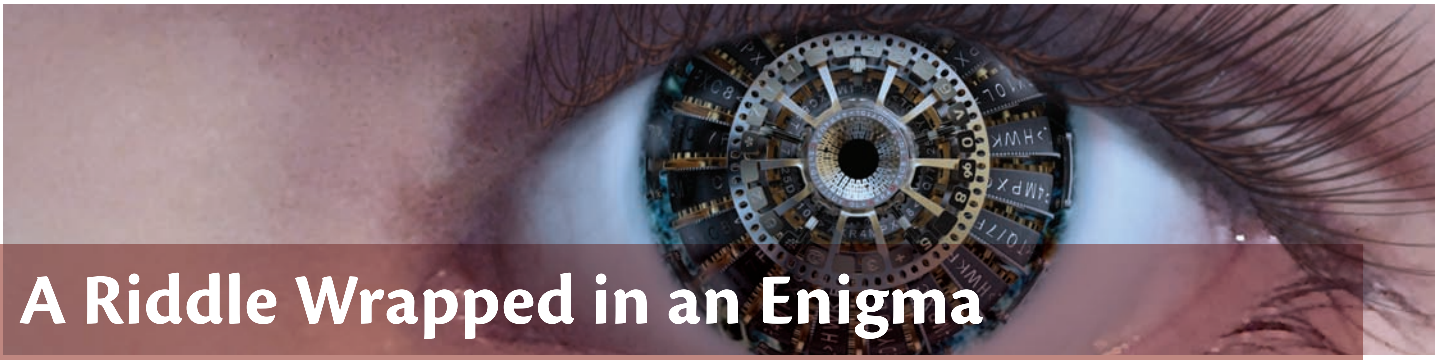

A Riddle Wrapped in an Enigma

Neal Koblitz | University of Washington
Alfred Menezes | University of Waterloo

In August 2015, the NSA released a major policy statement on the need for postquantum cryptography. Certain peculiarities in its wording and timing have puzzled many people and given rise to speculation concerning the NSA, elliptic curve cryptography, and quantum-safe cryptography. Of the various theories that have been proposed, some seem more plausible than others, but a definitive explanation is elusive.

> "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)

In August 2015, the US government's NSA released a major policy statement on the need to develop standards for postquantum cryptography (PQC).[1] The NSA, like many other organizations, believes that the time is right to make a major push to design public-key cryptographic protocols whose security depends on hard problems that can't be solved efficiently by a quantum computer. Ever since Peter Shor's pioneering work more than 20 years ago,[2] it has been known that both the integer factorization problem, upon which RSA is based, and the elliptic curve discrete logarithm problem (ECDLP), upon which elliptic curve cryptography (ECC) is based, can be solved in polynomial time by a quantum computer.

The NSA announcement will give a tremendous boost to efforts to develop, standardize, and commercialize quantum-safe cryptography. While standards for new postquantum 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 the Advanced Encryption Standard (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:[1]

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

1540-7993/16/$33.00 © 2016 IEEE

The NSA seemed to be suggesting that practical quantum computers were coming so soon that people who hadn't yet upgraded from RSA to ECC shouldn't bother to do so, and instead should save their money for the future upgrade to postquantum 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."[1] 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 NIST and elsewhere have noticed that the NSA hasn't 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 the NSA has no interest in this topic because it now views ECC as only a stopgap solution. The statement in fact advises against "making a significant expenditure" to upgrade to any of the Suite B algorithms, let alone to any new ECC standards using updated curves.[1] Even industrial and government users of antiquated protocols should just sit tight and wait for postquantum standards. This caught many people by surprise, because it's widely believed that a consensus on such standards is many years away, and ECC will continue to be used extensively for at least another decade or two.

