The Eurocrypt'92 Controversial Issue
Trapdoor Primes and Moduli

Introduction

Motivated by the public controversy surrounding the draft standard for digital signatures (DSS) proposed by NIST, the Program Committee of Eurocrypt'92 decided to hold a panel discussion on the larger issue of trapdoor primes and moduli. The panel members were:

Yvo Desmedt, University of Wisconsin

Peter Landrock, Aarhus University

Arjen Lenstra, Bellcore

Kevin McCurley, Sandia National Laboratories

Andrew Odlyzko, AT&T Bell Laboratories

Rainer Rueppel, R^3 Security Engineering

Miles Smid, National Institute of Standards and Technology

Each of the panel members was given time to make a personal statement on the subject. Then an open discussion followed. For this report each of the panel members was asked to provide a summary of his own personal statement. The following contributions are ordered in the same sequence as the statements were given at Eurocrypt'92.

For people interested in the public discussion on DSS, the special section in the July 1992 issue of the Communications of the ACM provides further information on the ongoing debate.

Rainer A. Rueppel

R^3 Security Engineering

There is an increased awareness that the electronic exchange of data requires security. Security comes in two flavours: authenticity and confidentiality. Authenticity means that the receiver can verify the origin and the integrity of a received message, confidentiality means that only the intended recipient is able to read the message. Two parties wishing to communicate securely must use exactly the same cryptographic algorithms and must be in possession of the right keys. To provide security in open systems requires national and international standards. Therefore, in the US the National Institute for Standardization (NIST) has developed and proposed the Digital Signature Standard (DSS). The objective of the DSS is to provide authenticity. Why is there such an uproar over this standards proposal which after all seems to be a step in the right direction ?

R.A. Rueppel (Ed.): Advances in Cryptology - EUROCRYPT '92, LNCS 658, pp. 194-199, 1993.

The public debate reveals various levels of interests. There is a level of social and political interests. Citizens and corporate users are concerned about the privacy and integrity of their communications. The government is concerned about the country's economy, about national security and law enforcement. There is some room for mutual distrust on this level.

But there is also a level of personal and business interests. There are manufacturers who want to protect their investments in security technology, there are patent holders who are concerned about royalties, there are scientists who use their reputation against or in favour of the DSS. Not many other research fields react so nervously to claims and allegations as cryptology. But after all, the security of modern computer and communication systems is a matter of trust; it cannot be proven, it can only be reassured to some degree.

It might be that the harsh attacks and criticism of the DSS will harm the larger issue of integrity and privacy protection. For non-experts it is difficult to follow the scientific discussion, the result might be a general distrust of all cryptographic techniques and standards. It might also be that the harsh attacks will improve the DSS (a first result is NIST's increase of the allowable size of the prime p). But the issue is not RSA or DSS or any other signature standard. In all likelihood there will be more than one signature standard anyway. The main issue is to establish trust in the integrity and the privacy of stored and communicated data without inhibiting the flow of information or the access to information services. Regarding the present situation where there is no security available on public telecommunication services, any step in the direction of secure systems must be considered a progress.

Arjen K. Lenstra

Bellcore

In this note, a trapdoor for a public key cryptosystem is an additional piece of information about the public key that undermines the computational infeasibility to derive the corresponding secret private key.

In RSA, the construction of a public/private key pair requires, as far as we know, the knowledge of secret information (i.e., the factorization) about the modulus. Therefore, in RSA one necessarily has to trust the vendor of the modulus. It hardly makes sense for a vendor to put trapdoors in the moduli he sells because the secret information is available to him anyway. If the modulus is not bought but generated using some software package, either all programmers involved in the production of that software package have to be trusted or the software has to be verified very carefully, which is far beyond the capabilities of the average user: it is not hard to put a trapdoor into moduli in such a way that the trap is undetectable for outsiders but easily recognizable for insiders.

