The ray attack, an inefficient trial to break RSA cryptosystems*

Andreas de Vries†

FH Südwestfalen University of Applied Sciences, Haldener Straße 182, D-58095 Hagen

Abstract

The basic properties of RSA cryptosystems and some classical attacks on them are described. Derived from geometric properties of the Euler functions, the *Euler function rays*, a new ansatz to attack RSA cryptosystems is presented. A resulting, albeit inefficient, algorithm is given. It essentially consists of a loop with starting value determined by the Euler function ray and with step width given by a function $\omega_e(n)$ being a multiple of the order $\mathrm{ord}_n(e)$, where e denotes the public key exponent and n the RSA modulus. For $n = pq$ and an estimate $r < \sqrt{pq}$ for the smaller prime factor p, the running time is given by $T(e,n,r) = O((r-p)\ln e \ln n \ln r)$.

Contents

1 Introduction

Since the revolutionary idea of asymmetric cryptosystems was born in the 1970's, due to Diffie and Hellman [4] and Rivest, Shamir and Adleman [9], public key technology became an indispensable part of contemporary electronically based communication. Its applications range

*This paper is a slight modification of [10]

†e-Mail: de-vries@fh-swf.de

from authentication to digital signatures and are widely considered to be an essential of future applications for e-commerce.

The most popular cryptosystem is RSA. There has been numerous, more or less unsuccessful, attacks on RSA. The strongness of RSA bases on the difficulty to factorize integers as well as to compute the discrete logarithm. For more details, see e.g. [1, 2, 3, 6]; cf. also `http://www.math-it.org`

2 RSA cryptosystem

The RSA cryptosystem, named after its inventors Ron Rivest, Adi Shamir, and Len Adleman (1978), was the first public key cryptosystem and is still the most important one. It is based on the dramatic difference between the ease of finding large prime numbers and computing modular powers on the one hand, and the difficulty of *factorizing* a product of large prime numbers as well as *inverting* the modular exponentiation.

Generally, in a public key system, each participant has both a *public key* and a *private key*, which is held secret. Each key is a piece of information. In the RSA cryptosystem, each key consists of a group of integers. The participants are traditionally called Alice and Bob, and we denote their public and secret keys as P_A, S_A for Alice and P_B, S_B for Bob. All participants create their own pair of public and private keys. Each keeps his private key secret, but can reveal his public key to anyone or can even publish it. It is very convenient that everyone's public key is available in a public directory, so that any participant can easily obtain the public key of any other participant, just like we nowadays can get anyones phone number from a public phone book.

In the *RSA cryptosystem*, each participant creates his public and private keys with the following procedure.

1. Select at random two large prime numbers p and q, $p \neq q$. (The primes might be more than 200 digits each, i.e. more than 660 bits.)
2. Compute $n = pq$ and the Carmichael function $\lambda(n) = \operatorname{lcm}(p-1, q-1)$.
3. Select an integer d relatively prime to $\lambda(n)$. (d should be of the magnitude of n, i.e., $d \lessapprox \lambda(n)$.)
4. Compute e as the multiplicative inverse of d modulo $\lambda(n)$, such that $ed = 1 \bmod \lambda(n)$. This is done efficiently by the extended Euclidean algorithm.
5. Publish the pair $P = (e, n)$ as the *public key*.
6. Keep secret the pair $S = (d, n)$ as the *private* or *secret key*.

For this procedure, the domain of the messages is $\mathbb{Z}_n$. For each participant of a cryptosystem, the four-tuple $(e, d, p, q) \in \mathbb{N}^4$ is called *(individual) RSA key system*. The key parameter e is also called the *encryption exponent*, d the *decryption exponent*, and n the *RSA modulus*.