In this article, we evaluate the various theories, speculations, and interpretations that have been proposed for the NSA's sudden change of course. In the interest of brevity, in this published version we've omitted mathematical details, most bibliographical references, and an appendix giving an overview of the main candidates for quantum-safe cryptography. The full version is in the Cryptology ePrint Archive.[3]

History: The NSA and ECC

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.

The First Decade (1985–1995)

Almost from the beginning there were indications that, of the available public-key systems, the NSA preferred ECC. In the early 1990s, NIST proposed a Digital Signature Algorithm (DSA) that the NSA had developed. Although DSA is based on the discrete log problem in a finite field—not on an elliptic curve—it signaled a dissatisfaction with factorization-based systems in the NSA (perhaps in part because of the high licensing fees for the patented RSA algorithm). Shortly after DSA was approved for government 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 it 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 the NSA publicly and decisively gave support to ECC occurred at an ANSI meeting in December 1995. 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 the NSA believed that ECC had sufficient security to be used for secure communications among all US 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 the NSA. The ECC standards were approved.

The Second Decade (1995–2005)

In the late 1990s, NIST proposed three families of five curves (each at a different security level) for ECC (published as a federal information-processing standard [FIPS] in 2000). These are the 15 "NIST curves." Two of the families consist of curves over binary fields, and the other one contains curves over prime fields. Because of recent progress attacking the ECDLP on curves over binary fields using summation-polynomial methods, some experts now have doubts about the long-term security of all elliptic curves over binary fields, believing that those methods will eventually be capable of solving the ECDLP more quickly than the Pollard-rho algorithm in the cryptographic range. Most experts, however, doubt these claims. Nevertheless, it's widely believed that 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 bit length of the prime.

The 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" on its website.[4] This paper described RSA as a "first generation" public-key technology that had been superseded by ECC.

In conjunction with this recommendation, on 16 February 2005, the NSA announced its Suite B recommended algorithms.[5] Ironically, in its original form it included no RSA (or DSA) protocols, but only ECC (ECDSA for signatures, and Elliptic Curve Diffie-Hellman [ECDH] and Elliptic Curve Menezes-Qu-Vanstone [ECMQV] for key agreement) and symmetric key systems (AES and Secure Hash Algorithm [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. Because Suite B can be used for classified US 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.

> **The Edward Snowden revelations had a dramatic impact on public perceptions of the NSA's role in promoting elliptic curve cryptography (ECC).**

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 to allow RSA (and DSA) to be used with a 2,048-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 2,048-bit modulus to protect classified information up to the Secret level."[5] (There was no mention of RSA/DH/DSA for the 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 5 September of that year, *The New York Times* reported that the Snowden documents showed that the NSA had put a back door in the standardized version of the Dual Elliptic Curve Deterministic Random Bit Generator (Dual EC_DRBG) and that, at the Crypto 2007 Rump Session, 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. (NIST's Dual EC_DRBG standard included a specific pair of points (P,Q) whose source wasn't explained, leaving open the possibility that the NSA had selected P and then set Q equal to a known multiple of P.)

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 expressing Q as a known multiple of P completely negates that security. In fact, the original Certicom patent application (WO 2006/076804 A1, published 27 July 2006) describes how the DRBG can be used for key escrow—namely, the relation between P and Q 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 the NSA would have bothered to get the EC_DRBG standardized with the back door, because it seemed that hardly anyone (except possibly some US government agencies) would ever use that random bit generator. It was roughly 1,000 times slower than DRBGs based on symmetric-key constructions, and three symmetric-key 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. 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 reported that the RSA company had received a secret $10 million payment from the NSA so that it would make the Dual EC_DRBG the default in its BSAFE toolkit.[6] (RSA never denied receiving the payment, although it said that under no circumstances would it take a bribe to weaken its customers' cryptographic protection.) 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 and Scott Aaronson, noted the NSA's role over the years in promoting ECC and suggested that that alone might be sufficient reason for people to stop using ECC. But despite widespread anger over the NSA's deliberate weakening of standards, 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 postquantum cryptography that was mentioned in the introduction. In it, the NSA hinted that it would soon have its own proposals for postquantum cryptosystems and stated that the "move…to a quantum resistant algorithm suite" will occur "in the not distant future."[1] 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 3,072-bit modulus).

