

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Survey of RSA Vulnerabilities

Anthony Overmars

Abstract

Rivest et al. patented (US) RSA. RSA forms the basis of most public encryption systems. It describes a public key encryption algorithm and certification process, which protects user data over networks. The patent expired in September 2000 and now is available for general use. According to Marketsandmarkets.com, the global network encryption market size is expected to grow from USD 2.9 billion in 2018 to USD 4.6 billion by 2023, at a compound annual growth rate (CAGR) of 9.8%. Major growth drivers for the market include increasing adoption of optical transmission, an increasing demand to meet various regulatory compliances and a growing focus on shielding organizations from network security breaches. In short, RSA forms the basis of almost all public encryption systems. This, however, is not without risk. This chapter explores some of these vulnerabilities in a mathematical context and provides the reader with an appreciation of the strength of RSA.

Keywords: survey, public keys, vulnerability

1. Introduction

Rivest et al. patented (US) RSA, which forms the basis for most public encryption systems. RSA describes a public key encryption algorithm and certification process, which protects user data over networks. The patent expired in September 2000 and now is available for general use. According to Marketsandmarkets.com [1], the global network encryption market size is expected to grow from USD 2.9 billion in 2018 to USD 4.6 billion by 2023, at a compound annual growth rate (CAGR) of 9.8%. Major growth drivers for the market include increasing adoption of optical transmission, an increasing demand to meet various regulatory compliances and a growing focus on shielding organizations from network security breaches. In short, RSA forms the basis of almost all public encryption systems. This, however, is not without risk. This chapter explores some of these vulnerabilities in a mathematical context and provides the reader with an appreciation of the strength of RSA.

RSA is secure and difficult to factorize in polynomial time. Conventional sequential computing machines, running in polynomial time, take an unfeasible amount of CPU cycles to find factorization solutions to RSA keys. Quantum computing holds great promise; this, however, is realistically still some way off. Opportunities exist using conventional computing (sequential and parallel) using better mathematical techniques. A discussion on exploiting implementation flaws is also considered.

Of keen interest is our lack of understanding of prime numbers and their structure. The current perception is that there appears to be some underlying structure, but essentially, primes are randomly distributed. This is explored in Sections 8 and 12.

Vulnerabilities in the selection of primes are exploited in Section 5 using Euler’s factorization.

Poor RSA key design and their exploits are considered in Section 6 using Wiener’s method and in Sections 15–17 using a combination of LLL, Coppersmith and Pohlig-Hellman. All of these attacks can be mitigated by designing the RSA keys with these exploits in mind. RSA key design (Section 2) consists of two parts, a private key (N, d) and a public key (N, e) . A composite number N , is derived from two prime numbers. The (d, e) numbers are selected in an ad hoc manner using Euler’s totient.

Development of quantum computing is continuing at breakneck speed; however useful machines are yet to appear. Parallel computing however is here and now, and whilst factorizing RSA keys is not achievable on conventional computers in polynomial time, parallel computing has allowed for multiple solutions to be tested simultaneously. This is an area where research continues and new algorithms as shown in Sections 20 and 14 lend themselves well to GPU parallel processing systems.

2. Structure of RSA numbers

Consider RSA100 challenge number

$$\begin{aligned} \text{RSA} - 100 &= 152260502792253336053561837813263742971806811496138 \\ &\quad 0688657908494580122963258952897654000350692006139 \\ &= 37975227936943673922808872755445627854565536638199 \\ &\quad \times 40094690950920881030683735292761468389214899724061 \end{aligned}$$

RSA100 is a 100 binary bit number made up of two 50 binary bit prime numbers. The motivation in breaking this composite number allows us to find the Euler’s totient number φ_n . Once this is known, using the public key $P_U = (N, e)$, it is possible to derive the private key $P_R = (N, d)$, and hence all cypher-text encrypted (e) messages can thus be decrypted back to plain text, using (d).

3. A simple RSA encryption/decryption example

Using two primes P_1 and P_2 to generate a composite number N ,

$$N = P_1P_2 = (1462001 \times 1462009) = 2137458620009$$

Totient φ (Euler’s totient function)

$$\text{Calculate totient } \varphi_n = (P_1 - 1) (P_2 - 1) = (1462001 - 1) (1462009 - 1) = 2137455696000$$

Arbitrarily choose a public key such that e is an integer, not a factor of $\text{mod } N$, and $1 < e < \varphi$, $e = 13$

The public key is made up of N and e , such that

$$P_U = (N, e) = (2137458620009, 13). \text{ A private key is made up of } N \text{ and } d, \text{ such that } P_R = (N, d) = (2137458620009, d).$$

d , is determined using the extended Euclidean algorithm.

$$e d \text{ mod } \varphi_n = 113 d \text{ mod } 2137455696000 = 1 \Rightarrow d = 1973036027077.$$

Therefore, private key, $P_R = (N_1, d) = (2137458620009, 1973036027077)$.

Encrypt a message m , into cipher text C , with public key P_U . Let the message $m = 1461989$. $C = m^e \bmod N = 146198913^{13} \bmod (2137458620009) = 1912018123454$. To recover the original message, decrypt using Private Key, $P_R = (N, d) = (1912018123454, 1973036027077)$ $m = C^d \bmod N = 1912018123454^{1973036027077} \bmod(2137458620009) = 1461989$.

From this simple example, consider the following: How can we use a known public key $P_U = (N, e)$ to decrypt the original message? To decrypt the message, the private key is used: $P_R = (N, d)$. How can d , be discovered? d is derived using Euler's totient function $[\varphi_n = (P_1 - 1) (P_2 - 1)]$, and the extended Euclidean algorithm $ed \bmod \varphi_n = 1$. However when a public key is transmitted, the totient φ_n and the two primes P_1 and P_2 remain secret. If φ_n , P_1 or P_2 can be determined, the private key will be compromised and the cypher-text will no longer be secure.

When the totient φ_n is known, d can be determined through the normal key generation processes, so the determination of the two primes (P_1, P_2) is not required to recover the message from the cypher-text. The following proof is provided for completeness and shows how the two primes P_1, P_2 can be recovered if the composite N and the totient φ_n are known.

4. If the composite N and the totient φ_n are known, the original primes can be recovered

The quadratic formula can be used to find P_1 and P_2
 $\varphi_n = (P_1 - 1)(P_2 - 1)$, $N = P_1 P_2$. General quadratic form: $ax^2 + bx + c = 0 \Rightarrow$
 $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

$$\varphi_n = (P_1 - 1) (P_2 - 1) = P_1 P_2 - P_1 - P_2 + 1 \text{ recalling } N = P_1 P_2 \Rightarrow \varphi_n = N - P_1 - P_2 + 1$$

Express primes in terms of N, φ_n $P_1 = N - \varphi_n - P_2 + 1$, $P_2 = N - \varphi_n - P_1 + 1$ $N = P_1 P_2$
 substitute for $P_2 \Rightarrow N = P_1 (N - \varphi_n - P_1 + 1) = P_1 N - P_1 \varphi_n - P_1^2 + P_1$

$$P_1^2 + P_1 (\varphi_n - N - 1) + N = 0 \quad ax^2 + bx + c = 0 : a = 1, b = (\varphi_n - N - 1), c = N, x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$P_1, P_2 = \frac{-(\varphi_n - N - 1) \pm \sqrt{(\varphi_n - N - 1)^2 - 4(1)N}}{2(1)} = \frac{-(\varphi_n - N - 1) \pm \sqrt{(\varphi_n - N - 1)^2 - 4N}}{2}$$

When N and φ_n are known: $N = 2137458620009$, $\varphi_n = 2137455696000$

$$P_1, P_2 = \frac{2924010 \pm \sqrt{8549834480100 - 8549834480036}}{2} = \frac{2924010 \pm \sqrt{64}}{2} = 1462005 \pm 4$$

$$P_1, P_2 = (1462001, 1462009)$$

Using the quadratic formula, P_1 and P_2 can be recovered if the composite N and the totient φ_n are known.

5. Fermat's factorization method

$N = a^2 - b^2 = (a - b)(a + b)$ is the difference of two squares.

$$P_1 = a - b, P_2 = a + b, P_1 + P_2 = 2a, P_2 - P_1 = 2b; a = \frac{P_2 + P_1}{2}, b = \frac{P_2 - P_1}{2}$$

$$N = a^2 - b^2 = \left(\frac{P_2 + P_1}{2}\right)^2 - \left(\frac{P_2 - P_1}{2}\right)^2 = \frac{1}{4} \left((P_2 + P_1)^2 - (P_2 - P_1)^2 \right) = P_1 P_2$$

As the first trial for a , $a_1 = \sqrt{N}$, then check if $\Delta a_1 = a_1^2 - N$ is a square number.

There are only 22 combinations of which the last two digits are a square number. The other 78 can be eliminated.

If Δa_1 is not a square number, then $a_2 : a_2 = a_1 + 1$.

$$\text{Now } \Delta a_2 = a_2^2 - N \implies (a_1 + 1)^2 - N = a_1^2 - N + 2a_1 + 1 = \Delta a_1 + 2a_1 + 1$$

$$\Delta a_3 = a_3^2 - N \implies (a_2 + 1)^2 - N = a_2^2 - N + 2a_2 + 1 = \Delta a_2 + 2(a_1 + 1) + 1 = \Delta a_2 + 2a_1 + 3$$

$$\Delta a_4 = a_4^2 - N \implies (a_3 + 1)^2 - N = a_3^2 - N + 2a_3 + 1 = \Delta a_3 + 2(a_1 + 2) + 1 = \Delta a_3 + 2a_1 + 5$$

so the subsequent differences are obtained by adding two.

Consider the example $N = 2137458620009$.

$$a_1 = \sqrt{N}, a_1 = \sqrt{2137458620009} \implies a_1 = 1462005$$

Check if $\Delta a_1 = a_1^2 - N$ is a square number.

$$\begin{aligned} \Delta a_1 = a_1^2 - N &= 1462005^2 - 2137458620009 = 2137458620025 - 2137458620009 = 16 = 4^2 \\ N &= 1462005^2 - 4^2 = (1462005 - 4)(1462005 + 4) = (1462001)(1462009) \end{aligned}$$

Maurice Kraitchik, a Belgian mathematician, considered only values of a and $b : a^2 \equiv b^2 \pmod{N}$.

$$a^2 \equiv b^2 \pmod{N} \implies \Delta 1462005^2 \pmod{2137458620009} \equiv 16$$

6. Euler's factorization method

Gaussian primes are of the form $4x - 1$, and primes of the form $4x + 1$ are Pythagorean. Fermat's Christmas theorem on sum of two squares states that an odd prime can be expressed as $P = x^2 + y^2$ iff $P \equiv 1 \pmod{4}$.

