEE P133, ANSI and ISO standards bodies.

In 1997, Araki and Satoh [45], Semaev [48], and Smart [51] independently showed that if q is prime and $\#E(\mathbb{F}_q) = q$, then the ECDLP can be solved very quickly. These so-called *prime-field anomalous elliptic curves* are extremely rare, and can easily be identified and avoided. Nonetheless, the attack was of concern to members of standards bodies who wondered whether there were any other weak classes of elliptic curves.

To assuage the fear that new classes of weak elliptic curves might be discovered in the future, the ANSI X9F1 standards committee decided to include in their standards some elliptic curves that had been generated at random. Random selection of these curves would ensure that the curves do not belong to a special class, and thus are unlikely to succumb to some as-yet-undiscovered attack that is effective on curves with very special properties.

For a fixed finite field $\mathbb{F}_q$, there are many elliptic curves to choose from. More precisely, there are approximately $2q$ different isomorphism classes of elliptic curves defined over $\mathbb{F}_q$. In order to conclude that an elliptic curve E defined over $\mathbb{F}_q$ avoids the known attacks, it is of utmost importance to determine the number of points $N = \#E(\mathbb{F}_q)$ on the curve. In 1997, counting the number of points on a random elliptic curve was still a formidable challenge. An NSA representative on the ANSI X9F1 committee offered to provide suitable curves. Note that once an elliptic curve and its alleged number of points N is given, the correctness of the value N can be verified very quickly with 100% certainty.

To ensure that the NSA-generated elliptic curves did not belong to a very special class of curves, a simple procedure was devised whereby the coefficients of an elliptic curve were derived by passing a seed through the hash function SHA-1. Given the seed and its associated elliptic curve, anyone could check that the curve had been generated from the seed. Since SHA-1 is (still) considered to be a one-way function, it would be infeasible for anyone to first select a curve with very special properties, and then find a seed which yields that curve.

The elliptic curves were generated by NSA mathematicians around 1997 and, together with the seeds, were included in the ANSI X9.62 ECDSA standard in 1999 [1] and NIST's FIPS 186-2 standard in 2000 [22]. There are five NIST curves over fields $\mathbb{F}_q$ of prime order, and 10 NIST curves over

characteristic-two fields $\mathbb{F}_q$. In particular, the NIST elliptic curves P-256 (defined over a 256-bit prime field) and P-384 (defined over a 384-bit prime field) were included in NSA's Suite B in 2005 [38].

It should be noted that in the case of elliptic curves over prime fields, no new classes of weak elliptic curves have been discovered since 1997. In particular, no weaknesses in the NIST curves have been discovered since they were proposed around 18 years ago.

Since the Snowden revelations, many people have cast doubts on the NSA-generated NIST elliptic curves even though no concrete weaknesses in them have been discovered since they were proposed in 1997. These people speculate that NSA researchers might have known classes of weak elliptic curves in 1997. With this knowledge, the NSA people could have repeatedly selected seeds until a weak elliptic curve was obtained.

This scenario is highly implausible for several reasons. First, the class of weak curves must be fairly large in order to obtain a weak curve with the seeded-hash method. For concreteness, suppose that p is a fixed 256-bit prime. There are roughly 2^{257} isomorphism classes of elliptic curves defined over $\mathbb{F}_p$. Let s be the proportion of elliptic curves over $\mathbb{F}_p$ that are believed (by everyone except hypothetically the NSA in 1997) to be safe. This class of curves includes essentially all elliptic curve of prime order (with the exception of prime-field anomalous curves and those that succumb to the Weil/Tate pairing attack). Since the proportion of 256-bit numbers that are prime is approximately $1/(256 \ln 2) \approx 2^{-8}$, the proportion of curves that are strong is at least 2^{-8}. Now suppose that the proportion of these curves that the NSA knows how to break is 2^{-40}. Then it can select such a weak curve by trying about 2^{48} seeds. The number of NSA-weak curves is thus approximately 2^{209}. The discovery today of such a large class of weak curves would certainly cast doubt upon the general security of elliptic curves and would be a good reason to abandon ECC altogether.

A second reason for the implausibility of the above scenario is that it is highly unlikely that such a large family of weak elliptic curves would have escaped detection by the cryptographic research community since 1997. It is far-fetched to speculate that NSA would have deliberately selected weak elliptic curves in 1997 for U.S. government usage (for both unclassified and classified communications [38]), confident that no one else would be able to discover the weakness in these curves in the ensuing decades.