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.

First, the Snowden documents are fascinating in part for what they don't contain. Judging by all the published and informal reports by journalists and experts who have seen the Snowden documents, there's no evidence that the NSA is ahead of outside researchers in attacking either integer factorization or the ECDLP.

Second, ECC has been around for three decades, and the NSA has been promoting it for more than two. If the 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.

Third, ECC started to get strong support from the NSA's Information Assurance Directorate (IAD) 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 the NSA. (The offensive arm, called SIGINT, is another matter.)

Almost all the dirty deeds revealed by Snowden are post-2001. Even before the Snowden leaks, it was well known that after the September 11 attacks and the passage of the PATRIOT Act by the US Congress in October 2001, the balance of power between IAD and SIGINT shifted abruptly. (In 2002, Snow was moved from the technical directorship of IAD to a different position in the NSA that had high status but little influence, particularly with regard to actions that were being proposed by SIGINT; Jacobs retired from the NSA the same year.)

A final reason to doubt that the NSA could break ECC is that it isn't in the 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 by critics outside the NSA and also by adversaries. In the former case, the NSA ends up losing credibility, and in the latter case, American users (including US 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 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 on the DRBG's pseudorandomness. And if it weren't for Snowden, most likely no one outside the NSA would have ever known that the NSA knew the secret relation between P and Q.

Are the NIST Curves Weak?

There are both historical and technical reasons why it's unlikely that the NIST curves are back-doored, although this in no way means that NIST's list of recommended curves, which is more than 15 years old, shouldn't be replaced. Here, we first summarize some of the central issues in curve selection and the circumstances when the NIST curves were generated. We omit mathematical details, which can be found in the full version.[3]

Recall that the presumed intractability of the ECDLP is at the heart of ECC. Roughly speaking, this means that, given two points on a (suitably chosen) elliptic curve, it's prohibitively time consuming to express one as an integer multiple of the other. The main premise for using ECC is that the fastest general-purpose algorithm known for solving the ECDLP is the Pollard-rho algorithm, which has fully exponential running time approximately equal to the square root of the size of the elliptic curve. This is in contrast to RSA, 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.[7] This wasn't a major concern, because this set of elliptic curves, known as "low embedding degree curves," can easily be identified by a simple divisibility check and thus avoided.

In the mid-1990s, standards bodies including IEEE P1363, ANSI, and the International Organization for

Standardization (ISO) began considering ECC. Many people still had doubts about ECC's security and thought that the ECDLP hadn't received enough scrutiny by cryptanalysts and mathematicians. Nevertheless, ECC standards were drafted by IEEE P133, ANSI, and ISO standards bodies.

In 1997, the public learned of a result that showed that if the number of points on the curve happens to be exactly equal to the (prime) number of elements in the field, then the ECDLP can be solved very quickly.[8–10] 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 standards bodies' members, 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 its standards some elliptic curves that had been generated at random. Random selection of these curves would ensure that the curves don't belong to a special class and thus are unlikely to succumb to an as-yet-undiscovered attack that's effective on curves with very special properties. When selecting an elliptic curve, to be sure that it avoids the known attacks, it's of utmost importance to determine the number of points 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 are given, the correctness of that value can be verified very quickly with 100 percent certainty.)

To ensure that the NSA-generated elliptic curves didn't belong to a very special class of curves, a simple procedure was devised whereby an elliptic curve's coefficients 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. Because SHA-1 is 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 that 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 and NIST's FIPS 186-2 standard in 2000. There are five NIST curves over fields of prime order, and 10 NIST curves over characteristic-two fields. 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 the NSA's Suite B in 2005.

Note 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 19 years ago.