The encryption of a message $m \in \mathbb{Z}_n$ associated with a public key $P = (e, n)$ is performed by the function $E : \mathbb{Z}_n \to \mathbb{Z}_n$,

$$E(m) = m^e \bmod n. \tag{1}$$

The decryption of a ciphertext $c \in \mathbb{Z}_n$ associated with the private key $S = (d,n)$ is done by the mapping $D : \mathbb{Z}_n \to \mathbb{Z}_n$,

$$D(c) = c^d \bmod n. \tag{2}$$

The procedure where Alice sends an encrypted message to Bob is schematically shown in Figure 1. A qualitatively new possibility offered by public key systems (and being unimple-

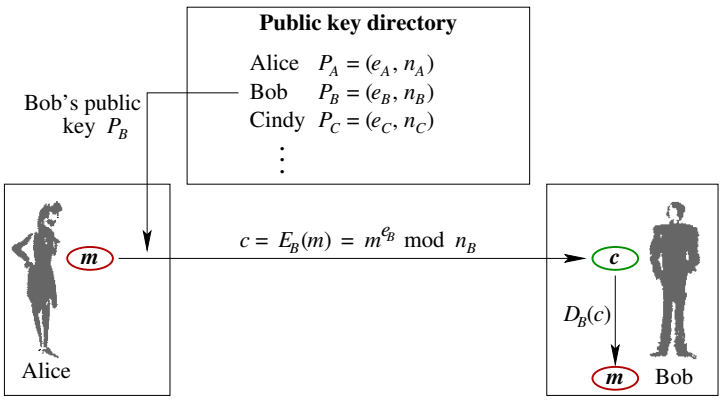

Figure 1: Alice sends an encrypted message m to Bob, using *his* public RSA key P_B.

mentable with symmetric key systems) is the procedure of *digital signature*. How an RSA cryptosystem enables Alice to digitally sign a message and how Bob can verify that it *is* signed by Alice is sketched in Figure 2. As a matter of course, this verification in fact is possible only if the authenticity of Alice's public key P_A is guaranteed such that Bob can assume that it is her key (and not a third person's one) which he uses. This guarantee is the job of so-called trust centers.

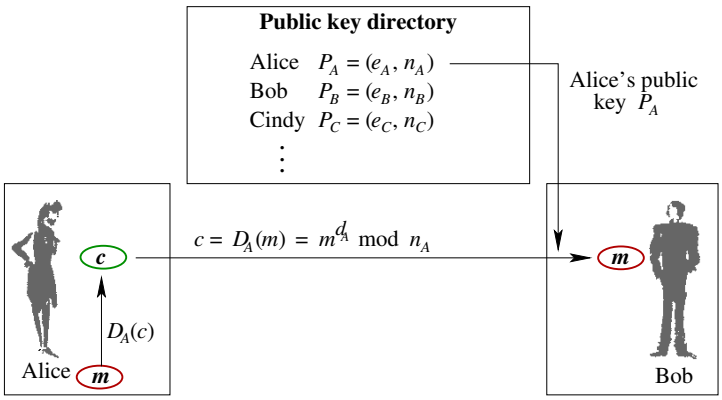

Figure 2: Alice sends a digitally signed message to Bob; Bob uses Alice's public key to decrypt the message and to verify this way that Alice has signed it with her private key.

The correctness of RSA, i.e., the fact that E and D define inverse functions on $\mathbb{Z}_n$ ($D \circ E = E \circ D = \mathrm{id}_{\mathbb{Z}_n}$) relies on the simple fact that

$$m^{ed} = m \bmod n \qquad \text{for } m \in \mathbb{Z}_n, \tag{3}$$

which is immediately proved by the corollary of Carmichael A.7, p. 18. For details see, e.g., [1, 2, 3].

Remark 2.1 Often one finds the definition of RSA cryptosystems based on the Euler function φ rather than on the Carmichael function λ, cf. [3]. However, since $\varphi(pq) = (p-1)(q-1)$, both function values $\varphi(pq)$ and $\lambda(pq)$ share the same divisors. Therefore, a possible key parameter d relatively prime to $\lambda(pq)$ is also relatively prime to $\varphi(pq)$, and vice versa. Only the resulting counter key e may differ. To be more precise, any possible RSA key pair of a system based on the Euler function is a possible key pair with respect to the Carmichael function, whereas the reverse is not generally true. (Proof: Since $\lambda(n)|\varphi(n)$, the equality $ed = 1 \bmod \varphi(n)$ implies $ed = 1 \bmod \lambda(n)$.) Using the Euler function φ, the correctness of RSA is shown with the Euler theorem A.2 on p. 16, instead of the corollary of Carmichael.