There is also an important historical reason why we think the NIST curves are safe. The NIST curves were generated by IAD under Brian Snow and Mike Jacobs in the 1990s, and the bulk of the Snowden revelations, including the Dual EC_DRBG back door, relate to much later events. It is ahistorical to take everything we know about NSA in the post-2001 period and project it back into the 1990s.

Although there is no plausible reason to mistrust the NIST curves, there are two reasons why it is nevertheless preferable to use other curves (either the Edwards curves recommended by Bernstein-Lange [5, 6], or the curves

being promoted by the Microsoft group [10], or perhaps some others). The first reason is public perception — even though it is unlikely that the NIST curves are unsafe, there has been enough speculation to the contrary that to keep everybody happy one might as well avoid the NIST curves. The second reason is that the other curves do have some advantages (Edwards curves have faster point-multiple running times and in certain conceivable side-channel attacks they offer some resistance). It is no discredit to the NIST curves that more efficient alternatives are now available — after all, it has been 18 years, and it would be surprising if no one had come up with anything better by now.

3.2. **Does NSA have an $n^{1/3}$-algorithm for finding elliptic curve discrete logs?** The reason for wondering about this is that in the latest revision of Suite B the NSA has dropped P-256, leaving only P-384. If solving the ECDLP in a group of order n requires roughly $n^{1/2}$ operations, then P-256 suffices for 128 bits of security. But if an $n^{1/3}$-algorithm were known, then one would need P-384 for the same level of security.[10]

It should also be noted that at Asiacrypt 2013 Bernstein and Lange [9] presented an $n^{1/3}$-algorithm. However, it needed a tremendous amount of precomputation, taking time $n^{2/3}$. So from a practical standpoint, as Bernstein and Lange pointed out, it is worthless. However, it is conceivable that the NSA has found (or believes that there might exist) a similar algorithm that requires far less precomputation.

3.3. **What about side-channel and intrusion attacks?** There is little doubt that NSA is the world's leading authority on how to mount these types of attacks. The history of successful attacks of this sort goes back at least to World War II (see §4.1 of [31]). During the Cold War both sides devoted tremendous resources to carrying out and defending against side-channel attacks.

Although parameter choices and implementation algorithms can sometimes prevent certain types of side-channel attacks, it is not realistic to expect that mathematical techniques and protocol design will guard against most such attacks. Rather, if one is really worried about intrusion and side-channel attacks by skillful adversaries, such as the NSA, then one needs tamper-proof devices and physical isolation; mathematics and software are of limited use.

4. Does NSA Know Something the Outside World Doesn't about Quantum Computers?

The Snowden revelations suggest that it does not. According to an article in the *Washington Post* [44], the NSA's efforts to develop a quantum

[10]Another reason why the NSA might have dropped P-256 is that P-256 might succumb to classical Pollard-rho attacks in the next few decades, whereas P-384 will be safe from such attacks far into the future.

computer are a part of a US$79.7 million program called "Penetrating Hard Targets." This is a very small fraction of the NSA's budget. If the NSA were close to developing a practical quantum computer — or if they believed that another nation was — then they would be devoting far more money to this project. The article [44] concludes that "the documents provided by Snowden suggest that the NSA is no closer to success [in quantum computation] than others in the scientific community."

Corporate users of cryptography who consult with the top researchers in quantum computing — some of whom are among the world's leading physicists and engineers — have been told that there's a 50%-50% chance that a practical quantum computer will be available in 15 years (see also [53, 17]). That is presumably at the low end of expert opinions, since people who work in the area would have an interest in erring on the side of optimism.[11] It is unlikely that the NSA has access to better experts than the ones who have been consulted by industry. Moreover, if the NSA really believed in a far quicker time frame for quantum computing, then, as mentioned before, its quantum computation program would not be just one of several projects covered by an $80 million budget.

If practical quantum computers are at least 15 years away, and possibly much longer, and if it will take many years to develop and test the proposed PQC systems and reach a consensus on standards, then a long time remains when people will be relying on ECC. But the NSA's PQC announcement makes it clear that improved ECC standards (for example, an updated list of recommended curves) are not on the Agency's agenda.