Gaussian primes are of the form $P \equiv 3 \pmod{4}$ and are not representable as the sum of two squares.

Consider a composite number $N: N = P_1 P_2$ and $P_1: P_1 = a^2 + b^2$,
 $P_2: P_2 = c^2 + d^2$.

$$N = P_1 P_2 = (a^2 + b^2)(c^2 + d^2) = (ac)^2 + (bc)^2 + (ad)^2 + (bd)^2$$

$$\text{let } A^2 = (ac)^2 + (ad)^2, B^2 = (bc)^2 + (bd)^2, C^2 = (ac)^2 + (bc)^2, D^2 = (ad)^2 + (bd)^2$$

$$N = P_1 P_2 = (a^2 + b^2)(c^2 + d^2) = (ac)^2 + (bc)^2 + (ad)^2 + (bd)^2 = A^2 + B^2 = C^2 + D^2$$

$$N = A^2 + B^2 = C^2 + D^2 \implies A^2 - C^2 = D^2 - B^2$$

$$A^2 - C^2 = D^2 - B^2 \implies (A - C)(A + C) = (D - B)(D + B)$$

$$P_1 = \left(\frac{\gcd(A-C, D-B)}{2}\right)^2 + \left(\frac{\gcd(A+C, D+B)}{2}\right)^2,$$

$$P_2 = \left(\frac{\gcd(A+C, D-B)}{2}\right)^2 + \left(\frac{\gcd(A-C, D+B)}{2}\right)^2$$

Consider the example $N = 2137458620009$; find the factorization values of P_1 and P_2 .

Using the sum of squares, $N = 2137458620009 = 324403^2 + 1425560^2 = 643603^2 + 1312720^2$.

Combining the even and odds: $1425560^2 - 1312720^2 = 643603^2 - 324403^2$.

$$A^2 - C^2 = D^2 - B^2 \Rightarrow (A - C)(A + C) = (D - B)(D + B) = (968006)(319200) = (2738280)(112840)$$

Using the greatest common divisor (gcd):

$$\frac{\gcd(A - C, D - B)}{2} = \frac{\gcd(968006, 2738280)}{2} = 1201, \quad \frac{\gcd(A + C, D + B)}{2} = \frac{\gcd(319200, 112840)}{2} = 140$$

$$\frac{\gcd(A + C, D - B)}{2} = \frac{\gcd(319200, 2738280)}{2} = 1140, \quad \frac{\gcd(A - C, D + B)}{2} = \frac{\gcd(968006, 112840)}{2} = 403$$

$$P_1 = \left(\frac{\gcd(A-C, D-B)}{2}\right)^2 + \left(\frac{\gcd(A+C, D+B)}{2}\right)^2 = 1201^2 + 140^2 = 1462001$$

$$P_2 = \left(\frac{\gcd(A+C, D-B)}{2}\right)^2 + \left(\frac{\gcd(A-C, D+B)}{2}\right)^2 = 1140^2 + 403^2 = 1462009$$

7. Wiener attack

Wiener's theorem. Let $N = P_1P_2$ and $P_1 < P_2 < 2P_1$ and a private key $P_R = (N, d)$ and a public key $P_U = (N, e)$. Let $d < \frac{1}{3}N^{\frac{1}{4}}$, given a public key $P_U = (N, e)$, with $e d \equiv 1 \pmod{\varphi_n}$. The attacker can efficiently recover d [2]. The attack uses the continued fraction method to expose the private key d , when d is small. It assumes $\frac{e}{N} \approx \frac{k}{d} \Rightarrow \varphi_n = \frac{ed-1}{k}$. Consider a public key $P_U = (N, e) : P_U = (2137458620009, 1973036027077)$

Continued fraction $\frac{1973036027077}{2137458620009} = [0; 1, 11, 1, 4684, 1, 125, 1, 10, 1, 2, 1, 1, 1, 2, 3, 7, 1, 17] =$

$$\left[\begin{array}{l} 1 \\ 1 + \frac{1}{11 + \frac{1}{1 + \frac{1}{4684 + \frac{1}{1 + \frac{1}{125 + \frac{1}{1 + \frac{1}{10 + \frac{1}{1 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2 + \frac{1}{3 + \frac{1}{7 + \frac{1}{1 + \frac{1}{17 + \frac{1}{\dots}}}}}}}}}}}}}}}}}}}}}}}}}} \end{array} \right]$$

$$\frac{e}{N} \approx \frac{k}{d} : \frac{e}{N} = \frac{1973036027077}{2137458620009} = \frac{1}{1 + \frac{1}{11 + 1 * \frac{1}{1}}} = \frac{12}{13} = \frac{k}{d}$$

$$\varphi_n = \frac{ed - 1}{k} = \frac{1973036027077 * 13 - 1}{12} = \frac{25649468352000}{12} = 2137455696000$$

As per Section 2, if the composite N and the totient φ_n are known, the original primes P_1 and P_2 can be recovered.

8. Sum of squares

Overmars [3] showed that all Pythagorean triples could be represented as $N = n^2 + (n + 2m - 1)^2$. If the composite number N , is constructed using two Pythagorean primes $(4x + 1)$ then two representations as the sum of two squares can be found. Euler's Factorization Method (Section 4) can be applied. Finding these two representations is non-trivial and CPU-intensive. The equation $N(m, n) = n^2 + (n + 2m - 1)^2$ provides a course search using increments of n and fine convergence using m . In this way n is incremented and m is decremented about N to find the two solutions along the diagonal of a field of $N(m, n) \approx N$.

Consider the example, $N = 2137458620009$.

$$N(m_1, n_1) = n_1^2 + (n_1 + 2m_1 - 1)^2 = 324403^2 + (324403 + 2(550579) + 1)^2 = 324403^2 + 1425560^2$$

$$N(m_2, n_2) = n_2^2 + (n_2 + 2m_2 - 1)^2 = 643603^2 + (643603 + 2(334559) + 1)^2 = 643603^2 + 1312720^2$$

$$N_1(324403, 550579) = N_2(643603, 334559) = 2137458620009$$

For completeness N can be represented as two Pythagorean triangles as shown [3] $\Delta(m, n) = \Delta(a, b, c)$.

$$a(m, n) = 2n(n + 2m - 1), \quad b(m, n) = (2m - 1)(2n + 2m - 1), \quad c(m, n) = n^2 + (n + 2m - 1)^2$$

$$\Delta(m_1, n_1) = \Delta(a_1, b_1, c_1) : \Delta(324403, 550579) = \Delta(28197495801360, 8357740887191, 29410042540009)$$

$$\Delta(m_2, n_2) = \Delta(a_2, b_2, c_2) : \Delta(643603, 334559) = \Delta(1689741060320, 1309008976791, 29410042540009)$$

Once the two sum of two squares has been found, Euler's factorization method (Section 4), can be used to find the prime constructions of $N : N = P_1P_2$.

If the composite number (N) is constructed using Pythagorean primes $(4x + 1)$, then a solution exists as two sums of two squares and Euler's factorization method can be applied.

9. Gaussian and Pythagorean primes

As shown in Section 4, if Pythagorean primes $(4x + 1 \equiv 4x - 3)$ are used to construct the composite number (N), a solution exists as two sums of two squares. However, if N is constructed using Gaussian primes $(4x - 1 \equiv 4x + 3)$, then Euler's sum of two squares method cannot be used. Is there a test that we can use to see if the composite has been constructed using Pythagorean primes? (Table 1)

Consider the following composite constructions:

i. $N = (4x + 1)(4y + 1)$ using Pythagorean primes

ii. $N = (4x - 1)(4y - 1)$ using Gaussian primes

iii. $N = (4x + 1)(4y - 1)$ using a mix of Pythagorean and Gaussian primes

i. Pythagorean prime construction

$N = (4x + 1)(4y + 1) = 16xy + 4(x + y) + 1$ Two sum of two squares representations exist and Euler's factorization can be used. $1 \equiv P \pmod{4}$.

$9 \equiv P \pmod{16}$. See Section 4. $793 = 13 * 61 = 3^2 + 28^2 = 8^2 + 27^2$

ii. Gaussian prime construction

$N = (4x - 1)(4y - 1) = 16xy - 4(x + y) + 1 \equiv 4m - 3 \equiv 4n + 1$ Sums of three squares exist. $1 \equiv P \pmod{4}$. $9 \equiv P \pmod{16}$.

	$4x - 1$	$4x + 1$	$x, y = 3, 15$	11	13
$4y - 1$	$16xy - 4(x + y) + 1$	$16xy - 4(x - y) - 1$	59	649	767
$4y + 1$	$16xy - 4(y - x) - 1$	$16xy + 4(x + y) + 1$	61	671	793

Table 1.
 Possible composite constructs using Pythagorean and Gaussian primes.

$$649 = 11 * 59 = 1^2 + 18^2 + 18^2 = 3^2 + 8^2 + 24^2 = 6^2 + 17^2 + 18^2 = 8^2 + 12^2 + 21^2 = 10^2 + 15^2 + 18^2 = 12^2 + 12^2 + 19^2$$

Legendre's three-square theorem can test the composite: $N = x^2 + y^2 + z^2$ true if $N \neq 4^a(8b + 7)$
 $a, b \in \mathbb{Z}$,

iii. Mixed Pythagorean-Gaussian prime construction

$$N = (4x + 1)(4y - 1) = 16xy - 4(x - y) - 1,$$

$$N = (4x - 1)(4y + 1) = 16xy + 4(x - y) - 1. \text{ Sums of four squares exist.}$$

$$3 \equiv P \pmod{4}. 13 * 59 = 767$$

$$1^2 + 1^2 + 6^2 + 27^2 = 1^2 + 1^2 + 18^2 + 21^2 = 1^2 + 3^2 + 9^2 + 26^2 = 1^2 + 6^2 + 17^2 + 21^2$$

$$= 1^2 + 9^2 + 18^2 + 19^2 = 1^2 + 10^2 + 15^2 + 21^2 = 2^2 + 3^2 + 5^2 + 27^2 = 2^2 + 3^2 + 15^2 + 23^2$$

$$= 3^2 + 6^2 + 19^2 + 19^2 = 3^2 + 7^2 + 15^2 + 22^2 = 3^2 + 11^2 + 14^2 + 21^2 = 5^2 + 6^2 + 9^2 + 25^2$$

$$= 6^2 + 9^2 + 11^2 + 23^2 = 6^2 + 9^2 + 17^2 + 19^2 = 6^2 + 11^2 + 13^2 + 21^2$$

$$= 7^2 + 9^2 + 14^2 + 21^2 = 7^2 + 13^2 + 15^2 + 18^2 = 9^2 + 9^2 + 11^2 + 22^2$$

$$= 9^2 + 10^2 + 15^2 + 19^2 = 11^2 + 14^2 + 15^2 + 15^2$$

In summary, a composite whose construction is based upon both Pythagorean and Gaussian primes can easily be identified when $P \pmod{4} \equiv 3$ is true. However, sums of four squares exist and Euler's factorization cannot be used. When $P \pmod{4} \equiv 1$ is true, the composite could be constructed using Pythagorean primes or Gaussian primes. Use the Legendre test to further discriminate. When the Pythagorean construct is confirmed, the two sums of two squares can be found, and Euler's factorization can be used. If the composite construction is both Pythagorean and Gaussian, sums of three squares exist and Euler's factorization cannot be used.