In discrete logarithm based cryptosystems no further secret information about the modulus (i.e., the prime) is needed to construct a public/private key pair. Therefore, a sufficiently large prime from any source can safely be used, at least if

it does not contain a trapdoor. For users who cannot be sure how much to trust the entity from which they receive their prime this may pose a threat. A well-known way to trap a prime p is to generate it such that p-1 has only small factors. This works for any size p, but is easily detectable, and therefore poses no real threat. Another trap is to generate p in such a way that p divides f(X/Y)*(Y**d) for an integral polynomial f of small degree d with small coefficients, and integers X and Y close to p**(1/d). For the discrete logarithm problem modulo such primes a fairly recent algorithm can be used which is based on the number field sieve, and which is faster than any of the previous methods. It is expected that this algorithm is currently only practical for trapped primes up to about 600 bits, so this trap only makes sense for primes up to that size. Furthermore, this kind of trap can be detected, although this requires more work than an average user will be able to invest. The probability that a randomly chosen prime turns out to be trapped in this way is negligibly small.

Miles E. Smid

National Institute of Standards and Technology

DSA "Trapdoor"

A claim was made that a dishonest Certification Authority could purposely select a value of p for its own users which would permit the Certification Authority to recover the private keys of the users. This property was called a "trapdoor" in the proposed NIST Digital Signature Algorithm (DSA).

Response:

No evidence of an intent to put a "trapdoor" in the DSA has been presented.

The NIST proposed Digital Signature Standard (DSS) specifies a digital signature algorithm. It does not discuss all the ways the algorithm may be used or misused. The Qualifications section of the DSS Announcement states that "The responsible authority in each agency or department shall assure that an overall implementation provides an acceptable level of security." The proposed DSS specifically states that, "Systems for certifying credentials and distributing certificates are beyond the scope of this standard." Therefore, one would not expect an algorithm specification standard to cover the case of a dishonest certification authority.

The DSS allows users to generate their own primes, p and q. The DSS also allows the user to use primes generated by a trusted party or a Certification Authority. If primes are known to be randomly generated, the user can even accept primes generated by a distrusted Certification Authority. One can construct special primes that are considered weak, and if they were used the private keys of the users might be recovered. (Note that many other algorithms have similar weak values.) However, the probability of generating a weak prime at random is

infinitesimally small. (The probability of generating a weak p at random has been estimated to be less than 10^-90.)

Two responses to the NIST request for comments pointed out that the use of a one-way function, such as the NIST proposed Secure Hash Algorithm (SHA), in the process that generates p and q could ensure that weak values occur only randomly. By making publicly known the input to the SHA, the resulting p, the resulting q, and the process, the user would be able to verify that weak primes were not purposely constructed. A technique which makes use of the SHA in the generation of DSA primes will be proposed by NIST.

Warning! As with all systems using a Certification Authority, the Certification Authority must be trusted to correctly establish the binding between the user's identity and the user's public key.

Kevin S. McCurley

Sandia National Laboratories

In their seminal 1976 paper on public-key cryptography, Diffie and Hellman described a trap-door cipher as one that ". . . allows the designer to break the system after he has sold it to a client and yet falsely maintain his reputation as a builder of secure systems." The subject of trapdoor moduli has lately received much attention in the popular press, particularly as it applies to the U.S. draft standard for digital signatures, known as DSS. In my opinion, the situation has been wildly distorted by the press, leading to a general distrust of DSS that lacks any serious scientific justification.

So far, it has not been demonstrated that trapdoor moduli for the discrete logarithm problem can be constructed such that a) they are hard to detect, and b) knowledge of the trapdoor provides a quantifiable computational advantage for parameter sizes that could actually be computed by known methods, even with foreseeable machines.

Even if trapdoor keys can eventually be constructed, this will have few consequences for DSS. Many people skilled in the art will recognize that there are numerous ways to build trapdoor moduli for RSA, but all this means is that one needs to be careful in the method of selecting keys. For example, if a user is provided software from another party to generate keys for a cryptographic scheme, then the user needs to trust the provider of the software to produce keys that are truly random and free of any predictability.

Most of the attention devoted to the issue of trapdoor primes and moduli has been motivated by political and business influences rather than scientific concerns. Some people hold political views that cause them to distrust DSS for the simple reason that it was proposed by a US government agency. To this day, some people believe that the US government designed DES to incorporate a trap-door, making it easy for them to break it. No evidence has ever been put forward for the existence of such a trapdoor, but conspiracy theories persist. Another source of influence is the fact that cryptographic policy and standards can have

serious business consequences. Care needs to be taken to distinguish conclusions that are motivated by such concerns rather than objective scientific judgements.