5. Theories about the NSA's Motives

One theory — that the timing and wording of the PQC announcement was a case of carelessness or sloppiness on the part of NSA — can be rejected immediately. A policy statement by NIST or by NSA is carefully crafted over a period of time. The committee responsible for drawing it up discusses every sentence; nothing is left to chance or to careless editing. In addition, when asked to clarify the August 2015 statement, the NSA released an updated version that did not differ in any significant way from the first one. So we should start with the premise that the NSA intended for the statement to convey exactly what it did.

[11]Some people are very skeptical about this timeline. Recent progress is quantum factoring is based on reformulating factorization as an optimization problem (see [13, 15]). Concerning Shor's algorithm [50], which applies to both factorization and discrete logs, Dattani and Bryans [15] say: "It is well known that factoring large numbers on classical computers is extremely resource demanding, and that Shor's algorithm could theoretically allow a quantum computer to factor the same number with drastically fewer operations. However, in its 20-year lifespan, Shor's algorithm has not gone far in terms of factoring large numbers. Until 2012 the largest number factored using Shor's algorithm was 15, and today the largest is still only 21. Furthermore, these factorizations were not genuine implementations of Shor's algorithm because they relied on prior knowledge of the answer to the factorization problem being solved in the first place."

We next examine some conjectures about the NSA's motives in its PQC announcement.

5.1. **The NSA can break PQC.** One theory about the NSA's motives is based on the observation that most quantum-resistant systems that have been proposed are complicated, have criteria for parameter selection that are not completely clear, and in some cases (such as NTRU) have a history of successful attacks on earlier versions. Perhaps the NSA believes that it can find and exploit vulnerabilities in PQC much more easily than in RSA or ECC, and for that reason they want the public to hurry toward PQC standards.

At present the process of developing standards for PQC is at an early stage. There is no consensus on the best approach, and the most common proposals are not based on "clean" mathematically hard problems. If NSA gets the standards bodies to rush the process, perhaps they'll make some mistakes, as they did in the case of Dual EC_DRBG. Then NSA can exploit the resulting vulnerabilities.

We believe that such a strategy by NSA is unlikely for the same reason that we don't believe that the NSA can break ECC. Although the NSA might be the best hackers in the world, this technical superiority does not seem to extend to mathematical attacks on basic algorithms, and the NSA knows this. If the NSA has some ideas on how to attack PQC, then it is likely that before long people outside the NSA would have similar ideas. In particular, the cryptographers of other nations (such as Russia and China) would soon be able to attack private and government users in the U.S., and part of the NSA's mission is to prevent this.

This is not to say that the NSA has no ideas of its own about PQC. On the contrary, NSA researchers have been studying PQC systems for many years, and have plans in the not-too-distant future to play an important role (through NIST) in the standardization of quantum-safe cryptographic algorithms.

For a brief overview of the existing candidates for post-quantum cryptography, see Appendix A.

5.2. **The NSA was thinking primarily of government users.** This is the explanation given by an NSA official when a corporate vendor questioned the tone and timing of the announcement. That is, the NSA knew that some U.S. government agencies with limited cybersecurity budgets had been dilatory about moving to ECC (this is why in 2010 they decided to include RSA in Suite B, asking that users at least upgrade to a larger modulus). They did not want those agencies to put their resources into an ECC upgrade and then have no money left for a later upgrade to PQC.

Whether or not this thinking makes sense for U.S. government agencies is hard to say. But it makes no sense in the corporate world. A company's security budget this year has nothing to do with what its security budget might be in 15 years. In addition, the announcement has an immediate

negative impact on ECC deployment. The adoption of ECDSA outside of certain specialized applications (such as Playstation and Bitcoin) has been slow, in large part because of resistance to change by the certification authorities (CAs), who are content with RSA signatures. Now the NSA announcement will give the CAs a further excuse not to update their software to support ECDSA.

More generally, in the commercial sector companies are often notoriously dilatory about improving their cybersecurity,[12] and so will certainly welcome a good justification for postponing any upgrade far into the future. The wording of the NSA announcement gives them an excuse to do precisely that.