2.1 Properties of an RSA key system

Theorem 2.2 *Let $p, q \in \mathbb{N}$ be two primes, $p,q > 1$, $p \neq q$. Then the number ν_{pq} of all possible key pairs $(P,S) = ((e,pq),(d,pq))$ is given by*

$$\nu_{pq} = \varphi(\lambda(pq)). \tag{4}$$

The (trivial) keys with $e = d = 1$ and with $e = d = \lambda(pq) - 1$ are always possible, and

$$2 < \nu_{pq} < \frac{(p-1)(q-1)}{\gcd(p-1,q-1)}. \tag{5}$$

Proof. Since $ed = 1 \bmod \lambda(pq)$, without restriction to generality we have $0 < e,d < \lambda(pq)$. Moreover, $\gcd(d,\lambda(pq)) = \gcd(e,\lambda(pq)) = 1$, because for an arbitrary integer a with $\gcd(a, \lambda(pq)) > 1$ there exists no $b \in \mathbb{N}$ such that $ab = 1 \bmod \lambda(pq)$. Therefore, $e,d \in \mathbb{Z}^*_{\lambda(pq)}$. In turn, to any $a \in \mathbb{Z}^*_{\lambda(pq)}$ there exists an integer b such that $ab = 1 \bmod \lambda(pq)$, since $\mathbb{Z}^*_{\lambda(pq)}$ is a group. But the order of $\mathbb{Z}^*_{\lambda(pq)}$ is exactly $\varphi(\lambda(pq))$.

It is clear that $1 \cdot 1 = 1 \bmod \lambda(n)$, so $e = d = 1$ are always possible as key parameters. If $e = d = \lambda(pq) - 1$, we have $ed = \lambda^2(pq) - 2\lambda(pq) + 1 = 1 \bmod \lambda(pq)$, so e and d are always possible, too. By (47), $\lambda(pq)$ is even and (by $pq \geqq 6$) greater than 2, so $\nu_{pq} > 2$. The maximum number of elements on the other hand is $\lambda(pq) - 1$. □

The plot of all possible RSA key parameters (e,d) reveals general symmetries in the (e,d)-plane. First we observe that if $P = (e,n)$, $S = (d,n)$ is a possible RSA key pair, then trivially also $P' = (d,n)$, $S' = (e,n)$ is possible, because $ed = de = 1 \bmod \lambda(n)$. Furthermore, if $ed = 1 \bmod \lambda(n)$ and $0 < e,d < \lambda(n)/2$, then

$$e' = \lambda(n) - e, \qquad d' = \lambda(n) - d \tag{6}$$

satisfy $\lambda(n)/2 < e', d' < \lambda(n)$ as well as

$$e'd' = \lambda^2(n) - \lambda(n)(e+d) + ed = ed \bmod \lambda(n).$$

Therefore, $P' = (e',n)$ and $S' = (d',n)$ are possible RSA keys, too.

To sum up, all possible RSA key parameters (e,d), plotted in the square lattice $[0,\lambda(n) - 1]^2 \subset \mathbb{N}^2$ with edges ranging from 0 to $\lambda(n) - 1$, form a pattern which is symmetric to both the principal and the secondary square diagonals, see Figure 3. Thus, the region

$$U = \{(e,d) \in [0,\lambda(n)-1]^2 : 0 < d \leqq \min(e, \lambda(n) - e)\} \tag{7}$$

contains all information to generate the rest of the square lattice by reflections at the main diagonal ($d \leftrightarrow e$) and at the secondary diagonal (6).