Since the Snowden revelations, many people have cast doubt 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 would have had to have been extremely large to obtain a weak curve with the seeded-hash method. A computation leads to the conclusion that there would have had to have been a set of roughly 2^{209} curves for which the NSA knew an attack that no one else was aware of. It's highly unlikely that such a large family of weak elliptic curves would have escaped detection by the cryptographic research community from 1997 to the present.

It's far-fetched to speculate that the NSA would have deliberately selected weak elliptic curves in 1997 for US government use (for both unclassified and classified communications), confident that no one else would be able to discover the weakness in these curves in the ensuing decades. Such a risky move by the NSA would have been inconsistent with the agency's mission.

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

Although there's no plausible reason to mistrust the NIST curves, there are two reasons why it's nevertheless preferable to use other curves (the Edwards curves recommended by Daniel Bernstein and Tanja Lange, the curves being promoted by the Microsoft group, or perhaps some others). The first reason is public perception—even though it's 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 might have some advantages, such as faster point-multiple running times and some resistance to certain types of side-channel attacks. It's no discredit to the NIST curves that more efficient alternatives are now available—after all, it's been 19 years, and it would be surprising if no one had come up with anything better by now.

Does the NSA Have a Cube-Root Algorithm for Finding Elliptic Curve Discrete Logs?

In the latest revision of Suite B, the NSA 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. (Another possible explanation for why the NSA might have dropped P-256 is that P-256 might succumb to classical Pollard-rho attacks—or small improvements thereof—in the next few decades, whereas P-384 will be safe from such attacks far into the future.)

Also note that at Asiacrypt 2013, Bernstein and Lange 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's worthless. However, it's conceivable that the NSA has found (or believes that there might exist) a similar algorithm that requires far less precomputation.

> **It isn't in the NSA's interest to support a cryptosystem based on a conjecturally hard mathematical problem that the NSA knows to be weak.**

What about Side-Channel and Intrusion Attacks?

There's little doubt that the 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. 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 isn't 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.

Does the 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*, the NSA's efforts to develop a quantum computer are a part of a $79.7 million program called "Penetrating Hard Targets."[11] This is a very small fraction of the NSA's budget. If the NSA were close to developing a practical quantum computer—or if it believed that another nation was—then it would be devoting far more money to this project. The article in the *Post* 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."[11]

Among corporate users of cryptography, the most commonly cited prediction coming from research leaders in quantum computing is that, based on progress toward scalable fault-tolerant quantum computing, there's a 50–50 chance that a practical quantum computer will be available in 15 years.[12] This prediction is presumably at the low end of possible time frames, since people who work in the area would have an interest in erring on the side of optimism. It should be noted that some people are very skeptical about this timeline.

It's unlikely that the NSA has access to more knowledgeable experts than the ones consulted by industry, who might have given them a forecast that's even more optimistic than the 15-year prediction. Moreover, if the NSA really believed in a far quicker time frame for quantum computing, then as mentioned before, its quantum computation program wouldn't 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 people will be relying on ECC for a long time to come. But the NSA's PQC announcement makes it clear that improved ECC standards (for example, an updated list of recommended curves) aren't on the agency's agenda.

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 the NSA—can be rejected immediately. A policy statement by NIST or by the 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 didn't 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.

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

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 aren't completely clear, and in some cases 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 it wants the public to hurry toward PQC standards.

At present, the PQC standards development process is at an early stage. There's no consensus on the best approach, and the most common proposals aren't based on "clean" mathematically hard problems. If the 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 the NSA can exploit the resulting vulnerabilities.

> **Any theory about the NSA being able to break RSA is as speculative as those about its ability to break ECC or postquantum cryptography.**

We believe that such a strategy by the NSA is unlikely for the same reason that we don't believe that the NSA can break ECC. Although the NSA might have the best hackers in the world, this technical superiority doesn't 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's 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 US, and part of the NSA's mission is to prevent this.