10. Overmars factorization method

Another classification of the composite number uses a different construct for primes and seeks to define the composite number as follows: Let $N = P_1P_2$ and test $N : (N \pm 1) \pmod{4} = 0$. Two cases are considered in the classification, and this determines the constructs of the primes used. Note the sign of ± 1 determines the case used, and the test is both simple and concise [4].

$$\text{Case (1) } \oplus \ominus (N + 1) \pmod{4} = 0, \quad P_1 = 2(m - n) + 1, \quad P_2 = 2(m + n) - 1$$

$$1. \text{ Let } m_0 \geq \frac{\sqrt{N}}{2}, m \in \mathbb{N}^+$$

$$2. \text{ Let } n_0 = \frac{\sqrt{4m_0^2 - N + 1}}{2}, n \in \mathbb{N}^+?, n \notin \mathbb{N}^+ \Rightarrow m_x = m_0 + 1$$

$$3. \text{ Let } n = \frac{\sqrt{4m_x^2 - N + 1}}{2}, n \notin \mathbb{N}^+, m_x = m_x + 1 \Rightarrow n : n \in \mathbb{N}^+$$

$$4. P_1 = 2(m - n) + 1, P_2 = 2(m + n) - 1$$

$$\text{Case (2) } \ominus\ominus(N-1) \bmod 4 = 0, \quad P_1 = 2(m-n) - 1, \quad P_2 = 2(m+n) - 1$$

$$1. \text{ Let } m_0 \geq \frac{\sqrt{N+1}}{2}, m \in \mathbb{N}^+$$

$$2. \text{ Let } n_0 = \frac{\sqrt{(2m_0-1)^2 - N}}{2}, n \in \mathbb{N}^+?, n \notin \mathbb{N}^+ \Rightarrow m_x = m_0 + 1$$

$$3. \text{ Let } n = \frac{\sqrt{(2m_x-1)^2 - N}}{2}, n \notin \mathbb{N}^+, m_x = m_x + 1 \Rightarrow n : n \in \mathbb{N}^+$$

$$4. P_1 = 2(m-n) - 1, P_2 = 2(m+n) - 1$$

Example $N = 5959$

$$1. \text{ Test } (N \pm 1) \bmod 4 = 0 : (5959 + 1) \bmod 4 = 0 \Rightarrow \text{case (1)} \oplus \ominus$$

$$2. m_0 \geq \frac{\sqrt{N}}{2} \Rightarrow m_0 = \frac{\sqrt{5959}}{2} \Rightarrow m_0 = 39, n = 6.09, n \notin \mathbb{N}^+$$

$$3. m_1 = m_0 + 1 = 39 + 1 = 40$$

$$4. n = \frac{\sqrt{4m_1^2 - N + 1}}{2} \Rightarrow n_1 = \frac{\sqrt{4(40)^2 - 5959 + 1}}{2} = 11, n_1 \in \mathbb{N}^+$$

$$5. P_1 = 2(m-n) + 1 \Rightarrow P_1 = 2(40 - 11) + 1 = 59, P_2 = 2(m+n) - 1 \Rightarrow P_2 = 2(40 + 11) - 1 = 101$$

$$N = P_1 P_2 = 59 \times 101 = 5959$$

This method is reasonable for small composites but becomes computationally unfeasible for large composites.

11. Extensions of the Overmars factorization method

$$\text{Case (1) } \oplus\ominus(N+1) \bmod a^2 = 0, \quad P_1 = a(m-n) + 1, \quad P_2 = a(m+n) - 1$$

$$N = [a(m-n) + 1][a(m+n) - 1] = a^2(m^2 - n^2) + 2an - 1$$

$$N = (am)^2 - [(an)^2 - 2an + 1] = (am)^2 - (an - 1)^2$$

$$\text{Case (2) } \ominus\oplus(N+1) \bmod a^2 = 0, \quad P_1 = a(m-n) - 1, \quad P_2 = a(m+n) + 1$$

$$N = [a(m-n) - 1][a(m+n) + 1] = a^2(m^2 - n^2) - 2an - 1$$

$$N = (am)^2 - [(an)^2 + 2an + 1] = (am)^2 - (an + 1)^2$$

$$\text{Case (1, 2) } \frac{N+1}{a} = a(m^2 - n^2) \pm 2n \quad a : a \text{ is a factor of } N + 1$$

$$n = \frac{\sqrt{(am)^2 - N \pm 1}}{a}, m \geq \frac{\sqrt{N}}{a} m = \sqrt{\frac{N + (an \pm 1)^2}{a^2}}$$

$$\text{Case (3) } \ominus\ominus(N-1) \bmod a^2 = 0, \quad P_1 = a(m-n) - 1, \quad P_2 = a(m+n) - 1$$

$$N = [a(m - n) - 1][a(m + n) - 1] = a^2(m^2 - n^2) - 2am + 1$$

$$N = (am)^2 - 2am + 1 - (an)^2 = (am - 1)^2 - (an)^2$$

Case (4) $\oplus\oplus(N - 1) \bmod a^2 = 0$, $P_1 = a(m - n) + 1$, $P_2 = a(m + n) + 1$

$$N = [a(m - n) + 1][a(m + n) + 1] = a^2(m^2 - n^2) + 2am + 1$$

$$N = (am)^2 + 2am + 1 - (an)^2 = (am + 1)^2 - (an)^2$$

Case (3, 4) $\frac{N-1}{a} = a(m^2 - n^2) \pm 2ma : a \text{ is a factor of } N - 1$

$$n = \sqrt{\frac{(am \pm 1)^2 - N}{a^2}}, m \geq \frac{\sqrt{N} \mp 1}{a}, m = \frac{\sqrt{N + (an)^2} \pm 1}{a}$$

$a : a = \gcd(m, n)$ for all cases. Choosing the largest value of a ensures a rapid convergence to the solution. This is illustrated by example.

Consider $N = 211276133$

Factors of $(N + 1) \Rightarrow 211276133 + 1 = (2)(3^3)(881)(4441)$ possible values for a

Factors of $(N - 1) \Rightarrow 211276133 - 1 = (2^2)(52819033)$ possible values for a

Case (3) $\ominus\ominus(N - 1) \bmod a^2 \Rightarrow (211276133 - 1) \bmod 4 = 0 \Rightarrow a = 2$

$$[2(m - n) - 1][2(m + n) - 1] = 211276133, m = 10247, n = 7223 \Rightarrow \gcd(10247, 7223) = 1$$

$$P_1 = 2(10247 - 7223) - 1 = 6047, P_2 = 2(10247 + 7223) - 1 = 34939$$

Case (2) $\ominus\oplus(N + 1) \bmod a^2 \Rightarrow (211276133 + 1) \bmod 9 = 0 \Rightarrow a = 3$

$$[3(m - n) - 1][3(m + n) - 1] = 211276133, m = 6831, n = 4815 \Rightarrow \gcd(6831, 4815) = 9$$

$$[27(m - n) - 1][27(m + n) - 1] = 211276133, m = 759, n = 535 \Rightarrow \gcd(759, 535) = 1$$

$$P_1 = 27(759 - 535) - 1 = 6047, P_2 = 27(759 + 535) + 1 = 34939$$

Consider $N = 5959$ (Section 8)

Factors of $(N - 1) \Rightarrow 5959 - 1 = (2)(3^2)(331)$ possible values for a

$$P_1 = 3(m - n) - 1, P_2 = 3(m + n) - 1, m = 27, n = 7, \gcd(27, 7) = 1$$

Factors of $(N + 1) \Rightarrow 5959 + 1 = (2^3)(5)(149)$ possible values for a

$$P_1 = 20(m - n) + 1, P_2 = 20(m + n) - 1, m = 4, n = 1, \gcd(4, 1) = 1$$

Consider RSA100

$$P_1 = 37975227936943673922808872755445627854565536638199$$

$$P_2 = 40094690950920881030683735292761468389214899724061$$

$$P_1 = (2)(3167)(3613)(1659412543822590349622856694449324700910569) + 1$$

$$P_1 = (2^3)(3)(5^2)(109)(409)(20839813)(60236089)(49147216823)(23011759155976667) - 1$$

$$P_2 = (2^2)(5)(41)(2119363)(602799725049211)(38273186726790856290328531) + 1$$

$$P_2 = (2)(3)(11)(59)(1029653080403720622569012658644444886804031773) - 1$$

$$\begin{aligned}
 N &= P_1P_2 \\
 &= [(2^3)(3)(5^2)(109)(409)(20839813)(60236089)(49147216823)(23011759155976667) - 1] \\
 &\quad * [(2^2)(5)(41)(2119363)(602799725049211)(38273186726790856290328531) + 1] \\
 \text{factors of } N + 1 &= (2^2)(5)(7)(13^2)(63421)(83694613) \\
 &\quad (121238883482226494959007093210067761113089 \\
 &\quad 3465646351221267386320068406978173999673) \\
 \text{factors of } N - 1 &= (2)(3^2)(210974974123) \\
 &\quad (400944086233670527306310281636760087998315 \\
 &\quad 351567377660286363410284049027879820778576767)
 \end{aligned}$$

$N + 1$ is the better candidate, as it has more factors to try. So cases (1,2) are considered.

$$\begin{aligned}
 \text{Case (2) } N &= [a(m - n) - 1][a(m + n) + 1] = a^2(m^2 - n^2) - 2an - 1 \frac{N+1}{a} = \\
 &= a(m^2 - n^2) - 2n \text{ Try } a : a = (2)(5) : \frac{N+1}{a} = a(m^2 - n^2) - 2n, \\
 \frac{N+1}{10} &= 10(m^2 - n^2) - 2n = \frac{N+1}{20} = 5(m^2 - n^2) - n \\
 m &\geq \frac{\sqrt{N}}{a} = 3902057185540126551228957333948437101890500690019
 \end{aligned}$$

$$\begin{aligned}
 \frac{N + 1}{a} &= (15226050279225333605356183781326374297180681149613806886 \\
 &\quad 57908494580122963258952897654000350692006139) + 1/20 \\
 &= 76130251396126668026780918906631871485903405748069034 \\
 &\quad 432895424729006148162947644882700017534600307
 \end{aligned}$$

$$\begin{aligned}
 a = 10 &\Rightarrow m = 3903495944393227747674630402410354812189021818113, \\
 n &= 105973150698860355393743126865792026732468154293 \text{ gcd}(m, n) = 1
 \end{aligned}$$

$$P_1 = 10(m - n) + 1 = 37975227936943673922808872755445627854565536638199,$$

$$P_2 = 10(m + n) - 1 = 40094690950920881030683735292761468389214899724061$$

When a is small, this method becomes computationally unfeasible.