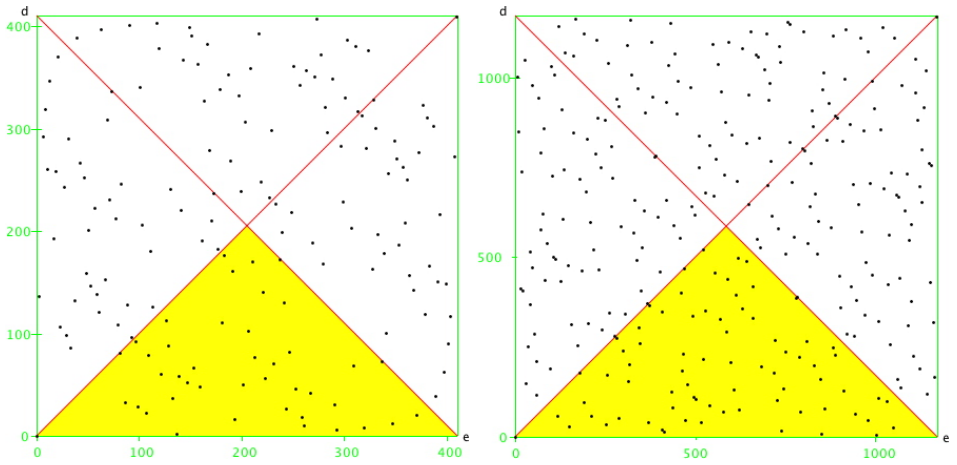

Figure 3: Plots of the possible RSA key parameter pairs $(e,d) \in [0,\lambda(n)-1]^2$ for different primes p and q, represented as points in the (e,d)-plane. For the first plot, $p = 11$ and $q = 83$, for the second one $p = 19$ and $q = 131$. The shaded region is U as given by (7).

2.2 Classical RSA attacks

There are several specific methods to break an RSA cryptosystem. The initial situation for an attack is that an eavesdropper knows the public key $P = (e,n)$ and the encrypted message c. For details see, e.g., [1] and [2, §7].

2.2.1 Factorization of the RSA modulus *n*

If the eavesdropper succeeds in finding the factorization $n = pq$ of n, knowing e he can easily compute d. But factorization of numbers $n = pq$ with

$$\boxed{p,q > 10^{200}} \tag{8}$$

(hence $n > 10^{400}$, i.e., n has length more than about 1320 bits), is difficult with current technology, if p and q differ enough,

$$\boxed{|p-q| > 10^{100}.} \tag{9}$$

Otherwise n can be factorized efficiently by exhaustive search of two integers n_+ and n_- satisfying $n = n_+^2 - n_-^2$, beginning at $n_+ = \lceil\sqrt{n}\rceil$ and $n_- = 0$. These two integers then necessarily obey $n_\pm = \frac{p\pm q}{2}$.

It can be proved that, knowing the public key (e,n), factorizing the RSA modulus n is as difficult as finding the secret key (d,n), see [2, §7.2.5].

Factorization is the most efficient known attack on RSA. The fastest known factorization method, the *number field sieve* of John Pollard in 1988, yields running times for a 10 GHz computer as given in Table 1.

2.2.2 Chosen-plaintext attack

The eavesdropper systematically encrypts all messages m with Bob's public key P_B until he achieves the ciphertext c. This attack is efficient if the set of messages m is small or if the

magnitude of the number	bits	operations	CPU time
$n \approx 10^{50}$	167	$1.4 \cdot 10^{10}$	14 seconds
$n \approx 10^{75}$	250	$9 \cdot 10^{12}$	2.5 hours
$n \approx 10^{100}$	330	$2.3 \cdot 10^{15}$	26.6 days
$n \approx 10^{200}$	665	$1.2 \cdot 10^{23}$	3.8 mio years
$n \approx 10^{300}$	1000	$1.5 \cdot 10^{29}$	$4.9 \cdot 10^{12}$ years

Table 1: CPU times for factorizing of numbers n on a 10 GHz computer.

message m is short.

> "Pad" each message such that its size is of the magnitude of the modulus. Use "probabilistic encryption," where a given plaintext is mapped onto several ciphertexts. (10)

2.2.3 Chosen-ciphertext attack

There is a similar method, the chosen-ciphertext attack, which can be applied if Bob *signs* a document with his private key. The eavesdropper receiving the ciphertext c and wishing to find the decryption $m = c^d \bmod n$ chooses a random integer s and asks Bob to digitally sign the innocent-looking message $\tilde{c} = s^e c \bmod n$. From his answer $\tilde{m} = \tilde{c}^d$ it is easy to recover the original message, because $m = \tilde{m}/s \bmod n$.

