Factoring Report

2001年12月4日

MEC Consulting (communicated via RSA Security)
Dr.Preda Mihailescu

Factoring Report

Dr. Preda Mihailescu
MEC Consulting – Seestr. 78,

8700 Erlenbach Zürich
Email: preda@upb.de

December 14, 2001

1. Purpose and Outlook

The purpose of this report is to evaluate the impact of state of the art factoring algorithms on cryptographic techniques based on the difficulty of factoring products of three primes, two of which may be equal. Products of the type $N=p^2q$ will be considered in most detail. Some generalizations to products of more than three prime factors will be made, thus responding to a facultative question of the task proposal; however the most significant information is revealed by the analysis of the case of three factors.

The structure of the report is as follows. In the first part we review the major state of the art methods of factoring, under consideration of both their theoretical properties and level of investigation and of the known implementations and their relative spread. We also show that the special form $N=p^2q$ brings some less investigated methods, based on arithmetic of imaginary quadratic extensions or continued fractions, to the foreground.

In the second part we discuss some current prognoses about the evolution of the art of factoring together with some recent special purpose architectures which have been proposed for enhancing the speed of factorization. This topics are considered with the due caution, while the main concern consists in distinguishing whether yes or not, under the assumption of these various prognoses and developments of the computation tools, the numbers of concern present special vulnerabilities with respect to general RSA moduli.

The report closes with conclusions and suggestions based upon the material developed in these two parts.

Certainly, the report was developed on the base of a far wider literature than suggested by the requester. This encompasses academic text books, research reports, implementation reports and a variety of specialized internet sites. We have included a short bibliography which, rather than being extensive, was conceived such as to provide the reader a sufficient base for the search tree to the literature we have used. It is a conceptional decision, to keep the text as untechnical as possible, in the natural assumption that the reader is sufficiently familiar with the topic for following the qualitative argumentation with the help of the additional references.

2. Target composites of the shape $N=p^2q$.

The quadratic factor of the target numbers of this report is definitely the particularity which requires special attention. It is thus important, even before discussing factoring to review some elementary consequences of the existence of such a quadratic factor.

2.1 Continued fractions.

The continued fraction development of quadratic irrationalities, i.e. of solutions of quadratic polynomial equations, and in particular square roots of integers, is long known to be periodic [Pe]. Furthermore, for integers of the investigated type, the periods of the continued fraction development of $\sqrt{N}$ and of $\sqrt{q}$ are equal up to permutation. It is thus sensible to expect to gather some information about q from the continued fraction development of $\sqrt{N}$, which can be expanded starting from known data. However, the classical bounds on the length of these periods are exponential and although they are known to be shorter in practice, it was not possible to find some modern complexity theory investigation of the expected value of the period of continued fractions of quadratic roots. It is conceivable that safety still resides in the length of the periods, the issue might though require some more in depth investigation, which was beyond the frame of this brief report.

2.2 Quadratic extension fields

The fundamental discriminant D of a quadratic extension field $\mathbf{K} = \mathbf{Q}(\sqrt{(\pm N)})$ is (up to some possible factor 4) equal to the *square free* part of N. In the present case thus $D \sim q$. Important properties of the field, such as for instance the size of the class group $C(\mathbf{K})$ depend on the size of D – and certainly not of N.

3. State of the Art Algorithms for Factoring

The problem of factoring an integer number N consists in providing a method for finding a non trivial factor $m \mid N$, in a way which can be iterated such as to recover eventually the full prime number factorization of N. The performance of such methods is measured in terms of the length of N, which we shall denote here, for simplicity by $n = \log(n)$. Note that we use the natural logarithm, so n is proportional but not equal to the bit length. The following function

$$(1) \quad L_n(a,c) = \exp(c \,.\, n^a \,.\, log(n)^{1-a})$$

is frequently occurring as a measure for the performance of the various subexponential factoring methods. Here a, c are constants and $0 < a < 1$. Note that performance increases with decreasing values of a and/or c. We shall write shortly $L(n) = L_n(1/2,1)$.