In response to the queries from a corporate vendor, the NSA source also mentioned that they were particularly thinking of government agencies that need very long-term security (at level Top Secret or above) that may extend beyond the time when practical quantum computers become available — hence the need to transition to PQC as soon as possible. However, for such users the NSA statement recommends using an additional layer of AES to provide quantum resistance, without waiting for quantum-safe public-key standards. In any case, the statement is directed at the general public and obviously is going to have a big impact in the private sector. If the NSA had wanted to give advice that was intended only for high-security government users, they would have done so.

5.3. **The NSA believes that RSA-3072 is much more quantum-resistant than ECC-256 and even ECC-394.** The quantum complexity of integer factorization or discrete logarithm essentially depends only on the bitlength of the group order, the current state-of-the-art being the successful factorization of 143.[13] Thus, there could be a big lag between the time when quantum computers can solve the ECDLP on P-256 and even P-384 and the time when they can factor a 3072-bit integer.

However, it will require major advances in physics and engineering before quantum computing can scale significantly. When that happens, of course P-256 and P-384 will fall first. But, as the head of cybersecurity research at a major corporation put it, "after that it's just a matter of money" before RSA-3072 is broken. At the point when P-384 is broken it would be unwise to use either ECC or RSA. It is not likely that the gap between quantum cryptanalysis of a 384-bit key and a 3072-bit key will be great enough to serve as a basis for a cryptographic strategy.

[12]Private companies are not the only ones who are notoriously dilatory. For example, the U.S. Department of State has a long history of negligence in the area of cybersecurity (see [18, 42]).

[13]This is not completely true. In [15] the authors point out that RSA moduli larger than 143 can be factored if the two prime divisors differ in only two bits and are not too large — for example, $56153 = 233 \cdot 241$.

5.4. **The NSA has a political need to distance itself from ECC.** There were some peculiarities in the release of the August 2015 statement about preparing for post-quantum crypto. Normally all of the big corporations that do cryptographic work for the U.S. government would have been given some advance notice, but this was not done. Even more surprising, the NIST people were not asked about it, and even researchers in IAD were caught by surprise. It seems that whoever at NSA prepared the release did so with minimal feedback from experts, and that includes their own internal experts.

This suggests that the main considerations might not have been technical at all, but rather Agency-specific — that is, related to the difficult situation the NSA was in following the Snowden leaks. The loss of trust and credibility from the scandal about Dual EC_DRBG was so great that NSA might have anticipated that anything further it said about ECC standards would be mistrusted. The NSA might have felt that the quickest way to recover from the blow to its reputation would be to get a "clean slate" by abandoning its former role as promoters of ECC and moving ahead with the transition to post-quantum cryptography much earlier than it otherwise would have.

If this is correct, then such a step by the NSA raises new questions about credibility. For commercial users of cryptography, the timing of the transition from one paradigm to another should be determined by state-of-the-art technical knowledge and best practices — not by the bureaucratic self-interest of a government agency. If the NSA wants to be regarded as a reliable partner for information assurance, it needs to base its policies and recommendations on a transparent process involving scientific collaboration between the commercial, academic, and government sectors. Such a process would not leave people puzzled and would not give rise to speculation (and occasional paranoia) about what the NSA's true motives might be.

Appendix A. Candidates for Post-Quantum Cryptography

Cryptographers have been working hard on developing public-key cryptosystems that, unlike RSA and ECC, will withstand attacks by quantum computers. In practice, the most important uses of public-key cryptography are digital signatures, key agreement, and encryption (the latter mainly for transporting secret symmetric keys). Quantum-safe digital signatures do not need to be available until ECDSA can be broken by quantum computers.[14] The reason is that signatures are normally verified only at the time they are produced or shortly thereafter; once one has a document with a verified signature, its authenticity does not need to be established again later. In contrast, if an encryption scheme is broken at a later date, then the secret data are revealed.

[14]We are not accounting for the time it will take engineers to replace implementations of conventional public-key cryptosystems with quantum-safe ones.

If one needs to protect long-lived data possibly beyond the time when quantum attacks become practical, then as soon as possible one should start using a quantum-resistant key agreement or public-key encryption algorithm. However, in the interim authentication of the keys by a Certificate Authority can continue to be done using a conventional digital signature scheme.

The following is a brief overview of the current viable candidates for post-quantum cryptography; for further information, see [7, 20].