12. Overmars factorization using smooth factors

Consider the construction of primes (Sections 8 and 9), $P = a(m \pm n) \pm 1$. More generally, $P : P = a(m \pm n) \pm x$ Consider $N = P_1P_2 \Rightarrow 8079781 = 1249 \times 6469$ (Table 2).

$$\text{Case (1) } \oplus \ominus (N + x^2) \bmod a^2 = 0, \quad P_1 = a(m - n) + x, \quad P_2 = a(m + n) - x$$

$$N = [a(m - n) + x][a(m + n) - x] = a^2(m^2 - n^2) + 2anx - x^2$$

$$N = (am)^2 - [(an)^2 - 2anx + 1] = (am)^2 - (an - x)^2$$

$$\text{Case (2) } \ominus \oplus (N + x^2) \bmod a^2 = 0, \quad P_1 = a(m - n) - x, \quad P_2 = a(m + n) + x$$

$$N = [a(m - n) - 1][a(m + n) + 1] = a^2(m^2 - n^2) - 2anx - x^2$$

$$N = (am)^2 - [(an)^2 + 2anx + 1] = (am)^2 - (an + x)^2$$

x	$N - x^2$	$\pm x$	a	m	n	$\gcd(m,n)$	Smoothness
1	$2^2 3 5 311 433$	$\ominus\ominus$	10	386	261	1	5-smooth
3	$2^2 479 4217$	$\ominus\ominus$	2	1931	1305	1	
5	$2^2 3 673313$	$\ominus\ominus$	6	644	435	1	
7	$2^2 3^2 103 2179$	$\oplus\oplus$	18	214	145	1	3-smooth
11	$2^2 3^2 5 44887$	$\ominus\ominus$	90	43	29	1	5-smooth
13	$2^2 3 211 3191$	$\oplus\oplus$	6	641	435	1	
17	$2^2 3 673291$	$\ominus\ominus$	6	646	435	1	
19	$2^2 3 5 17 89^2$	$\oplus\oplus$	30	128	87	1	5-smooth
23	$2^2 3 673271$	$\ominus\ominus$	6	647	435	1	
29	$2^2 3^4 5 4987$	$\ominus\ominus$	18	216	145	1	5-smooth

Table 2.
 $N - x^2$.

Case (1,2) $\frac{N+x^2}{a} = a(m^2 - n^2) \pm 2nx$ $a : a$ is a factor of $N + x^2$

$$n = \frac{\sqrt{(am)^2 - N \pm x}}{a}, m \geq \frac{\sqrt{N + (a \mp x)^2}}{a}, m = \sqrt{\frac{N + (an \pm x)^2}{a^2}}$$

Case (3) $\ominus\ominus(N - x^2) \bmod a^2 = 0$, $P_1 = a(m - n) - x$, $P_2 = a(m + n) - x$

$$N = [a(m - n) - x][a(m + n) - x] = a^2(m^2 - n^2) - 2amx + x^2$$

$$N = (am)^2 - 2amx + x^2 - (an)^2 = (am - x)^2 - (an)^2$$

Case (4) $\oplus\oplus(N - x^2) \bmod a^2 = 0$, $P_1 = a(m - n) + x$, $P_2 = a(m + n) + x$

$$N = [a(m - n) + x][a(m + n) + x] = a^2(m^2 - n^2) + 2amx + x^2$$

$$N = (am)^2 + 2amx + x^2 - (an)^2 = (am + x)^2 - (an)^2$$

Case (3,4) $\frac{N-x^2}{a} = a(m^2 - n^2) \pm 2mx$ $a : a$ is a factor of $N - x^2$

$$n = \sqrt{\frac{(am \pm x)^2 - N}{a^2}}, m \geq \frac{\sqrt{N \mp x^2}}{a}, m = \frac{\sqrt{N + (an)^2 \pm x^2}}{a}$$

$$N = [90(43 - 29) - 11][90(43 + 29) - 11] = 1249 \times 6469$$

When a smooth x can be found, larger a values allow for faster convergence to a solution. The selection of x and a is somewhat arbitrary and prime constructs are a modification of Fermat's $a^2 - b^2$. Smooth factors of $N \pm x^2$ produce larger a values and convergence faster to a solution.

13. Primes

The current state of the art in prime number generation is Atkin's sieve [5, 6].

The algorithm completely ignores any numbers with remainder mod 60 that is divisible by 2, 3 or 5, since numbers with a mod 60 remainder divisible by one of

these three primes are themselves divisible by that prime. Atkin stated three theorems given below:

1. All numbers n with mod 60 remainder 1, 13, 17, 29, 37, 41, 49 or 53 are mod 4 \equiv 1. These numbers are prime if the number of solutions to $4x^2 + y^2 = n$ is odd and the number is squarefree.
2. All numbers n with mod 60 remainder 7, 19, 31 or 43 have a mod 6 \equiv 1. These numbers are prime if and only if the number of solutions to $3x^2 + y^2 = n$ is odd and the number is squarefree.
3. All numbers n with mod 60 remainder 11, 23, 47 or 59 have a mod 12 \equiv 11. These numbers are prime if and only if the number of solutions to $3x^2 - y^2 = n$ is odd and the number is squarefree.

None of the primes are divisible by 2, 3 or 5 and are not divisible by their squares (2^2 , 3^2 , and 5^2). For a thorough analysis of “primes of the Form $x^2 + ny^2$ ” the reader is referred to a text by Cox [7].

The often overlooked works of Dubner, who is credited with the term “primorial” [8] are now considered [9, 10]. The primorial is a factorial of primes: $1\# = 2$, $2\# = 2 \times 3 = 6$, $3\# = 2 \times 3 \times 5 = 30$, $4\# = \#3 \times 7 = 210$ and so on. $0\# = 1$. The primorial is by definition squarefree.

The n th primorial is the product of n primes, where $\pi(n)$ is the prime counting function.

$$n\# = \prod_{i=1}^{\pi(n)} p_i = p_{\pi(n)}\#$$

Using this structure, Dubner was able to create series of primes in a particular primorial.

It can be shown that the structure of primes is palindromic in the primorials [11].

For example, in **Figure 1**, take the discrete derivative of the numbers in the third primorial, $3\#$. The following palindromic sequence can be added to $\#3 = 30$ and subtracted from $\#4 = 210$ to determine all of the primes in that primorial:

$$\begin{array}{r} 30 + 1, 10, 2, 4, 2, 4, 6, 2, 6, 4, 2, 4, 6, 6, 2, 6, 4, 2, 6, 4, 6, 8, 4, 2 \\ 210 - 1, 10, 2, 4, 2, 4, 6, 2, 6, 4, 2, 4, 6, 6, 2, 6, 4, 2, 6, 4, 6, 8, 4, 2 \end{array}$$

This describes the second table in **Figure 1**. All of the primes in the third primorial can be found using 24 *small* numbers. Mod 7 is used to sieve and eliminate composite multiples of 7. Mod 11 and 13 are used to highlight further composites, but these are kept and used to generate primes in the next primorial.

Modulo testing: $P \bmod m = 0$, $P_k < m < \sqrt{(k+1)\#}$

For $k = 3$, $P_k : P_3 = 5$, $P_{k+1} : P_4 = 7$, $\#3 = 30$, $\#4 = 210$, $\sqrt{210} \approx 14$, $m = 7, 11, 13$, eliminate $P_{k+1} = 7$

As shown in **Figure 2**, 24 small numbers are used to derive 482 new values. This uses 10 modulo tests to identify composites and 1 modulo test to eliminate factors of 11 (**Figure 3**).

$P_n\#, \Delta P_{n-1}\#$ Current primorial and the difference between primes from the previous. Simple array descriptor provides rich prime fields of higher densities. Small numbers describe primes of higher magnitude. Large arrays of primes can be stored in much less memory.

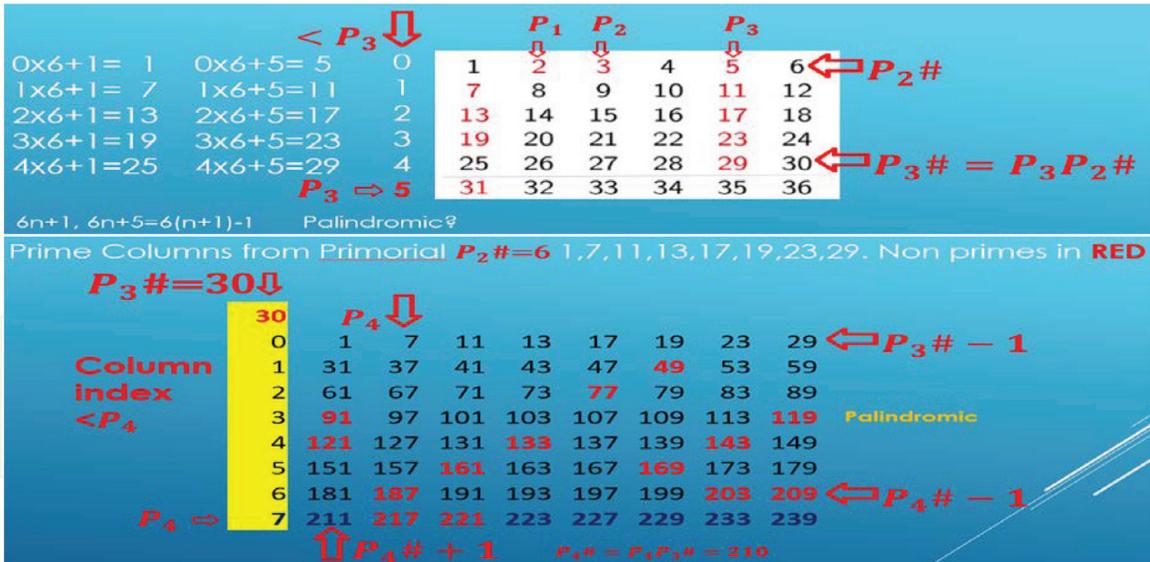


Figure 1. Creating primes using primorials.