Yvo Desmedt

University of Wisconsin

Because NIST has not standardized a prime in the DSS proposal, the discussion on trapdoor primes is not an issue.Moreover the probability of having a weak prime is small.

It is unfortunate that many comments on the DSS proposal, e.g., claims that NIST has lost considerable credibility with the non-military cryptographic research community, have been exaggerations. It seems that these comments have no scientific grounds and that, unfortunately, other interests have overshadowed a scientific discussion. The real issue was the size of p, which now has been adjusted by NIST. Implementations which neither use p's of 1024 bits nor allow to update p to a 1024 bit should not receive a standard certification, but only be validated for a very short time. Although there is no scientific evidence today that q could be too small, the DSS proposal could stimulate a lot of research on breaking such discrete logarithms.

There is a need for future standards, such as one for privacy protection, one for very fast authentication, and standards that allow to fulfill different needs such as threshold signatures (signatures in which the secret key is shared such that a threshold of shareholders can sign, but less cannot) which can be obtained based on RSA, but it is not known how to achieve them (in a practical way) using DSS.

Andrew Odlyzko

AT&T Bell Laboratories

1. Progress in factoring and discrete logarithms

There has been substantial progress on both fronts in the last 15 years. When the RSA system was invented, the largest integers that experts could be sure of being able to factor with the algorithms and computers that were available to them at that time were on the order of 38 to 45 decimal digits. Today, integers of between 115 and 130 digits can be factored. Many of the projections that one sees of where the field is going take into account the progress in computer technology, with individual machines becoming much faster and more of them becoming available on networks. However, it is only prudent to allow for progress in algorithms. If one considers what it was that allowed the advance from factoring 38 decimal digit integers 15 years ago to factoring 115 decimal digit integers today, it seems that only about half of this was due to computer technology, with the other half coming from better algorithms. There is no reason to expect that the future will be any different. Terefore the moduli that are used should be large enough to guard against such algorithmic improvements.

2. Trapdoor primes and moduli

Trapdoor primes and moduli are not a significant issue. The only methods that have been suggested for constructing such primes and moduli yield only a slight advantage to the person who chooses them, an advantage that depends on the use of particular algorithms for factoring and discrete logarithms. If one chooses large enough moduli to guard against advances in algorithms and computers that one can prudently expect, this threat disappears.

Peter Landrock

Aarhus University

In any known public key scheme based on properties of primes, there seems be so me primes that are weaker than others. Thus it must be considered in the key generation phase first of all if these keys should be avoided explicitely by the key generation program, and - if so - secondly how they can be avoided. An important property of any key generation program is that the keys be chosen randomly from a key space, which is sufficiently large. Any key generation program not achieving this at an acceptable level, which can be estimated by its creator, is dangerous to use.

Once the properties of weak keys have been identified, the strategy to try to avoid them should depend on an estimate of the probability that the key generation program will return a prime with that property. For instance, if DES keys are generated in a random manner, there is no need to check if a weak or semi-weak key is returned. The probability that this will happen is so small that it can be completely ignored .

Consequently, if the estimation shows that the probability for the occurrence of weak primes is sufficiently small, the property can be ignored. We have seen some papers classifying all primes with a certain property. Then a warning is issued that primes with this property should be avoided by the key generation program. However, in most situations, any phenomenon that can be classified in this manner is so unlikely to occur that there is no need to worry.
If the propability in question is not insignificant, it will be necessary to avoid primes with that kind of property in the key generation. As an example, it can be estimated that the propability that a randomly chosen prime of 256 bits is strong as an RSA key is not sufficiently large, whereas a prime of 512 bits is strong with sufficiently high probability. (See [1])

An independent problem which needs attention is that of a sufficiently good random generator. However, this must be solved as a separate and equally important issue, which has nothing to do with prime generation.

[1] J. Brandt, I. Damgaard and P. Landrock, "Speeding up up prime number generation," Abstracts of ASIACRYPT'91, Fujiyoshida, Japan.