Most of the algorithms which we shall discuss use in different ways y-smooth numbers, i.e. integers x which split into prime factors p which are all $p < y$. Their practical relevance stems from the following seminal theorem of Canfield, Erdos and Pomerance: [Co, p. 482]

If $\psi(x,y)$ is the number of y-smooth numbers $< x$, then

$$(2)\ \psi(x, L(\exp(x))^{b}\ ) = x\ L(\exp(x))^{-1/2b+o(1)}, \quad \text{for any } 0 < b < 1.$$

As a direct consequence, setting $b = \surd 2$, a number of length n is $L(n)^{b}$ – smooth with probability $L(n)^{b}$. [C]

3.1 The continued fractions method,

is the first among the subexponential factoring methods. If N is the integer to be factored, the factoring idea consists in finding positive integers $x, y < N$ such that

$$(3)\ x^2 = y^2 \bmod N.$$

If such a pair is found, then N divides $(x + y)(x - y)$ and a non trivial factor of N is found by building the greatest common divisor $(N, x\text{-}y)$. The congruent squares are sought among smooth numbers of the form $r(x) = x^2 \bmod N$. After having gathered sufficiently many such smooth relations, a congruence (3) is found by linear algebra. Continued fractions yield good candidates for x, while the relation (2) helps measure the odds of finding the relation (3). Based on such estimates, it turns out that the expected time for factoring a number of length n with the continued fractions method is $L(n)$.

3.2 The quadratic sieve algorithm

also uses congruences of the type (2) as factoring method. It differs from the continued fractions method in the fact that these relations are found by a sieving technique which is superior to the continued fractions approach. The asymptotic expected running time of the quadratic sieve is $L(n)$, the same as for the continued fractions method. However, the fact that the single steps required for building up the relations are very simple and performant for the first, makes that this algorithm is preferable to the latter.

The quadratic sieve has been intensively investigated, generalized and improved upon since it was proposed by C. Pomerance, in 1983. It has the property of factoring any numbers of given size with like probability. As we already mentioned, the asymptotic behavior does not identify alone the performance of a factoring method; the simplicity of its building blocks has an important influence. Thus, among various algorithms with runtime $L(n)$, the enhancements of the quadratic sieve method are the most performant all purpose factoring algorithms.

3.3 The p-1 method

is rather special and works for the case that the number N to be factored has some prime factor p, such that p-1 is a B - smooth number, for some integer B typically of the order of magnitude of $L(n)$. If this is the case, one lets $K=\prod_{q<B} q$ be the product of all prime powers $q < B$ and chooses a base a mod N, computing then $A = a^K$ mod N. With probability close to 1, the greatest common divisor (A-1, N) will then reveal a non trivial factor of N. The odds for the number p-1 being smooth are in general vanishing, and therefore the p-1 method is only accidentally successful. It is however the simplest illustration of the general idea of *subgroup factoring methods*, which has more practical exponents.

3.4 The (quadratic imaginary) class group method,

is a subexponential improvement of older ideas of Shanks and Pollard and was discovered independently by Atkin and Rickert and published by Lenstra and Schnorr. If N is the number to be factored, the factoring method considers the class group of imaginary quadratic extensions $\mathbf{Q}((-kN)^{1/2})$, for *small* integers k. Again, this is done in order to find values of k for which the order of the class group is smooth. If this is the case, one uses the subgroup factoring idea in order to find nontrivial factors of N. By choosing various values of k, one generates independent class groups, and thus bypasses the problem encountered in the p-1 method.

If $D \mid kN$ is the fundamental discriminant of the field $\mathbf{Q}((-kN)^{1/2})$ and n is its length, then the class group method runs in time $L(n)$. As a general purpose algorithm, this is thus comparable to, yet less efficient then the quadratic sieve. It might however be **particularly dangerous** method for numbers of the shape $N = p^2q$. As we have shown in 2.2, in this case the fundamental discriminant is essentially $D \sim q$ and thus the security of the modulus N is measured by the size of q rather than the size of N ! In such a case, the class group factoring method would be likely to be by far the most dangerous method against the investigated moduli, although it is otherwise currently a rather marginally used method. However, the method works for *fundamental discriminants* N (i.e., N square free); if N is not square free, the method leads to computations in a *ray class group* rather than the class group of the field itself. The ray class group has an order which is divisible by $p\pm1$ and has thus no particular reasons to become smooth. In fact, smooth values of $p\pm1$ are a trap door to elementary factoring methods (which incidentally bear the same name). They are exponentially rare in the range of primes of fixed size. One can explicitly avoid them (Gordon strong primes) or rely upon the low odds of generating such a prime by accident. In the latter case the threat is no higher for the moduli under investigation than for classical RSA moduli.