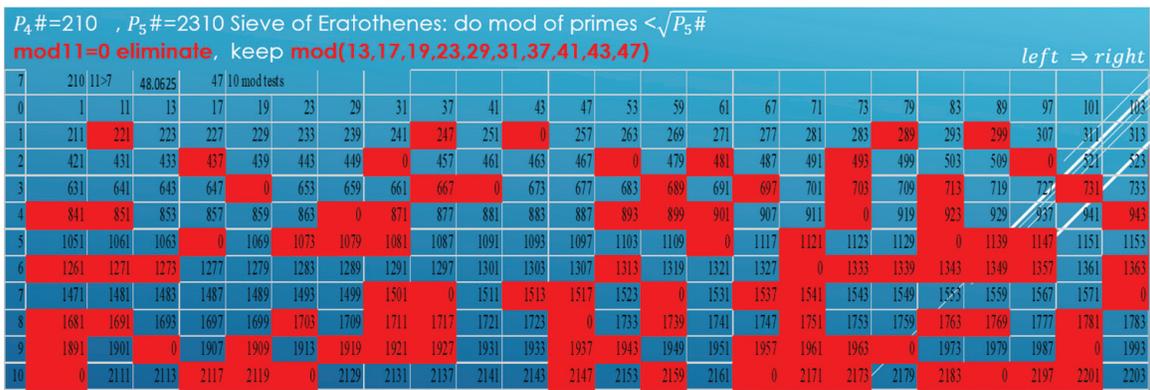


Figure 2. Primes in the 4th primorial.

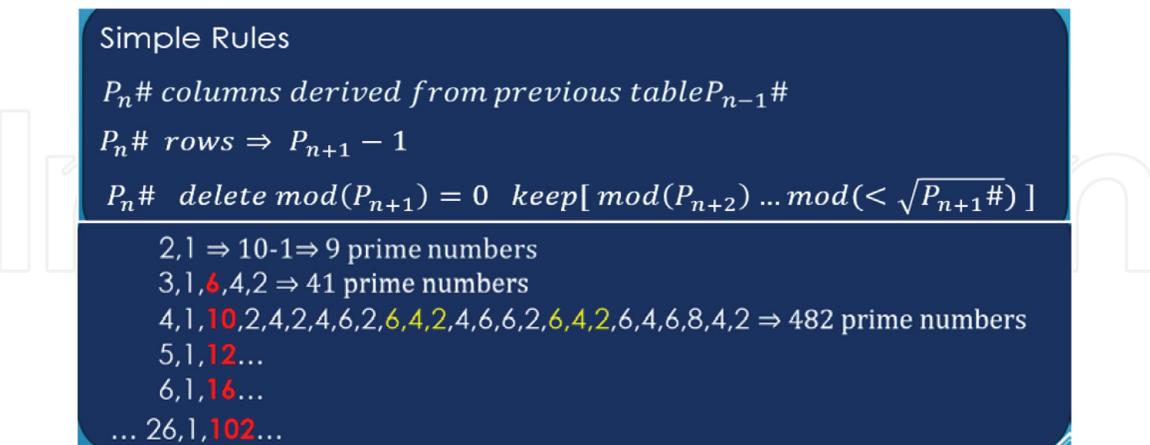


Figure 3. Gaps between primes of each successive primorial.

14. Number systems

Conventional numbering systems consist of a base (or radix).

The primorial number system is said to be ‘primoradic’; having a primorial base. The primorial number system is a mixed radix numeral system adapted to the numbering of the primorials (Table 3).

n	...	7	6	5	4	3	2	1
p_n	$n...$	17	13	11	7	5	3	2
$n\#$...	510510	30030	2310	210	30	6	2
highest	$P_{n+1} - 1$	18	16	12	10	6	4	1

Table 3.
Primorial radix number system.

General properties of mixed radix number systems apply to the base primorial system. The primorial number system OEIS A000040 is denoted by a subscript “ Π ”.

Consider the following example:

Primorial to decimal, Base Π to Base $_{10}$

$3\ 4\ 1\ 0\ 1_{\Pi}$ stands for $3_4 4_3 1_2 0_1 1_0$, whose value is

$$= 3 \times p_{4\#} + 4 \times p_{3\#} + 1 \times p_{2\#} + 0 \times p_{1\#} + 1 \times p_{0\#} = 3 \times 210 + 4 \times 30 + 1 \times 6 + 0 \times 2 + 1 \times 1$$

$$= (((((3 \times 7 + 4) \times 5 + 1) \times 3 + 0) \times 2 + 1) \times 1 = 757_{10}.$$

Decimal to primorial, Base $_{10}$ into Base Π

757_{10} into a primorial representation by successive divisions:

$$757 \div 2 = 231, \text{ remainder } 1$$

$$378 \div 3 = 126, \text{ remainder } 0$$

$$126 \div 5 = 25, \text{ remainder } 1$$

$$25 \div 7 = 3, \text{ remainder } 4$$

$$3 \div 11 = 3, \text{ remainder } 3 \Rightarrow 3\ 4\ 1\ 0\ 1_{\Pi}$$

15. RSA100 factorization using primorials

$$N = (P_1)(P_2) = (aP_{k\#} + c)(aP_{k\#} + d) = (aP_{k\#})^2 + (c + d)aP_{k\#} + cd$$

$$P_{k\#}^2 \leq N \quad 1522605027922533360535618378132637429718068114961380688657908 \dots$$

$$494580122963258952897654000350692006139 / p_{31\#}^2$$

$$(aP_{k\#})^2 \leq N \quad 1522605027922533360535618378132637429718068114961380688657908 \dots$$

$$494580122963258952897654000350692006139 / (9p_{31\#}^2)$$

$$N = (aP_{k\#} + c)(aP_{k\#} + d) = (aP_{k\#} + cP_{k-1\#} + e)(aP_{k\#} + dP_{k-1\#} + f)$$

$$P_{k\#} = P_k(P_{k-1\#})$$

$$N = (aP_k(P_{k-1\#}) + cP_{k-1\#} + e)(aP_k(P_{k-1\#}) + dP_{k-1\#} + f)$$

$$= ((aP_k + c)P_{k-1\#} + e)((aP_k + d)P_{k-1\#} + f)$$

$$N = (aP_k + c)(aP_k + d)(P_{k-1\#})^2 + (f(aP_k + c) + e(aP_k + d))(P_{k-1\#}) + ef$$

$$(aP_k + c)(aP_k + d)(P_{k-1\#})^2 \leq N \Rightarrow (aP_k + c)(aP_k + d) = \frac{N - N \bmod (P_{k-1\#})^2}{(P_{k-1\#})^2}$$

$$N = 1523830x^2 + 27406046005166967437863263040740903499726862x$$

$$+ 12231378224719217781270707850591564671548897759$$

$1523830 = 2 \times 5 \times 7 \times 11 \times 1979 = (770)(1979) = (1234 - 464)(1234 + 745)$
 Not symmetrical about square root [12]
 $1522868 = 2^2 \times 317 \times 1201 = (1201)(1268) = (1234 - 33)(1234 + 34)$
 Symmetrical about square root.

$$N = (aP_k + c)(aP_k + d)(P_{k-1\#})^2 + (f(aP_k + c) + e(aP_k + d))(P_{k-1\#}) + ef$$

$$(aP_k + c)(aP_k + d)(P_{k-1\#})^2 \leq N \Rightarrow (aP_k + c)(aP_k + d) = \frac{N - N \bmod (P_{k-1\#})^2}{(P_{k-1\#})^2}$$

1521642935492617539765579106664136748401379615914 ∙.
 312169315386041883234627722692028711378934397966 ∙.
 $800/p_{30\#}^2$

Consider each congruency and look for a factorization that is symmetrical about the square root.

In this case $1234 + 34 = 1268$, $1234 - 33 = 1201$.

$$N = (aP_k + c)(aP_k + d)(P_{k-1\#})^2 + (f(aP_k + c) + e(aP_k + d))(P_{k-1\#}) + ef$$

30431475913593577738588710930551227419722971658953x+

151816659580901664885523419281115998823527019067345405631 ∙.
 401183567090345342039152734187917869,

$$N = ((aP_k + c)P_{k-1\#} + e)((aP_k + d)P_{k-1\#} + f)$$

$k = 31$, $P_{31} = 127$, $(aP_k + c) = 1201$, $(aP_k + d) = 12$
 $a = 9$, $c = 58$, $d = 125$, $P_{31} = 127$

$$N = (9P_{31\#} + 58P_{30\#} + e)(9P_{31\#} + 125P_{30\#} + f)$$

$$N = (1201)(1268)P_{30}^2 + (1201f + 1268e)P_{30} + ef$$

$$N = (a^2 + m)P_{31}^2 + (a(c + d) + n)P_{31} + cd$$

$$a^2 + m = \frac{N - N \bmod P_{k\#}^2}{P_{k\#}^2} = 94 \Rightarrow a = 9, m = 13$$

$$a^2 P_{k\#}^2 + [a(c + d) + m P_{k\#}] P_{k\#} + (n P_{k\#} + cd)$$

$$P_{k\#} = P_k(P_{k-1\#}) \Rightarrow N = (1201)(1268)P_{30\#}^2 + (1201f + 1268e)P_{30\#} + ef$$

$$N = (9P_{31\#} + 58P_{30\#} + e)(9P_{31\#} + 125P_{30\#} + f)$$

Repeat these steps for $P_{29\#}$ and so on... (Table 4)

$$N = (9P_{31\#} + 58P_{30\#} + 41P_{29\#} + g)(9P_{31\#} + 125P_{30\#} + 46P_{29\#} + h)$$

k	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	...	1
P_k	127	113	109	107	103	101	97	89	83	79	73	71	67	61	59	53		2
P_1	9	58	41	32	43	101	13	14	60	50	54	33	3	32	12	12		1
P_2	9	125	46	106	75	95	71	79	21	3	19	58	23	32	30	13		1