A.1. **Symmetric-key cryptography.** Symmetric-key encryption schemes such as AES have the property that the fastest quantum attack known for recovering a k-bit secret key takes time $2^{k/2}$. Thus AES with 256-bit keys is believed to provide a 128-bit security level against quantum attacks (that is, half the number of bits of security that it has against conventional attacks). The NSA announcement recommends that vendors use a "layered commercial solution to protect national security information with a long intelligence life" with the layer of quantum-resistant protection implemented by means of a large symmetric-key system. Of course, such systems do not have the functionality of public-key cryptography, and key management will be more cumbersome.

A.2. **Lattice-based cryptography.** Lattices are being intensively studied by cryptographers, in part because they can be used to achieve cryptographic objectives such as fully homomorphic encryption and code obfuscation not known to be achievable using conventional RSA and discrete logarithm cryptography. The most mature lattice-based public-key encryption scheme is NTRU [26], which has been standardized in several forums [2, 28, 46]. For a recent survey of lattice-based cryptosystems, see [41].

A.3. **Hash-based cryptography.** Hash functions are believed to have the same security against quantum computers as against conventional ones, namely, $k/2$ bits of security, where k is the bitlength of hash values. The classical one-time signature scheme, attributed to Diffie and Lamport [32], has been developed into a commercially viable signature scheme using conventional hash functions [12, 27]. However, public-key encryption schemes cannot be based on hash functions, because an encryption function must be invertible rather than one-way and many-to-one.

A.4. **Code-based cryptography.** In 1978, McEliece proposed a public-key encryption scheme using binary Goppa error-correcting codes [34]. Since then, numerous variants that replace Goppa codes with other classes of error-correcting codes have been proposed and later broken. McEliece's original proposal (with updated key sizes) remains the most viable code-based public-key encryption scheme [8]. Code-based signature schemes, on the other hand, are very inefficient and relatively new and untested.

A.5. **Multivariate polynomial cryptography.** The security of these schemes, pioneered by Matsumoto and Imai [33], is based on the difficulty of solving a multivariate system of polynomial equations over a finite field. Many proposed systems have been broken, most spectacularly the SFLASH signature scheme that was standardized by NESSIE in 2003 and completely broken in 2007 [19]. Recent proposals for signature schemes have been more conservative. However, proposals for public-key encryption schemes are still relatively recent and not yet thoroughly studied.

A.6. **Isogeny-based cryptography.** The security of these schemes is based on the difficulty of computing an isogeny of a certain degree between two isogenous supersingular elliptic curves over $\mathbb{F}_{p^2}$ [16]. Note that security has nothing to do with hardness of the conventional ECDLP. Protocols have been designed for public-key encryption and key agreement; however, there has not yet been a proposal for a quantum-safe general-purpose digital signature scheme.

A.7. **Quantum key distribution.** The security of quantum key distribution (QKD) [4] is based not on the hardness of a conventional computational problem, but rather on the Heisenberg uncertainty principle of quantum mechanics. Several companies have marketed QKD devices (see [20]). The challenges in wide-scale deployment include alleviating the requirement for a point-to-point communications channel, reducing the cost, and safeguarding the physical security of devices.

Acknowledgments

We wish to thank Dustin Moody and Susan Landau for reading an earlier draft and making helpful comments.

References

[1] ANSI X9.62, Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), 1999.

[2] ANSI X9.98, Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry, Part 1: Key Establishment, Part 2: Data Encryption, 2010.

[3] E. Barker and J. Kelsey, NIST SP 800-90A, Recommendations for Random Number Generation Using Deterministic Random Bit Generators, National Institute of Standards and Technology, January 2012.

[4] C. Bennett and G. Brassard, Quantum cryptography: public key distribution and coin tossing, International Conference on Computers, Systems & Signal Processing, 1984, pp. 175-179.

[5] D. Bernstein, Curve 25519: new Diffie-Hellman speed records, *Public Key Cryptography — PKC 2006*, LNCS 3958, Springer-Verlag, 2006, pp. 207-228.

[6] D. Bernstein, P. Birkner, M. Joye, T. Lange and C. Peters, Twisted Edwards curves, *Progress in Cryptology — AFRICACRYPT 2008*, LNCS 5023, Springer-Verlag, 2008, pp. 389-405.

[7] D. Bernstein, J. Buchmann and E. Dahmen (editors), *Post-Quantum Cryptography*, Springer-Verlag, 2009.