> Never sign unknown documents; before signing a document, always apply a one-way hash function to it. (11)

2.2.4 Message iteration

Let be $c_i \in \mathbb{Z}_n$ be iteratively defined as

$$c_0 = m, \qquad c_i = c_{i-1}^e \bmod n \qquad (i = 1, 2, \ldots).$$

In fact, $c_i = m^{e^i} \bmod n$, and $c_1 = c$ is the ciphertext. The smallest index k with $c_{k+1} = c_1$ is the iteration exponent or period of m, cf. definition 3.3: it exactly shows (!) the original message,

$$c_k = m.$$

Such a period k uniquely exists, it is the order of e modulo $\lambda(n)$, $k = \mathrm{ord}_{\lambda(n)}(e)$, cf. (15). Thus it divides $\lambda(\lambda(n))$ and $\varphi(\lambda(n))$. To avoid an efficient attack by iteration, $\lambda(\lambda(n))$ and the order of e with respect to $\lambda(n)$ have to be large,

$$\boxed{\lambda(\lambda(n)),\ \mathrm{ord}_{\lambda(n)}(e) > 10^{200}.} \tag{12}$$

This condition is satisfied for so-called "doubly safe primes" p and q: A prime p is *doubly safe*, if both $(p-1)/2$ and $(p-3)/4$ are primes. For instance, 11, 23, 47, 167, 359 are doubly safe primes. A doubly safe prime $p \neq 11$ always has the form $24a - 1$, or $p = -1 \bmod 24$. For two doubly safe primes p, q, we have $\lambda(pq) = 2\ \frac{p-1}{2}\ \frac{q-1}{2}$, and therefore $\lambda(\lambda(pq)) = \mathrm{lcm}(2, \frac{p-3}{2}, \frac{q-3}{2}) = \frac{1}{2}\ \frac{p-3}{2}\ \frac{q-3}{2} = (p-3)(q-3)/8$.

2.2.5 Broadcast decryption by the low-exponent attack

In general, it may be convenient to use a small public key parameter e such that the encryption of a message is easy to compute (for instance for a small chip card). However, suppose Alice sends the same message to l different participants whose public keys are $P_i = (e, n_i)$ where the n_i's are relatively prime to each other and $l \leqq e$; to emphasize, the public keys have the same encryption exponent e. If an eavesdropper receives the l ciphertexts $c_i' = m^e \bmod n_i$, he can easily compute $c' = c_i' \bmod n_1 \cdots n_l$ by the Chinese remainder theorem. But if the product $n_1 \cdots n_l$ is great enough, this is the same as $c' = m^e$. This equation is invertible, viz., $m = \sqrt[e]{c'}$, and the original message is computed. To avoid this attack, each pair of public keys $P_i = (e_i, n_i)$ $P_j = (e_j, n_j)$ and any broadcast message m must satisfy

$$\boxed{e_i \neq e_j \quad \text{or} \quad m^{e_i},\ m^{e_j} > n_i n_j} \tag{13}$$

2.2.6 Broadcast decryption by the common modulus attack

If a plain text m is encrypted twice by the RSA system using two public keys $P_i = (e_i, n)$, $i = 1, 2$, with a common modulus n and $\gcd(e_1, e_2) = 1$, then m can be recovered efficiently from the two ciphertexts c_1 and c_2, each of which given by $c_i = m^{e_i} \bmod n$. This is done by the following procedure.

1. Compute x_1, x_2 satisfying $x_1 e_1 + x_2 e_2 = 1$ by the extended Euclidean algorithm, where the indices are chosen such that $x_2 < 0$.

2. Determine y satisfying $1 = y c_2 + kn$ by the extended Euclidean algorithm.

3. Calculate ${c_1}^{x_1} y^{-x_2}$ — this is the plain text!

The reason is that ${c_1}^{x_1} y^{-x_2} = {c_1}^{x_1}\ {c_2}^{x_2} = m^{x_1 e_1 + x_2 e_2} = m \bmod n$. E.g., let be $P_1 = (3, 493)$ and $P_2 = (5, 493)$, and the corresponding ciphertexts $c_1 = 293$ and $c_2 = 421$. Then the extended Euclidean algorithm yields $x_1 = 2$ and $x_2 = -1$, and thus $y = 89$ and $k = -76$ (such that $89 \cdot 421 - 76 \cdot 493 = 1$); finally, $293^2 \cdot 89^1 = 67 \cdot 89 = 5963 = 47 \bmod n$, i.e. $m = 47$ is the plaintext. In fact, $493 = 17 \cdot 29$, and $S_1 = (17, 29, 75)$, $S_2 = (17, 29, 45)$, and $m = c_1^{75} = c_2^{45} = 47 \bmod 493$.