Although the class group factoring method in its state of the art version is not suited for factoring $N=p^2q$, it points out to a suite of rich additional structure which might be used for special purpose factoring methods in this context. It is for instance conceivable, that the information about ray class groups may be elaborated up to a factoring method.

3.5 The Elliptic Curve Factoring Method (ECM)

is the most powerful subgroup factoring method. It uses the group structure of elliptic curves modulo the number N to be factored. Just like the class group method, it draws back on a large reserve of potential groups, having thus the necessary odds for finding some group of smooth order. This is done by trial and error and once a smooth group is encountered, non trivial factors of N are brought to evidence in a manner similar to the one described in 3.3. The particular advantage of ECM consists in ability to find prime factors p of size l in time $L(l)^d$, with $d = \sqrt{2}$. The time for finding the first factor thus depends on the factor alone and not on the product which is input to the algorithm. It is herewith the most performant algorithm currently known for finding small and moderate sized prime factors irrespective of the size of the number to be factored. As a general purpose algorithm it is however less practical than the quadratic sieve, with which it shares the same asymptotic behavior.

The following two remarks are important in our context:

- the elliptic curve method has increasing chances of success with growing number of factors of N. This holds irrespective of the fact if some prime factor is repeated - thus occurring as a prime power dividing N - or not - in which case N is square free.

- few improvements of the basic idea of ECM are known; the most popular is Montgomery's FFT multiple curve variation. For primes of the shape $N=p^2q$, Peralta and Okamoto have devised a dedicated variant, which improves slightly the performance of ECM in this case, without modifying its asymptotical behavior.

We shall take these remarks into account in the next chapter. There we shall compare directly the odds of success of the various factoring methods presented, when attempting to break moduli of cryptographic sizes.

3.6 The generalized number field sieve (NFS),

is a powerful factoring method for which the first constant in (1) is a = 1/3. Since in all previously discussed cases, a = 1/2, the NFS is asymptotically much faster. However, the second constant c = 1.92... is larger than for the previous cases. We shall discuss in the next chapter issues concerning the estimation of the trade-off size beyond which NFS can be expected to be the most performant general purpose factoring method.

3.7 The lattice reduction method for $N=p^rq$,

was discovered by Boneh et. al. [BDH] and it indicates that composites of this special form can be factored efficiently also by methods which do not rely on smoothness. Its practical relevance as factoring method is reduced: assuming that p and q are primes of length n, the method is better than ECM only for $r > \sqrt{n}$. If n is the length of 512 bit integers, then $r > 23$, which is beyond any realistic context. The lattice reduction method in itself need thus not be considered as any threat for factoring composites of the investigated shapes. However, a side result of the paper [BDH] shows that moduli $N=p^2q$ are more sensitive with respect to bit leaking than plain RSA moduli. This information is of certain relevance for our report.

4. Implementations and Prospectives

Making midrange predictions about the security of certain key lengths against future factorization attacks is a task at which it happened that even very important researchers failed, see [Od]. Given the necessity for estimating such evolutions in spite of all, forecasts continue to be made. It will help the task of this report to make a brief review of current forecasts about the security of key lengths for cryptographic algorithms.

4.1 Forecasts

In 1999, Lenstra and Verheul published [LV] a detailed analysis of state of the art implementations of "attacks" against the different number theoretical hard problems upon which cryptographic primitives are based. The paper culminates with a long term table of predicted safe key lengths, with respect to the hard problems of factoring, discrete logarithms in multiplicative groups of finite fields and in subgroups thereof, and the elliptic curve discrete logarithm. It was rightly observed by Silverman [Si] that the table was too optimistic for the evolution of the NFS when applied to factoring: time had been the main concern and too little attention was paid to the problems of space, bus delay etc., which may become prohibitive. It is sufficient for the purpose of our report to point out to the debate, noting that the algorithms taken in consideration are basically *unable to distinguish between RSA moduli and composites of more, eventually repeated factors*.

Two further valuable overviews of the state of the art of the factoring and discrete logarithm problems are [Br] and [Od]. Their conclusions are more qualitative. It is important to underline that by today, the multiple quadratic sieve, ECM and the NFS are the most widely spread factoring methods. Consequently, more implementational improvements, which do not modify the asymptotic behavior but bring about valuable constant performance enhancements, were discovered for each and any of these. For these reasons, it is hardly possible to compare, for instance, the class group method with ECM on a base different from their asymptotical behavior.