[8] D. Bernstein, T. Chou and P. Schwabe, McBits: fast constant-time code-based cryptography, *Cryptographic Hardware and Embedded Systems — CHES 2013*, LNCS 8086, Springer-Verlag, 2013, pp. 250-272.
[9] D. Bernstein and T. Lange, Non-uniform cracks in the concerete: the power of free precomputation, *Advances in Cryptology — ASIACRYPT 2013*, LNCS 8270, Springer-Verlag, 2013, pp. 321-340.
[10] J. Bos, C. Costello, P. Longa and M. Naehrig, Selecting elliptic curves for cryptography: an efficiency and security analysis, *Journal of Cryptographic Engineering*, to appear.
[11] D. Brown and K. Gjøsteen, A security analysis of the NIST SP 800-90 elliptic curve random number generator, *Advances in Cryptology — Crypto 2007*, LNCS 4622, Springer-Verlag, 2007, pp. 466-479.
[12] J. Buchmann, E. Dahmen and A. Hülsing, XMSS — a practical forward secure signature scheme based on minimal security assumptions, *Post-Quantum Cryptography — PQCrypto 2011*, LNCS 7071, Springer-Verlag, 2011, pp. 117-129.
[13] C. J. C. Burges, Factoring as optimization, *Microsoft Research*, MSR-TR-200, 2002.
[14] Committee on National Security Systems, Use of public standards for the secure sharing of information among national security systems, Advisory Memorandum 02-15, July 2015.
[15] N. S. Dattani and N. Bryans, Quantum factorization of 56153 with only 4 qubits, arXiv:1411.6758v3 [quant-ph], 27 Nov 2014.
[16] L. De Feo, D. Jao and J. Plût, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies, *Journal of Mathematical Cryptology*, **8** (2014), pp. 209-247.
[17] M. H. Devoret and R. J. Schoelkopf, Superconducting circuits for quantum information: An outlook, *Science*, **339** (2013), pp. 1169-1174.
[18] K. Dilanian, AP exclusive: Under Clinton, State's cybersecurity suffered, 19 October 2015, http://tinyurl.com/nuyqcxd
[19] V. Dubois, P. Fouque, A. Shamir and J. Stern, Practical cryptanalysis of SFLASH, *Advances in Cryptology — CRYPTO 2007*, LNCS 4622, Springer-Verlag, 2007, pp. 1-12.
[20] ETSI White Paper No. 8, Quantum Safe Cryptography and Security: An Introduction, Benefits, Enablers and Challenges, June 2015.
[21] FIPS 186, Digital Signature Standard (DSS), National Institute of Standards and Technology, 19 May 1994.
[22] FIPS 186-2, Digital Signature Standard (DSS), National Institute of Standards and Technology, 27 January 2000.
[23] G. Frey and H. Rück, A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves, *Mathematics of Computation*, **62** (1994), pp. 865-874.
[24] S. Galbraith and S. Gebregiyorgis, Summation polynomial algorithms for elliptic curves in characteristic two, *Progress in Cryptology — INDOCRYPT 2014*, LNCS 8885, Springer-Verlag, 2014, pp. 409-427.
[25] R. Gallant, R. Lambert and S. Vanstone, Improving the parallelized Pollard lambda search on an anomalous binary curve, *Mathematics of Computation*, **69** (2000), pp. 1699-1705.
[26] J. Hoffstein, J. Pipher and J. Silverman, NTRU: a ring-based public key cryptosystem, *Algorithm Number Theory*, LNCS 1423, Springer-Verlag, 1998, pp. 267-288.
[27] A. Huelsing, D. Butin and S. Gazdag, XMSS: Extended Hash-Based Signatures, IETF Internet Draft, 9 March 2015.
[28] IEEE 1363.1, Standard Specification for Public Key Cryptographic Techniques Based on Hard Problems over Lattices, 2008.