Table 4.
 P_1 and P_2 as base Primorial numbers.

$$N = 1522868x^2 + 3043147581359377738588710930551227419722971658953x + 151816659580901664885523419281115998823527019067345405631 \cdot.$$

$$401183567090345342039152734187917869.$$

$$N = (1268x + 13141666871354355315613715084104347742596620741) (1201x + 11552313802126969246479999301689200142637563209), x = p_{30\#}$$

$$N = (9P_{31\#} + 58P_{30\#} + e)(9P_{31\#} + 125P_{30\#} + f)$$

$$N = (9P_{31\#} + 58P_{30\#} + 11552313802126969246479999301689200142637563209) * (9P_{31\#} + 125P_{30\#} + 13141666871354355315613715084104347742596620741)$$

$$N = (9P_{31\#} + 58P_{30\#} + 41P_{29\#} + g)(9P_{31\#} + 125P_{30\#} + 46P_{29\#} + h)$$

$$N = (9P_{31\#} + 58P_{30\#} + 41P_{29\#} + 83178932594916863170676664934419945962676779) * (9P_{31\#} + 125P_{30\#} + 46P_{29\#} + 273857017733028251413011637989228497546748161)$$

The conversion to a decimal from the base primorial (Section 12) provides P_1 and P_2

$$P_1 = (37975227936943673922808872755445627854565536638199)_{10}$$

$$P_2 = (40094690950920881030683735292761468389214899724061)_{10}$$

16. Lenstra-Lenstra-Lavász lattice reduction (LLL)

The (LLL) forms the basis of the Coppersmith attack (Section 15), and a brief explanation is given here with further reading and references for the reader. The Lenstra-Lenstra-Lavász (LLL) lattice basis reduction algorithm [13] calculates an LLL-reduced, short, nearly orthogonal lattice basis, in time $O(d^5 n \log^3 B)$, where B is the largest length of b_i under the Euclidean norm, given a basis $B = \{b_1, b_2, \dots, b_d\}$ with n -dimensional integer coordinates, for a lattice L (a discrete subgroup of R^n) with $d \leq n$ and giving polynomial-time factorization of polynomials with rational coefficients.

A thorough explanation is given by Bosma [14], and a summary of the example contained in the reference is given below.

INPUT: Let lattice basis $b_1, b_2, b_3 \in Z^3$ be given by the columns of
$$\begin{bmatrix} 1 & -1 & 3 \\ 1 & 0 & 5 \\ 1 & 2 & 6 \end{bmatrix}$$

OUTPUT: LLL-reduced basis
$$\begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 2 \end{bmatrix}$$

Using the Lenstra-Lenstra-Lavász lattice reduction (LLL), the short vectors in a lattice can be found. This is used by the Coppersmith attack. Coppersmith's algorithm uses the LLL to construct polynomials with small coefficients that all have the same root modulo. When a linear combination is found to meet inequality conditions, standard factorization methods can find the solutions over integers.

17. Coppersmith attack

When d is small (and e is large, via the Euler totient rule), the Wiener attack (Section 5) can be used. Conversely, when d is large, e is small. Particular applications of the Coppersmith method for attacking RSA include cases when the public exponent e is small or when partial knowledge of the secret key is available (Section 13) [15].

A small public exponent e , reduces the encryption time. Common choices for e are 3, 17 and $65537(2^{16}+1)$ [16]. These are Fermat primes $F_x : F_x = 2^{2^x} + 1$ and are chosen because the modular exponent derivation is faster. The Coppersmith method reduces the solving of modular polynomial equations to solving polynomial equations over integers.

Let $F(x) = x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ and $F(x_0) \equiv 0 \pmod{M}$ for an integer $|x_0| < M^{\frac{1}{n}}$. Coppersmith can find the integer solution for x_0 by finding a different polynomial f related to F that has the root $x_0 \pmod{M}$ but only has small coefficients. The small coefficients are constructed using the LLL (Section 14). Given F , the LLL constructs polynomials $p_1(x), p_2(x), \dots, p_n(x)$ that all have same root $x_0 \pmod{M^a}$, $a \in \mathbb{Z}$. a depends on the degree of F and the size of x_0 . Any linear combination has the same root $x_0 \pmod{M^a}$.

The next step is to use LLL to construct a linear combination $f(x) = \sum c_i p_i(x)$ of the $p_i(x)$ so that the inequality $|f(x_0)| < M^a$ holds. Then standard factorization provides the zeroes of $f(x)$ over \mathbb{Z} .

Let N be an integer and $f \in \mathbb{Z}[x]$ be a monic polynomial of degree d , over integers such that $x^d + c_{n-1}x^{d-1} + \dots + c_2x^2 + c_1x + c_0$. Set $X = N^{\frac{1}{d}-\epsilon}$ for $\frac{1}{d} > \epsilon > 0$. Given (N, f) then all integers $x_0 < X : f(x_0) \equiv 0 \pmod{N}$ can now be found. All roots of $f \pmod{N}$, smaller than $X = N^{\frac{1}{d}}$ can be found.

18. Pohlig-Hellman

The Pohlig-Hellman [17] algorithm is a method to compute a discrete logarithm (which is a difficult problem) on a multiplicative group. The order of which is a smooth number (also called *friable*), meaning its order can be factorized into small primes. A positive integer is called B -smooth if none of its prime factors is greater than B . For example, 1620 has prime factorization $2^2 \times 3^4 \times 5$; therefore 1620 is 5-smooth because none of its prime factors are greater than 5. This is similar to that of the Overmars factorization method (Section 10). The Pohlig-Hellman [17] algorithm applies to groups whose order is a prime power. The basic idea is to iteratively compute the p -adic digits of the logarithm by repeatedly “shifting out” all but one unknown digit in the exponent and computing that digit by elementary methods. This is a similar idea to Section 13.

INPUT: A cyclic group G of order n with a generator g , an element $h \in G$, and a prime factorization $n = \prod_{i=1}^r p_i^{e_i}$ OUTPUT: The unique integer $x \in \{0, \dots, n-1\} : g^x = h$

Example: Let $p = 41, \alpha = 7, \beta = 12$ solve $12 = 7^x \pmod{41}$

1. Find the prime factors of $p-1 \Rightarrow 41-1 = 40 = 2^3 \cdot 5 \Rightarrow g_s = 2, 5$. Find one x for each g .
2. For $g = 2, x = 2^0 x_0 + 2^1 x_1 + 2^2 x_2 \cdot 2^3 \Rightarrow \text{cubic} \rightarrow \text{three terms}$

i. $x_0 : \beta_{g_0}^{\frac{p-1}{g}} = \alpha^{\frac{p-1}{g}x_0} \Rightarrow 12^{\frac{40}{2}} = \left(\frac{40}{7^2}\right)^{x_0} - 1 \pmod{41} = (-1)^{x_0} \pmod{41}$ test for $x_0 : x_0 = 0, 1, 2, \dots$

$-1 \pmod{41} \equiv (-1)^0 \pmod{41} - 1 \pmod{41} \equiv (-1)^1 \pmod{41}$ hence $x_0 = 1$

ii. $x_1 : \beta_1 = \beta_0 \alpha^{-(x_0)} = 12(7)^{-(1)} = 31 \pmod{41}$

$\beta_1^{\frac{p-1}{g_1}} = \alpha^{\frac{p-1}{g_1}x_1}$, $g_1 = 2^2 31^{\frac{40}{4}} = \left(\frac{40}{7^2}\right)^{x_1} \Rightarrow 31^{10} = (7^{20})^{x_1}$ $31^{10} \Rightarrow (1 \pmod{41})$ hence $x_1 = 0$

iii. $x_2 : \beta_2 = \beta_1 \alpha^{-(x_1)} = (31)(7^{-(0)}) = 31 \pmod{41}$

$\beta_2^{\frac{p-1}{g_2}} = \alpha^{\frac{p-1}{g_2}x_2}$, $g_2 = 2^3 31^{\frac{40}{8}} = \left(\frac{40}{7^2}\right)^{x_2} \Rightarrow 31^5 = (7^{20})^{x_2} - 1 \pmod{41} = -1^{\frac{1}{2}} \pmod{41}$ hence $x_2 = 1$

Recall: $X = 2^0x_0 + 2^1x_1 + 2^2x_2$ so $X = 1.1 + 2.0 + 4.1 = 5$

$x = 5 \pmod{2^3} = 5 \pmod{8}$. Now we need another x from the other g

3. For $g = 5, x = 5^0x_0$ only one 5, only one term.

i. $x_0 : \beta_{g_0}^{\frac{p-1}{g}} = \alpha^{\frac{p-1}{g}x_0} \Rightarrow 12^{\frac{40}{5}} = \left(\frac{40}{7^5}\right)^{x_0} \Rightarrow 12^8 = (7^8)^{x_0} \Rightarrow 18 \equiv 37^{x_0} \pmod{41}$

$x_0 \neq 0, 1$ try $x_0 = 2$ $18 \equiv 37^2 \pmod{41}$ $18 \equiv 37^3 \pmod{41}$ hence $x = 5^0x_0 = (1)(3) = 3$

Hence $x = 3 \pmod{5}$, so $x = 5 \pmod{8}$ and $x = 3 \pmod{5}$

By the Chinese remainder theorem, $x = 13 \pmod{40}$ since the exponents are $p - 1 = 41 - 1 = 40$ hence $12 \equiv 7^{13} \pmod{41}$. So the solution to $12 = 7^x \pmod{41} \Rightarrow x = 13$.

19. Shor's algorithm

Shor's algorithm [18], factors composite numbers, $N = P_1P_2$, consisting of two primes in polynomial time using quantum computing techniques. The algorithm evaluates the period of $a^x \pmod{n}$ where $\gcd(a, n) = 1$. This is inefficient using sequential computing on a conventional computer. When run on a quantum computer, a congruence of squares with probability 0.5 occurs in polynomial time. For two co-prime sinusoids of period P_1 and P_2 , at what point do they zero-cross each other? The phase of each sinusoid at any given point is observed, and if they are

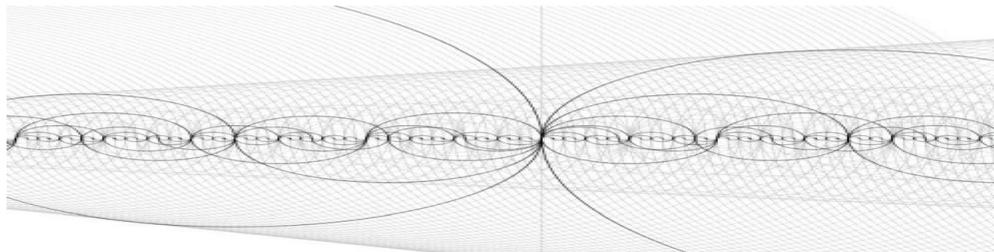


Figure 4.
 N as a composite of two Sinusoids P_1 and P_2 [19].

factors of N then the phase of P_1 and P_2 is zero. Shor's algorithm tests the phase of $P_1 = P_2 = N = 0$ (Figure 4).