4.2 Implementations and ECM/NFS trade-off.

Based on this premise, when considering multiprime moduli, the immediate practical problem is the trade -- off point between ECM and NFS. Certainly, for a fixed size of *m* bits, the more prime factors the modulus *N* has, the smaller these are and the higher the advantage of ECM. However, since NFS is asymptotically faster, this advantage will eventually drop for any *fixed* number *k* of factors, when *m* grows to infinity. On the other hand, unlike the class group factoring method, there is *no indication that ECM may distinguish between inputs with repeated factors and square free inputs*. For ECM, products $N = p.q.r$ of three prime factors and products $N=p^2q$ of a square of a prime by a second prime are indistinguishable and the trade-off point is the same. The NFS record for factoring general numbers lays currently at 512 bits, while the ECM record is of finding a prime factor of 187 bits [RR] (caveat: different from factoring a 187 bit integer !); it was achieved by using roughly 1/50 of the computing power used in the NFS record: NFS challenges are highly social events in which large networks collaborate over the Internet, while

ECM factoring is still made by individuals using local resources or at most some small Internet collaboration.

Extrapolating these data is easy: if *A* is the amount of work for factoring out an 220 prime with ECM and *B* the one for factoring a 512 integer with NFS, we set *A* ~ *B*, which corresponds to the empirical data [RR]. A 220.*c* bit prime will then be factored out with ECM in $t_1= A^{c^\wedge(1/2)}$ while a 512.*d* bit integer will be factored with NFS in $t_2=B^{d^\wedge(1/3)}$. Setting $t_{1=}$ t_2, i.e. asking for the values of *c* and *d*, such that the ECM and NFS factoring *times* be equal, one finds (given *A* ~ *B*) that $d = c^{3/2}$. The security of a 1024 bit product against NFS is thus given for *d* = 2, the trade off is found when both lengths and times are equal. For instance, for products of three primes, 3.220.*c* = 512.*d* leads to $c = (660/512)^2$ etc.

One estimates that with the current performance of the best known implementation and an equal computing time, ECM could find a prime factor of about 220 bits. This shows that *three primes products of 768 bits* are as of today *highly unsafe* against ECM (660 bit three primes moduli can be regarded as state of the art, while only 512 bits products of two primes are so, with NFS). Two prime products of the same size, while still unrecommended, are still beyond reach for NFS. Rough extrapolations show that the trade – off is given by the following table.

Nr. of factors in key	**Key Size below which ECM is faster than NFS**	**Key Size for security of factoring 1024 bit with NFS**
3	1104	1048
4	2600	1400
5	5080	1752
6	8776	2096

The sizes are rounded up to multiples of 8, for simplicity. The first column indicates the trade – off key size for a fixed number of factors, while the second gives the size of a *k*-factored key for which an ECM attack has comparable time requirements as an NFS factorization of a 1024 composite. Note that unlike [Si], the space requirements were not taken into consideration; they would increase the values in the second column, since space is no problem for ECM, but it is one for NFS. There is a difference of 5-10% between our estimate and Silverman's [Si]. The reason is that we chose a crude empiric approach, based on extrapolating the data extracted from individual records, while [Si], and also [LV] develop their estimates on the base of complex theoretical predictions about hardware costs and performance.

For instance, we are essentially stating that factoring out a 350 prime with ECM takes the same time as factoring a 1024 integer with NFS, which is subject to further analysis. The two approaches lead to relatively close predictions. We consider for more than one reason this rough extrapolation to be sufficiently suggestive in the current context. For practical purposes, its implication is that at 1024 bit key size, products of three primes are still slightly less secure than classical RSA moduli. The difference is tolerable and balanced risks are found beyond roughly 1100 bits. More precise extrapolations can be provided upon request.

4.3 Special purpose devices.

A new direction in factoring consists in the prospection of dedicated hardware for factoring. Certainly, the quantum computer is the ultimate, yet, according to any serious information, far away Damocles sword: sufficiently many stable quantum bits are known to allow breaking any cryptogrpahic scheme based upon commutative algebraic number theory in polynomial time.