Therefore, to avoid common modulus attacks, a sender should regard:

$$\boxed{\text{Never send identical messages to receivers with the same modulus and relatively prime encryption exponents.}} \tag{14}$$

3 The Euler function ray attack

3.1 The ω-function and the order of a number modulo n

Definition 3.1 Let be $n \in \mathbb{N}$, $n > 1$, and $\mathbb{Z}_n^*$ the multiplicative group modulo n. Then the *order* $\mathrm{ord}_n(m)$ of $m \in \mathbb{Z}_n^*$ is given by

$$\mathrm{ord}_n(m) = \min\{k \in \mathbb{N} :\ k > 0,\ m^k = 1 \bmod n\}. \tag{15}$$

If $\gcd(m, n) > 1$, $\mathrm{ord}_n(m) = \infty$.

Let $\langle m \rangle$ denote the subgroup of $\mathbb{Z}_n^*$ generated by m. E.g., $\langle 2 \rangle = \{1, 2, 4\}$ in $\mathbb{Z}_7$, and $\mathrm{ord}_7(2) = 3$. Note that $\varphi(7) = \lambda(7) = 6$.

Lemma 3.2 *Let be $m, n \in \mathbb{N}$, with $\gcd(m,n) = 1$ and $m < n$. Then*

$$\mathrm{ord}_n(m) \mid \lambda(n). \tag{16}$$

Moreover,

$$\lceil \log_m n \rceil \leqq \mathrm{ord}_n(m) \leqq \lambda(n) \leqq n-1. \tag{17}$$

Proof. With Carmichael's theorem A.4 and with the Lagrange theorem [3, §33] equation (16) is deduced.

Let $a = \mathrm{ord}_m(n)$. Since $m > 1$, we have $m^a > n$ to obtain $m = 1 \bmod n$. This implies $a > \log_m n$. The upper limits follow from the relations (55) and (16). □

Definition 3.3 Let be $m, n, e \in \mathbb{N}$, $n > 1$, and define the sequence $(c_0, c_1, c_2, \ldots)$ iteratively by

$$c_0 = m, \qquad c_i = c_{i-1}^e \bmod n \qquad (i = 1, 2, \ldots). \tag{18}$$

Then the smallest $k \geqq 1$ such that $c_k = c_0$ is called (n,e)-*iteration exponent* $s(n, e, m)$ of m. It is the period of the *cycle* $(c_0, c_1, \ldots, c_{s(n,e,m)-1})$ to which m belongs. A cycle with period one is a fixed point.

Lemma 3.4 *Let be e, m, n and the sequence $(c_0, c_1, c_2, \ldots)$ as in definition 3.3. Let moreover be e relatively prime to $\lambda(n)$. Then the (n,e)-iteration exponent $s(n,e,m)$ satisfies*

$$s(n,e,m) \mid \lambda(\lambda(n)). \tag{19}$$

Proof. Note that for the sequence (18) we have $c_i = m^{e^i} \bmod n$. For $s(n,e,m)$ we thus have

$$m^{e^{s(n,e,m)}} = m^e \bmod n. \tag{20}$$

By (54) we have $e^{s(n,e,m)} = e \bmod \lambda(n)$, which implies by definition 3.3 that $\mathrm{ord}_{\lambda(n)}(e) = s(n,e,m)$. Relation (16) yields the assertion. □

Example 3.5 Let be $e = 7$, $n = 55 = 5 \cdot 11$. Then we have $\lambda(55) = 20$, and $\lambda(\lambda(55)) = 4$. Denoting $c_0 = 51$, we obtain

$$\begin{aligned} c_1 &= 51^7 \bmod 55 = 6 \\ c_2 &= 6^7 \bmod 55 = 41 \\ c_3 &= 41^7 \bmod 55 = 46 \\ c_4 &= 46^7 \bmod 55 = 51 = c_0 \end{aligned}$$

Hence, the period of the cycle which 51 belongs to is $s(n,e,m) = 4$. Note by (19) that this is the maximum value. Analogously, there are the following cycles.