Phase estimation is well suited to quantum computers and hence this factorization technique produces solutions in polynomial time. For further information on quantum phase estimation, the reader is directed to WIKI [20]. The impact of this type of attack is discussed in detail by Mosca [21].

1. Choose $a < N$
2. Find the period r of $a^n \bmod N$ (using Quantum computing)
3. Check r is even : $a^{\frac{r}{2}} + 1 \equiv 0 \bmod N$
4. $P_1 = \gcd(a^{\frac{r}{2}} - 1, N), P_2 = \gcd(a^{\frac{r}{2}} + 1, N)$

Consider $N = 35$,

1. $a : a < N$, choose $a = 8$
2. Find the period r of $a^n \bmod N$
 - a. $8^1 \bmod 35 = 8$
 - b. $8^2 \bmod 35 = 29$
 - c. $8^3 \bmod 35 = 22$
 - d. $8^4 \bmod 35 = 1$
 - e. $8^5 \bmod 35 = 8 \Rightarrow \text{period } r = 4$

3. $r : r$ even, $r = 4$ is even

$$4. P_1 = \gcd(a^{\frac{r}{2}} - 1, N) = \gcd(8^2 - 1, 35) = \gcd(63, 35) = 7$$

$$P_2 = \gcd(a^{\frac{r}{2}} + 1, N) = \gcd(65, 35) = 5$$

Euler's factorization (Section 6) cannot be used because 7 has no sum of squares nor does 35.

Fermat's factorization (Section 5)

$$N = (a - b)(a + b) = a^2 - b^2 = 36 - 1 = 6^2 - 1^2 = (6 - 1)(6 + 1) = (5)(7) = 35$$

Overmars factorization (Section 10)

$$N = [a(m - n) + 1][a(m + n) + 1] = [2(4 - 2) + 1][2(4 + 2) + 1] = [5][7]$$

Overmars triangles (Section 8) $\Delta(m, n) = \Delta(a, b, c) : \Delta(3, 1) = \Delta(12, 35, 37)$ Recalling $b(m, n) = (2m - 1)(2n + 2m - 1) \Rightarrow b(3, 1) = (5)(7)$

20. Attacking public key infrastructure

Public infrastructure cryptographic hardware uses a library *RSALib*. This is found in both NIST FIPS 140-2 and CC EAL 5+. These are certified devices for use in identity cards, passports, Trusted Platform Modules, PGP and tokens for authentication and software signing. This is in use in tens of millions of devices worldwide. Nemec et al. [22] have identified a vulnerability that allows for the factorization of 1024 and 2048 bit keys in less than 3 CPU months.

RSALib primes are of the form $p = kM + (65537^a \bmod M)$.

These can be fingerprinted using the discrete logarithm $\log_{65537} N \bmod M$.

$$N = P_1 P_2 = (k * M + 65537^a \bmod M) (l * M + 65537^b \bmod M) \Rightarrow N \equiv 65537^{a+b} \equiv 65537^c \bmod M$$

The public modulus N is generated by 65537 in the multiplicative group \mathbb{Z}_M . The public modulus of *RSALib* can thus be fingerprinted with the discrete logarithm $c = \log_{65537} N \bmod M$. This can be factorized using Pohlig-Hellman (Section 16). The group $G = 65537$ is smooth $|G| = 2^4 * 3^4 * 5^2 * 7 * 11 * 13 * 17 * 23 * 29 * 37 * 41 * 53 * 83$ for RSA_{512} keys. The smoothness of G is due to the smoothness of M being Primorial.

Factorization is achieved using the Coppersmith algorithm with a known $p \bmod M : 65537^a \bmod M$. Nemec *et al* used the Howgrave-Graham[23] implementation of the Coppersmith's algorithm to find a small solution x_0 of:

$$f(x) = x + (M^{p-1} \bmod N) (65537^a \bmod M) \pmod{N}$$

A summary of *RSALib* vulnerability and its impact is now given and the reader is directed to Memec et al. [22] for further detail. eIDs used in passports for citizens are affected. Code signing is vulnerable. Twenty-four percent of TPMs used in laptops are affected (sample size 41). A third of PGP, used in email systems could be factorizable. There was no observable impact on TLS/HTTPS. One hundred percent of SCADA systems sampled were affected (sample 15). E-health and EMV payment cards were also likely to be susceptible.

Mitigating the impact of the *RSALib* vulnerability requires changing the algorithm. This requires a firmware replacement which is not possible in already deployed devices such as smartcards and TPMs whose code is stored in read-only memory. Key lengths not of 512, 1024, 2048 and 4096, such as RSA_{3936} appear to be resilient. The use of key pairs outside of vulnerable devices could be deployed using another library. Changes to *RSALib* are required so that proveable safe primes are constructed not using the vulnerability.

21. Overmars factorization, bringing it together

Section 11 considered the following cases. The following discussion generalizes these cases and provides the structure for algorithmic solutions to be found. The palindromic nature of primes (Section 12) can be exploited further to explore solutions in a particular Primorial range. Recall;

$$\text{Case } (1\oplus\ominus, 2\ominus\oplus) (N + x^2) \bmod a^2 = 0, \quad P_1 = a(m - n) \pm x, \quad P_2 = a(m + n) \mp x$$

$$N = [a(m - n) \pm x][a(m + n) \mp x] = a^2(m^2 - n^2) \mp 2anx - x^2 = (am)^2 - (an \mp x)^2$$

$$\frac{N + x^2}{a} = a(m^2 - n^2) \pm 2nx \quad a : a \text{ is a smooth factor of } N + x^2$$

$$n = \frac{\sqrt{(am)^2 - N} \pm x}{a}, m : \frac{\sqrt{N + (a \mp x)^2}}{a} \leq m < \infty m = \frac{\sqrt{+(an \mp x)^2}}{a}$$

$$P_1 = a(m - n) \mp x = am - \sqrt{(am)^2 - NN} \bmod \left[am - \sqrt{(am)^2 - N} \right] \equiv 0$$

Case (3 $\ominus\ominus$, 4 $\oplus\oplus$) $(N - x^2) \bmod a^2 = 0$, $P_1 = a(m - n) \mp x$, $P_2 = a(m + n) \mp x$
 $N = [a(m - n) \mp x][a(m + n) \mp x] = a^2(m^2 - n^2) \mp 2amx + x^2 = (am \mp x)^2 - (an)^2$
 $\frac{N - x^2}{a} = a(m^2 - n^2) \mp 2mx$ $a : a$ is a smooth factor of $N - x^2$

$$n = \frac{\sqrt{(am \mp x)^2 - N}}{a}, m : \frac{\sqrt{N + a^2} \pm x^2}{a} \leq m < \infty, m = \frac{\sqrt{N + (an)^2} \pm x^2}{a}$$

$$P_1 = a(m - n) \mp x = am \mp x - \sqrt{(am \mp x)^2 - NN} \bmod \left[(am \mp x) - \sqrt{(am \mp x)^2 - N} \right] \equiv 0$$

Now we need to develop the methodology for finding (selecting) a and x . This brings together the concepts of primorials [9], Smooth [24], small factors [17], factorization (Fermat), modulo testing as per Atkin's Sieve [5] and the structure of primes (Sections 12 and 18), to find as large an a as possible so that Overmars Factorization [4] converges more rapidly to a solution.

Recall the following (Section 12). Primes are of the form $P = 4x \pm 1$ and $P = 6x \pm 1$. Composite numbers, constructed from these primes: $N = P_1P_2$, are a combination of Pythagorean and Gaussian primes. The following test $(N \pm 1) \bmod 4 \equiv 0$ can be used to determine which combination of primes was used to construct the composite. If $(N + 1) \bmod 4 \equiv 0$ is true a mix of Pythagorean and Gaussian primes was used. If $(N - 1) \bmod 4 \equiv 0$ is true then the composite consists of only Gaussian or only Pythagorean primes. The Sieve of Atkin [5] uses $\bmod 12 \equiv 0$ and $\bmod 60 \equiv 0$. This is now applied as per Overmars [4] in the following manner, if $\bmod 12 \equiv 0$ is true then $a = 6$, if $\bmod 60 \equiv 0$ is true let $a = 30$. The ideas of Atkin are further extended in both directions: $\bmod 4 \equiv 0 \Rightarrow a = 2$, $\bmod 420 \equiv 0 \Rightarrow a = 210$, $\bmod 4620 \equiv 0 \Rightarrow a = 2310$, $\bmod 60060 \equiv 0 \Rightarrow a = 30030$...

This is Primorial, $P_{k\#} : P_{k\#}$, k^{th} Primorial is "Smooth". The general form (Section 19) is now given: Case (1 $\oplus\oplus$, 2 $\ominus\oplus$) $\frac{N+x^2}{a} = a(m^2 - n^2) \pm 2nx$,

$$(N + x^2) \bmod a \equiv 0, a : a = 2P_{k\#}, x : 1 \leq x \leq \frac{\sqrt{N}}{a}$$

$$\text{Case (3 } \ominus\ominus, 4 \oplus\oplus) \frac{N-x^2}{a} = a(m^2 - n^2) \mp 2mx, (N - x^2) \bmod a \equiv 0, a : a = 2P_{k\#}, x : 1 \leq x \leq \frac{\sqrt{N}}{a}$$

If $a : a = 2P_{k\#}$ can be chosen, then we search x in the primes to find solutions to $(N \pm x^2) \bmod (2P_{k\#}) \equiv 0$ A solution is found for $P_1(m)$, when $P_1 \in \mathbb{Z}$. Case

$$(1 \oplus\oplus, 2 \ominus\oplus) N \bmod [P_1] \equiv 0, P_1 : P_1 = am - \sqrt{(am)^2 - N}$$

$$\text{Case (3 } \ominus\ominus, 4 \oplus\oplus) N \bmod [P_1] \equiv 0, P_1 : P_1 = am \mp x - \sqrt{(am \mp x)^2 - N}$$

Consider Section 11 example, $N = P_1P_2 \Rightarrow 8079781 = 1249 * 6469$

Integer solutions $x = \sqrt{N - 2bP_{k\#}}$. From **Table 5**, determining which x value should be used is not clear. Whilst $x = 1$ should work, no solutions will be found if $a : a = 30$. From **Table 5** only when $x = 11$ or 19 do we find solutions. Ranking the possible solutions in terms of factors 29 (8) would be first, 19 (7) second and 11 (6) third.