Shamir´s **TWINKLE** is a light diode based variation of an old mechanical special purpose computer of Lehmer. It would help speed up the relation gathering step for the quadratic field sieve, so that according to Shamir, 512 bit integers could be factored with ease. Hardware specialists and engineers pointed out that the density of diodes required by the TWINKLE is hard to achieve; it is a fact that 5 years after the idea has been made public, no physical realization has been made.

Daniel Bernstein has recently published [Be] a new idea for a **dedicated parallel computing architecture**. According to the author, the architecture is suited for accelerating the main steps -- search for smooth numbers and solving of large sparse linear systems -- of most sieving methods used for factoring and discrete logarithms. This includes the NFS and also the class group factoring. Bernstein promises that his machine could factor for the same cost numbers three time as large as may be done with current architectures. Bernstein´s idea has the nice property that it can be tested on current field programmable gate arrays (FPGAs) and thus no extravagant investment or sophisticated hardware development is required in order to verify the validity of the idea. It is rather unfortunate however, that his seminal paper develops the asymptotic improvements expected from the new architecture up to many decimal digits of the constants in the exponents (c, in (1)); it avoids yet to address any simple questions concerning the tasks which should be completed in order to be able to run some experiments for very moderate sized integers -- say 512 bits or even less. In lack of such statements it is quite hard to estimate -- especially for this reviewer, who is far less than a computer architecture expert -- how serious a threat this idea may be for the next future. Bernstein´s exposition is sufficient however for understanding that, if any improvement of factoring comes along his ideas, this will *equally impact upon RSA moduli and multi -- factored* ones.

5. Exploiting the square factor in $N= p^2.q$

As observed in section 2, these particular shape of moduli leads to classical domains of mathematics, such as continued fraction expansions of quadratic irrationalities, class groups and ray class groups of imaginary quadratic fields, modular forms, etc. This report is dedicated to a research of the state of the art and we have shown that even the class group method 3.5, in its existing form can not take advantage of the particular shape of N. However there is a very rich mathematical background to draw back upon, when trying to develop dedicated methods for factoring . The mathematical research of the topic is close to null.

Fifteen years after the spreading of the elliptic curve cryptography, Odlyzko writes in [Od]: "Although elliptic curve cryptosystems are becoming more widely accepted, they are still regarded with suspicion by many ... What worries elliptic curve skeptics is that there is much mathematical structure that could potentially be exploited". The statement can be fairly used for this kind of moduli. They have certainly been less investigated during the last 10 years than the elliptic curve discrete logarithm has. Since, quoting again Odlyzko, like for elliptic curves, "... the deep mathematics that is required to work [on this factoring problem] reduces the pool of people qualified to examine it", the incitement given by suite of factorization **challenges might help trigger the research**. It is certainly *not a coincidence*, that general factoring and elliptic discrete logarithms and point counting have been subject to much more intense computing efforts and internet collaborations than various instances of the discrete logarithm in finite fields. The first two problems are object of challenges supported by major cryptographic companies, why the latter is not. The same holds for factoring integers of the shape $N=p^2q$.

Negative results are scarce in mathematics. Given the importance that elliptic curves have for cryptography, an analysis of the reasons why successful ideas of the index calculus are *not likely* to be useful for the discrete logarithm for these particular groups was published nevertheless by Silverman and Suzuki [SS]. Triggering a similar investigation might be a welcome side effect of an eventual factorization challenge for moduli and could *help increase the confidence in their security*. At the time of this report, one can only note that there is no known threat. Yet hardly any research can be found regarding this instance of square free factorization; meanwhile the mathematical ideas which could be used are very rich. It is important to underline that the ideas we refer to apply specifically to these moduli and could result in dedicated factoring methods which do not apply to square free numbers. On the other hand, any improvement of general factoring method trivially applies to $N=p^2q$. It is thus fair to conclude that the **state of the art reveals no immediate threat, yet a higher risk potential than in the general (square free) case**. Triggering the research on square free factorization should be one useful means for establishing the security of cryptographic schemes based on the difficulty of factoring $N=p^2q$. In the case that no dedicated algorithm for factoring these moduli is found and maybe some arguments are given, which suggest reasons why it should not be possible to find such an algorithm, the confidence in the cryptographic schemes under debate would certainly raise. This report could however not find any indication of such arguments; on the contrary, we consider an increased research on square free factorization as an important topic of the context.