This isn't 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 to play an important role (through NIST) in the standardization of quantum-safe cryptographic algorithms.

The NSA Can Break RSA

Another theory is that the NSA has found a much faster integer factorization algorithm and wants to discourage users from transitioning from RSA to ECC so that the agency can use this new algorithm to decrypt a significant proportion of Internet traffic. It's perhaps more likely that the NSA can break RSA than that it can break ECC. Indeed, for the ECDLP on a random prime-field curve, there's been no significant progress in the entire 30-year history of ECC, whereas there have been major breakthroughs in the integer factorization problem.

It's a bit odd that the revised Suite B allows 3,072-bit RSA but only P-384 and not P-256 for ECC. A 3,072-bit RSA modulus provides only 128-bits of security against known factoring attacks, whereas P-384 provides 192 bits of security against known ECDLP attacks. One reason Suite B might permit 3,072-bit RSA instead of 7,680-bit RSA (which provides 192 bits of security against factoring attacks) is in the interest of efficiency. However, 7,680-bit RSA isn't really much slower than 3,072-bit RSA. On the other hand, if it is the case that the NSA can break 3,072-bit RSA but not P-384, it could then direct its vendors to supply P-384 to US government users and hope that the rest of the world uses 3,072-bit RSA (which is a possibility given the general level of mistrust in ECC and the fact that RSA is still much more widely used than ECC).

Of course, any theory about the NSA being able to break RSA is as speculative as those about its ability to break ECC or PQC.

The NSA Was Thinking Primarily of Government Users

This was 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 US government agencies with limited cybersecurity budgets had been dilatory about moving to ECC (this is why in 2010 it decided to include RSA in Suite B, asking that users at least upgrade to a larger modulus). It didn't 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 US 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 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 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. Thus, many users will 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 it was particularly thinking of government agencies that need very long-term security (at level Top Secret or above) that might 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, it would have done so.

The NSA Believes that RSA-3,072 Is Much More Quantum Resistant than ECC-256 and ECC-384

The quantum complexity of integer factorization or discrete logarithm essentially depends only on the group order's bit length. 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 3,072-bit integer.

However, major advances in physics and engineering are required 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-3,072 is broken. At the point when P-384 is broken, it would be unwise to use either ECC or RSA. It's unlikely that the gap between quantum cryptanalysis of a 384-bit key and a 3,072-bit key will be great enough to serve as a basis for a cryptographic strategy.

The NSA Is Using a Diversion Strategy Aimed at Russia and China

Suppose that the NSA believes that, although a large-scale quantum computer might eventually be built, it will be hugely expensive. From a cost standpoint, it will be less analogous to Alan Turing's bombe than to the Manhattan Project or the Apollo program, and it will be within the capabilities of only a small number of nation-states and huge corporations.

Further suppose that the NSA already has the most valuable national-security secrets protected by quantum-safe symmetric key primitives such as AES256, and doesn't rely on public-key cryptography for material that has the most restricted circulation (for which Suite A rather than Suite B would be used). Indeed, the NSA might conclude that, more generally, quantum computation will have very limited impact on cybersecurity: a criminal enterprise is unlikely to invest several billion dollars to be able to break into Alice's RSA-protected credit card account or Bob's ECC-protected Bitcoin wallet.

Suppose also that, in thinking about the somewhat adversarial relationship that still exists between the US and both China and Russia, especially in the area of cybersecurity, the NSA asked itself "How did we win the Cold War? The main strategy was to goad the Soviet Union into an arms race that it could not afford, essentially bankrupting it. Its GNP was so much less than ours; what was a minor setback for our economy was a major disaster for the Soviet Union's. It was a great strategy. Let's try it again."

In other words, according to this theory, the main intended audiences for the NSA announcement were the Chinese and Russians, who would be goaded into a pointless expenditure of vast sums of money to build a quantum computer that won't do them much good. In this way, those resources won't be put to uses that the NSA would regard as much more threatening.