Based upon low order factors the rankings would be 29 ($2^2 3^4$) first and 11 ($2^2 3^2$) second. Setting $a = 30, x = 29$ will not find solutions for m, n . Setting $a = 30, x = 11 \Rightarrow m = 129, n = 57, \gcd(129, 57) = 3$, so the optimal value for

x	mod60	mod180	mod1620	$N - x^2$	$\pm x$	b	a	m	n	gcd(m,n)	Smoothness
1	0			$2^2 3 5 311 433$	$\ominus\ominus$	10	386	261	1	1	5-smooth
1	0			$2^2 3 5 311 433$	$\oplus\oplus$	6	643	435	1	1	5-smooth
11	0	0		$2^2 3^2 5 44887$	$\ominus\ominus$	3	90	43	29	1	5-smooth
19	0			$2^2 3 5 17 89^2$	$\oplus\oplus$	1	30	128	87	1	5-smooth
29	0	0	0	$2^2 3^4 5 4987$	$\ominus\ominus$	18	216	145	1	1	5-smooth

Table 5.
Smooth candidates of the factors of $N - x^2$.

$a = 90$. $P_1 = 30m - 11 - \sqrt{(30m - 11)^2 - N}$. Look for solutions to $(30m - 11)^2 - N$ which are a perfect square. In this case, $m = 129 \Rightarrow (30 * 129 - 11)^2 - 8079781 = 6812100 = 2610^2$.

Recall that the starting value for $m : \frac{\sqrt{N+a^2+x^2}}{a} \leq m < \frac{N-1}{2a} \Rightarrow 99 \leq m < 134663$, 30 iterations.

Whilst this is quite a good result the first failure needs also to be taken into account. This would be bound by the Primorial and

$$P_1: 1 < P_1 < \sqrt{N} : am - \sqrt{(am)^2 - N} = 1 \Rightarrow m < \frac{N+1}{2a}$$

Here $m : \frac{\sqrt{N+a^2+x^2}}{a} \leq m < \frac{N+1}{2a} \leq 123 \leq m < 134663 \Rightarrow 134540$ iterations.

This can be further bound by the Primorial. In the case of RSA numbers, the binary bits available to represent a particular prime number range can also be used to bound the range (**Table 6**).

Consider $N = 23852269081$.

In this case, solutions using modulo testing generate good candidates to solve for (m, n) , however for $a = 30030$, three of the candidates have no solution. Using sequential programming, each possible candidate is considered one after another, until the maximum m value. However, using parallel programming techniques on GPUs (such as nVIDIA P100s), all of the candidates can be tested simultaneously and the processes are all terminated when one of the processes finds a solution. This is very efficient and effective in finding P_1, P_2 . Once these are known, along with the public key $P_u = (N, e)$, using Euler's totient, the private key $P_R = (N, d)$ can be determined. Once the private key is known the cypher-text is no longer secure.

x	Modulo testing				$N - x^2$	a	m	n	gcd(m,n)	Smoothness	
	60	420	4620	60060							
1	0				$2^3 3^2 5 101 461 1423$	30			1	5-smooth	
11	0				$2^5 3 5 13 97 157 251$	30	5524	2002	2	5-smooth	
19	0	0			$2^4 3^3 5 7 1577531$	210	789	286	1	1	7-smooth
61	0	0	0		$2^4 3 5 7 11^3 10667$	2310					11-smooth
401	0	0	0	0	$2^3 3 5 7^2 11 13 19 1493$	30030					13-smooth
1601	0	0	0	0	$2^3 3^3 5 7 11 13^2 1697$	30030					13-smooth
45281	0	0	0	0	$2^3 3 5 7 11 13 181501$	30030					13-smooth
45589	0	0	0	0	$2^5 3 5 7 11 13 45317$	30030	4	2	2	2	13-smooth

Table 6.
Smooth candidates of the factors of $N - x^2$.

22. Conclusion

In short RSA is secure and difficult to factorise. Conventional sequential computing machines, running in polynomial time, take an infeasible amount of CPU cycles to find factorization solutions to RSA keys. Quantum computing holds great promise and Shor's algorithm [18] demonstrates how this can be achieved. However, quantum computing is realistically still some way off. Opportunities exist using conventional computing (sequential and parallel) with better mathematical techniques. Section 18 showed how implementation vulnerabilities are introduced when "clever" low cost (CPU cycles) are implemented. The case in point showed methods for signature identification, upon which tailored targeted attacks could be launched against infrastructure FIPS140-2 devices, such as cryptographic routers. These sorts of attacks can be deployed in polynomial time using sequential programming techniques. Section 20, Overmars shows how factorization can be implemented using parallel processing techniques.

There is still much to be done and areas of further interest are a better understanding of the structure of primes. This will lead to faster prime number generating algorithms and hence faster solutions to the factorization problem. This will also lead to the generation of more robust primes that are less susceptible to factorization methods. An example of this is the use of non-Pythagorean primes. Section 5 showed how Euler's factorization could be used to attack such composite numbers. Hence a simple method to thwart this would be to use a mix of Pythagorean and Gaussian primes. Section 6 showed how small d values in the RSA private key $P_R = (N, d)$ could be attacked using Wiener's method. Small e values in the public key $P_U = (N, e)$ can be attacked using a combination of LLL, Coppersmith and Pohlig-Hellman (Sections 15–17). All of these attacks can be mitigated by choosing d and e carefully and ensuring that both are sufficiently large.

Development of quantum computing is continuing at break-neck speed, however useful machines are yet to appear. Parallel computing however is here and now and whilst factorizing RSA keys is not achievable on conventional computers in polynomial time, parallel computing has allowed for multiple solutions to be tested simultaneously. This is an area where research continues and new algorithms such as shown in Sections 20 and 14 lend themselves well to GPU parallel processing systems.

"There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know" [25].

Author details

Anthony Overmars
Curtin University, Perth, Australia

*Address all correspondence to: 3@crykey.com

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Available from: <https://www.marketsandmarkets.com/Market-Reports/network-encryption-market-187543224.html> [Retrieved: 17 January 2019]
- [2] Boneh D. Twenty years of attacks on the RS cryptosystems. Notices for the American Mathematical Society (AMS). 1999;**46**(2)
- [3] Overmars A, Ntogramatzidis L, Venkatraman S. A new approach to generate all pythagorean triples. AIMS Mathematics. 2019;**4**(2):242-253. DOI: 10.3934/math.2019.2.242
- [4] Overmars A, Venkatraman S. A fast factorisation of semi-primes using sum of squares. Computers & Mathematics with Applications. 2019;**24**(2):62. <https://doi.org/10.3390/mca24020062>
- [5] Atkin AOL, Bernstein DJ. Prime sieves using binary quadratic forms. Mathematics of Computation. 2004;**73**: 1023-1030
- [6] Sieve of Atkin. Wikipedia. Available from: https://en.wikipedia.org/wiki/Sieve_of_Atkin
- [7] Cox DA. Primes of the Form $x^2 + ny^2$. 2nd ed. John Wiley & Sons; 2013. ISBN: 978-1-118-39018-4
- [8] Primorial. Wikipedia. Available from: <https://en.wikipedia.org/wiki/Primorial>
- [9] Dubner H. Factorial and primorial primes. Journal of Recreational Mathematics. 1987;**19**(3):197-203
- [10] Dubner H. A new primorial prime. Journal of Recreational Mathematics. 1989;**21**(4):276
- [11] Overmars A. The palindromic structure of primes in the primorials. TBA; 2019
- [12] Overmars A. RSA100 factorisation using primorials. TBA; 2019
- [13] Lenstra AK, Lenstra HW Jr, Lovász L. Factoring polynomials with rational coefficients. Mathematische Annalen. 1982;**261**(4):515-534
- [14] Bosma W. Chapter 4 LLL. In: Lecture Notes. 2010. pp. 86-109. Available from: <http://www.math.ru.nl/~bosma/onderwijs/voorjaar07/compalg7.pdf> [Retrieved: 17 January 2019]
- [15] Coppersmith D. Finding a small root of a bivariate integer equation; factoring with high bits known. Lecture Notes in Computer Science. 1996;**1070**:178-189
- [16] Salah IK, Darwish A, Oqeili S. Mathematical Attacks on RSA cryptosystems. Journal of Computer Science. 2006;**2**(8):665-671
- [17] Pohlig SC, Hellman ME. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Transactions on Information Theory. 1978;**IT-24**(1):106-110
- [18] Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. 1995. Available from: <https://arxiv.org/abs/quant-ph/9508027>
- [19] Davidson M. Prime numbers visualized as periodic waveforms. 2012. Available from: <https://cycling74.com/articles/prime-numbers-visualized-as-periodic-waveforms> [Retrieved: 31/01/2019]
- [20] WIKI. Quantum phase estimation algorithm. Available from: https://en.wikipedia.org/wiki/Quantum_phase_estimation_algorithm
- [21] Mosca M. A resource estimation framework for quantum attacks against

cryptographic functions—Part 2 (RSA and ECC). 2017. Available from: <https://globalriskinstitute.org/publications/resource-estimation-framework-quantum-attacks-cryptographic-functions-part-2-rsa-ecc/>

[22] Nemeč M, Sys M, Svenda P, Klinec D, Matyas V. The return of Coppersmith's attack: Practical factorization of widely used RSA moduli. In: CCS'17, Session H1: Crypto Attacks; October 30-November 3, 2017, Dallas, TX, USA. 2017. pp. 1631-1648

[23] Howgrave-Graham N. Finding small roots of univariate modular equations revisited. In: Proceedings of the 6th IMA International Conference on Cryptography and Coding. Springer-Verlag; 1997. pp. 131-142

[24] Schumaker R. The formulas for the distribution of the 3-smooth, 5-smooth, 7-smooth and all other smooth numbers. 2016. Available from: <https://arxiv.org/abs/1608.06928>

[25] Rumsfeld D. 2002. Available from: https://en.wikipedia.org/wiki/There_are_known_knowns

IntechOpen