6. Conclusions

The analysis above leads to the following conclusions concerning the cryptographic security of multi -- factor moduli N:

- ECM is no special threat, if the moduli are chosen large enough. Current state of the art sizes (1024 bits) are sufficient for three factors moduli. Parity of risks between two and three factor moduli settles roughly at 1256 bits.

- The lattice reduction method of Boneh et. al. is indicative of a particular sensitivity of moduli $N=p^2q$ with respect to bit leaking. Protocols designed on base of such moduli should thus be analyzed with particular attention as to bit leaking. Bit leaking is no special problem for square free multiprime moduli. Factoring is no realistic threat with this approach.

- Square free moduli and such ones with repeated factors have essentially the same risk with respect to state of the art factoring methods (at the same number of factors and the same length, naturally).

- Special purpose factoring devices are likewise dangerous for two- and multi factored moduli, square free or not. Whether or not they can successfully be developed is thus irrelevant to distinguishing the cryptographic security of the two types of moduli.

- The class group factoring method is not a threat to moduli of the type $N=p^2q$ in its current version. It indicates however a direction for potential dedicated factoring algorithms for this special form of moduli. Quadratic imaginary fields, quadratic and modular forms, continued fraction developments of quadratic irrationalities, are mathematical subjects with very rich structure, which all can potentially use the specific shape under consideration. We estimate that there is insufficient research in the direction of applying any of these fields for dedicated factorization. This fact leads to a **higher risk potential for these moduli**.

Literature:

[Be] D. Bernstein: Circuits for integer factorization: A proposal, Preprint, 11/09/2001.

[BDH] D. Boneh, G. Durfee and N. Howgrave - Graham: Factoring $N = p^rq$ for Large r, Advances in Cryptology, CRYPTO'99, Springer, LNCS **1666** (1999), pp. 326-337.

[Br] R. Brent: Recent Progress and Prospects for Integer Factorization Algorithms, COCOON 2000, D.-Z. Du et. al. (eds.), LNCS **1858**(2000), p. 3-22.

[Co] H. Cohen: A Course in Computational Algebraic Number Theory, Springer Verlag (1996), Graduate Texts in Mathematics **138**.

[FMO] A. Fujioka, S. Miyaguchi and T. Okamoto: ESIGN: An Efficient Digital Signature Implementation for Smart Cards, in Advances in Cryptology, EUROCRYPT'91. Springer (1992), pp. 446-457.

[LL] A. K. Lenstra and H. W. Lenstra Jr.: Algorithms in Number Theory in Handbook of theoretical computer science, J. van Leeuwen et. al. (eds.), Elsevier, Amsterdam (1990), pp. 673-715.

[LS] H. W. Lenstra Jr. and C. P. Schnorr: A Monte-Carlo Factoring Algorithm with Linear Storage, Math. Comp **43** (1984), pp. 289-312.

[LV] A. Lenstra and E. Verheul: Selecting cryptographic key sizes, preprint, available from http://www.cryptosavvy.com

[MOV] A. Menezes, Paul C. van Oorschot and S. Vanstone: Handbook of Applied Cryptography, CRC Press (1997)

[Od] A. Odlyzko: Discrete Logarithms: The past and the future, Designs, Codes and Cryptography **19** (2000), pp. 129-145.

[OP] E. Okamoto and R. Peralta: Faster Factorng of Integers of a Special Form, IEICE Trans. onf Fund. Of Electronics, Communications and Comp. Sci. **E79-A, 4** (1996)

[OS] T. Okamoto and A. Sirashi: A Fast Signature Scheme Based on Quadratic Inequalities, Proc of the ACM Symp. on Security and Privacy, ACM Press (1985)

[Ok] T. Okamoto: A Fast Signature Scheme Based upon Congruential Polynomial Operations, IEEE Trans. on Inf. Theory, **IT-36, 1** (1990), pp. 47-53.

[RR] A site of factoring records, http://www.crypto-world.com/FactorWorld.html

[Si] R. Silverman: A cost – based security analysis of symmetric and asymmetric key lengths, Bulletin 13, RSA Laboratories, Bedford MA, http://www.rsasecurity.com/rsalabs/bulletins/

[SS] J. H. Silverman and J. Suzuki, pp. 110-125, Advances in Cryptology - ASIACRYPT´98, LNCS **1514**, (1998).