[29] J. Kelsey, Dual EC in X9.82 and XP 800-90, May 2014, http://csrc.nist.gov/groups/ST/crypto-review/documents/dualec_in_X982_and_sp800-90.pdf
[30] A. H. Koblitz, N. Koblitz, and A. Menezes, Elliptic curve cryptography: The serpentine course of a paradigm shift, *Journal of Number Theory*, **131** (2011), pp. 781-814.
[31] N. Koblitz and A. Menezes, Another look at security definitions, *Advances in Mathematics of Communications*, **7** (2013), pp. 1-38.
[32] L. Lamport, Constructing digital signatures from a one-way function, Technical Report CSL-98, SRI International, 1979.
[33] T. Matsumoto and H. Imai, Public quadratic polynomial-tuples for efficient signature-verification and message-encryption, *Advances in Cryptology — EUROCRYPT '88*, LNCS 330, Springer-Verlag, 1988, pp. 419-453.
[34] R. McEliece, A public-key cryptosystem based on algebraic coding theory, JPL DSN Progress Report #44-22, 1978, pp. 114-116.
[35] A. Menezes, T. Okamoto and S. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, *IEEE Transactions on Information Theory*, **39** (1993), pp. 1639-1646.
[36] J. Menn, Secret contract tied NSA and security industry pioneer, Reuters, 20 December 2013, http://tinyurl.com/osq39us
[37] National Security Agency, The case for elliptic curve cryptography, archived on 13 October 2005, https://web.archive.org/web/20051013062853/http://www.nsa.gov/ia/industry/crypto_elliptic_curve.cfm?
[38] National Security Agency, Fact Sheet NSA Suite B Cryptography, archived on 25 November 2005, https://web.archive.org/web/*/http://www.nsa.gov/ia/industry/crypto_suite_b.cfm?.
[39] National Security Agency, Fact Sheet NSA Suite B Cryptography, archived on 22 March 2010, https://web.archive.org/web/20100322225318/http://www.nsa.gov/ia/programs/suiteb_cryptography/.
[40] National Secuity Agency, Cryptography today, August 2015, https://www.nsa.gov/ia/programs/suiteb_cryptography/.
[41] C. Peikert, A decade of lattice cryptography, available at http://eprint.iacr.org/2015/939.
[42] D. Perera, Auditors: State Department has history of poor cybersecurity, 17 November 2014, http://tinyurl.com/o4quu7g
[43] N. Perlroth, J. Larson, and S. Shane, N.S.A. able to foil basic safeguards of privacy on web, *The New York Times*, 5 September 2013.
[44] S. Rich and B. Gellman, NSA seeks to build quantum computer that could crack most types of encryption, *Washington Post*, 2 January 2014.
[45] T. Satoh and K. Araki, Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves, *Commentarii Mathematici Universitatis Sancti Pauli*, **47** (1998), pp. 81-92.
[46] J. Schanck, W. Whyte and Z. Zhang, Quantum-Safe Hybrid (QSH) Ciphersuite for Transport Layer Security (TLS), IETF Internet Draft, 20 September 2015.
[47] B. Schneier, NSA surveillance: A guide to staying secure, *The Guardian*, 6 September 2013.
[48] I. Semaev, Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p, *Mathematics of Computation*, **67** (1998), pp. 353-356.
[49] I. Semaev, Summation polynomials and the discrete logarithm problem on elliptic curves, available at http://eprint.iacr.org/2004/031.
[50] P. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, *Proc. 35th Annual Symp. Foundations of Computer Science*, IEEE, 1994, pp. 124-134.

[51] N. Smart, The discrete logarithm problem on elliptic curves of trace one, *Journal of Cryptology*, **12** (1999), pp. 193-196.
[52] J. Solinas, An improved algorithm for arithmetic on a family of elliptic curves, *Advances in Cryptology — CRYPTO '97*, LNCS 1294, Springer-Verlag, 1997, pp. 357-371.
[53] M. Steffen, D. P. DiVincenzo, J. M. Chow, T. N. Theis and M. B. Ketchen, Quantum computing: An IBM perspective, *IBM Journal on Research and Development*, Vol. 55, No. 5, Paper 13, 2011.
[54] S. Vanstone and D. Brown, Elliptic curve random number generation, International patent application, WO 2006/076804 A1, published on 27 July 2006.
[55] M. Wiener and R. Zuccherato, Faster attacks on elliptic curve cryptosystems, *Selected Areas in Cryptography — SAC '98*, LNCS 1556, Springer-Verlag, 1999, pp. 190-200.

Department of Mathematics, Box 354350, University of Washington, Seattle, WA 98195 U.S.A.
E-mail address: koblitz@uw.edu

Department of Combinatorics & Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1 Canada
E-mail address: ajmeneze@uwaterloo.ca