9 fixed points	(0), (1), (10), (11), (21), (34), (44), (45), (54)	
3 cycles of period 2	(12, 23), (22, 33), (32,43)	
10 cycles of period 4	(2, 18, 17, 8),	(3, 42, 48, 27)
	(4, 49, 14, 9),	(5, 25, 20, 15)
	(6, 41, 46, 51),	(7, 28, 32, 13)
	(16, 36, 31, 26),	(19, 24, 29, 39)
	(30, 35, 40, 50),	(37, 38, 47, 53)

Definition 3.6 Let be $m, n \in \mathbb{Z}$, $n \geqq 0$. Then we define the function

$$\omega_m(n) = \begin{cases} \mathrm{ord}_n(m) & \text{if } \gcd(m,n) = 1, \\ 0 & \text{if } \gcd(m,n) \neq 1. \end{cases} \tag{21}$$

It is obvious that $m^{\omega_m(n)} = 1 \bmod n$ for any $m, n \in \mathbb{Z}$, $n \geqq 0$ (since this is the definition of the order function). Substituting n by $\omega_m(n)$ immediately yields

$$m^{\omega_m(\omega_m(n))} = 1 \bmod \omega_m(n). \tag{22}$$

Here "$a = b \bmod 0$" has to be understood as a congruence in $\mathbb{Z}$, i.e. as "$a = b$." By iteration, we obtain the cascading-ω equation

$$m^{\omega_m^{(r)}(n)} = 1 \bmod \omega_m^{(r-1)}(n), \qquad \text{where } r \geqq 1. \tag{23}$$

where $\omega_m^{(r)}(n) = \omega_m(\omega_m(\ldots(\omega_m(n))\ldots))$ denotes the r-fold composition of ω_m.

Theorem 3.7 *Let be $d, e, n \in \mathbb{N}$, such that $n > 1$, $\gcd(e,n) = 1$, and $d \cdot e = 1 \bmod \lambda(n)$. Then $\omega_e(\omega_e(n)) > 0$, and*

$$d = e^{\omega_e(\omega_e(n))-1} \bmod \omega_e(n). \tag{24}$$

Proof. First we note by (16) that $\omega_e(n) \mid \lambda(n)$. Therefore, $de = 1 \bmod \lambda(n)$ implies

$$d \cdot e = 1 \bmod \omega_e(n). \tag{25}$$

(If $de - 1 = k\lambda(n)$ for a $k \in \mathbb{Z}$, then $de - 1 = k'\omega_e(n)$, where $k' = k\lambda(n)/\omega_e(n)$.) If we had now $\omega_e(\omega_e(n)) = 0$, then e would divide $\omega_e(n)$ and hence $\lambda(n)$: But then there would be no d with $de = 1 \bmod \lambda(n)$. Hence, $\omega_e(\omega_e(n)) > 0$. Moreover, by the cascading-ω equation (22) we have

$$e^{\omega_e(\omega_e(n))-1} \cdot e = 1 \bmod \omega_e(n). \tag{26}$$

Equation (24) follows immediately from (25) and (26). □

Example 3.8 Let be $n = 221$ and $e = 11$. Then $\omega_{11}(221) = 48$, $\omega_{11}(48) = 4$, hence

$$d = 11^3 = 35 \bmod 48.$$

Therefore, the possible $d < 221$ are $d = 35, 83, 131, 179$. In fact, $221 = 13 \cdot 17$, and $\lambda(221) = 48$; this means that $11 \cdot 35 = 1 \bmod \lambda(221)$, or $d = 35$.

The two shoulders on which Theorem 3.7 rests are equations (25) and (26). They can be extended to analogues for the following corollary.

Corollary 3.9 *Let be $e, n, a, b \in \mathbb{N}$ such that $n > 1$ and $\gcd(e,n) = 1$, as well as $\lambda(n) \mid \omega_e(a\omega_e(n))$. Then the integer*

$$\tilde{d} = e^{b\omega_e(a\omega_e(n))-1} \bmod a\omega_e(n) \tag{27}$$

satisfies $\tilde{d}e = 1 \bmod a\omega_e(n)$, and for any number $m \in \mathbb{Z}_n$ we have

$$m^{e\tilde{d}} = m \bmod n. \tag{28}$$

If the integer a is such that $\omega_e(a\omega_e(n)) \mid \lambda(n)$, then the unique $d < \lambda(n)$ with $de = 1 \bmod \lambda(n)$ is related to $\tilde{d}$ by

$$d = \tilde{d} \bmod a\omega_e(n). \tag{29}$$

Proof. Substituting n by $a\omega_e$, from $e^{\omega_m(n)} = 1 \bmod n$ for any $m \in \mathbb{Z}$ we deduce that $e^{b\omega_e(a\omega_e(n))} = 1 \bmod a\omega_e(n)$. Especially, with (27) we have

$$\tilde{d} \cdot e = e^{b\omega_e(a\omega_e(n))-1} \cdot e = 1 \bmod a\omega_e(n). \tag{30}$$