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 the big corporations that do cryptographic work for the US government would have been given some advance notice, but this wasn't done. Even more surprising, the NIST people weren't asked about it, and even IAD researchers were caught by surprise. It seems that whoever at the NSA prepared the release did so with minimal feedback from experts, and that includes its 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 NSA's difficult situation following the Snowden leaks. The loss of trust and credibility from the scandal about Dual EC_DRBG was so great that the 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 postquantum 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 not on some political calculation but rather on a transparent process involving scientific collaboration among the commercial, academic, and government sectors. Such a process wouldn't leave people puzzled and wouldn't give rise to speculation (and occasional paranoia) about what the NSA's true motives might be.

We cannot offer a definitive conclusion; the reason for the NSA's pulling back from ECC remains an enigma. Readers are invited to choose from the possible explanations we've given, or come up with their own theories. ■

References

1. "Cryptography Today," Nat'l Security Agency, Aug. 2015; tinyurl.com/SuiteB.
2. P. Shor, "Algorithms for Quantum Computation: Discrete Logarithms and Factoring," *Proc. 35th Ann. Symp. Foundations of Computer Science* (SFCS 94) 1994, pp. 124–134.
3. N. Koblitz and A. Menezes, "A Riddle Wrapped in an Enigma," Cryptology ePrint Archive, 20 Oct. 2015; eprint.iacr.org/2015/1018.
4. "The Case for Elliptic Curve Cryptography," Nat'l Security Agency, 13 Oct. 2005; tinyurl.com/NSAandECC.
5. "Fact Sheet NSA Suite B Cryptography," Nat'l Security Agency, 22 Mar. 2010; tinyurl.com/NSASuiteB.
6. J. Menn, "Secret Contract Tied NSA and Security Industry Pioneer," Reuters, 20 Dec. 2013; tinyurl.com/osq39us.
7. A. Menezes, T. Okamoto, and S. Vanstone, "Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field," *IEEE Trans. Information Theory*, vol. 39, no. 5, 1993, pp. 1639–1646.
8. T. Satoh and K. Araki, "Fermat Quotients and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves," *Commentarii Mathematici Universitatis Sancti Pauli*, vol. 47, no. 1, 1998, pp. 81–92.
9. I. Semaev, "Evaluation of Discrete Logarithms in a Group of p-Torsion Points of an Elliptic Curve in Characteristic p," *Mathematics of Computation*, vol. 67, no. 221, 1998, pp. 353–356.
10. N.P. Smart, "The Discrete Logarithm Problem on Elliptic Curves of Trace One," *J. Cryptology*, vol. 12, no. 3, 1999, pp. 193–196.
11. S. Rich and B. Gellman, "NSA Seeks to Build Quantum Computer that Could Crack Most Types of Encryption," *Washington Post*, 2 Jan. 2014; tinyurl.com/NSAQuantumComputer.
12. M. Mosca, "Cybersecurity in an Era with Quantum Computers: Will We Be Safe?," Cryptology ePrint Archive, 5 Nov. 2015; eprint.iacr.org/2015/1075.

Neal Koblitz is a professor at the University of Washington in Seattle. He works in number theory and cryptography and has written extensively on educational issues. Koblitz received a PhD in mathematics from Princeton. He's the author of six books, of which the last one, *Random Curves: Journeys of a Mathematician* (Springer 2007), is autobiographical. Contact him at koblitz@uw.edu.

Alfred Menezes is a professor in the Department of Combinatorics and Optimization at the University of Waterloo. His research interests are in cryptography and algorithmic number theory. Menezes received a PhD in mathematics from the University of Waterloo. He's coauthor of four books including *Handbook of Applied Cryptography* and *Guide to Elliptic Curve Cryptography*. Contact him at ajmeneze@uwaterloo.ca.