If $\lambda(n) \mid \omega_e(a\omega_e(n))$, we have $m^{\tilde{d}e \bmod a\omega_e(n)} \bmod n = m^{1 \bmod a\omega_e(n)} \bmod n = m^{1 \bmod \lambda(n)} \bmod n = m \bmod n$. (Note that $\lambda(n)$ enters the scene in the second last equation to fulfill the equation for *all* m!) In turn, if $\omega_e(a\omega_e(n)) \mid \lambda(n)$, then $\tilde{d}e = 1 \bmod \lambda(n)$ implies $\tilde{d}e = 1 \bmod a\omega_e(n)$; thus (29) follows from (30). □

Example 3.10 Let be $n = 143$ and $e = 47$. Then $\omega_{47}(143) = 20$, and with $a = 2$, $b = 3$, we have $3\omega_{47}(40) = 12$, hence

$$d = 47^{11} = 23 \bmod 40.$$

Therefore, $m^{ed} = m^{1081} = m \bmod 143$. In fact, $143 = 11 \cdot 13$, and $\lambda(143) = 60$; this means that $47 \cdot 23 = 1 \bmod \lambda(143)$, or $d = 23$.

Remark 3.11 Given two relatively prime integers e and n, corollary 3.9 enables us to choose an (almost) arbitrary multiple of the order $\mathrm{ord}_n(e) > 0$ to find an integer d being a kind of "inverse" of e: If the multiple is small enough such that it divides $\lambda(n)$, our result supplies a list of values, one of which satisfies $ed = 1 \bmod \lambda(n)$; if the multiple is also a multiple of $\lambda(n)$, we can compute $\tilde{d}$ such that $\tilde{d}e = 1 \bmod a\,\mathrm{ord}_n(e)$. In particular, by (47) and (16) the Euler function is a multiple of both $\lambda(n)$ and $\mathrm{ord}_n(e)$.

3.2 Properties of composed numbers $n = pq$

Let be p, q be two primes, $p \neq q$. Then $n = pq$ is an integer composed of two primes. Among the integers n less than 50 there are 13 ones composed of two primes, $n = pq$, whereas less than 100 there are 30 ones, shown in the following tables.

n	6	10	14	15	21	22	26	33	34	35	38	39	46	51	55
$\varphi(n)$	2	4	6	8	12	10	12	20	16	24	18	24	22	32	40
$\lambda(n)$	2	4	6	4	6	10	12	10	16	12	18	12	22	16	20

n	57	58	62	65	69	74	77	82	85	86	87	91	93	94	95
$\varphi(n)$	36	28	30	48	44	36	60	40	64	42	56	72	60	46	72
$\lambda(n)$	18	28	30	12	22	36	30	40	16	42	28	12	30	46	36

Let us now study the geometric structure of the Euler function.

Theorem 3.12 *Let $n = pq$ be a positive integer, composed of two primes p and q with $p < q$. For any integer $p_{\min} \in \mathbb{N}$ satisfying $p_{\min} \leqq p$ we then have*

$$\varphi(n) \geqq (p_{\min} - 1)\left(\frac{n}{p_{\min}} - 1\right). \tag{31}$$

The inequality is strict, if $p_{\min} < p, q$.

Proof. We have $\varphi(n) = (p-1)\left(\dfrac{n}{p}-1\right)$, and $\varphi(n)$ is a function of p:

$$g(p) = \varphi(n) = n - p - \frac{n}{p} + 1.$$

Since $g'(p) = -1 + n/p^2 < 0$, for fixed n the function g is strictly decreasing with respect to p, as long as $p < q$, i.e. as $n/p^2 > 1$. □

Geometrically, this result means that in the graph of $\varphi(n)$ the point $(n, \varphi(n))$ lies above the "Euler function ray" (see Figure 4)

$$f_p(x) = \left(x, (p-1)\left(\frac{x}{p}-1\right)\right). \tag{32}$$

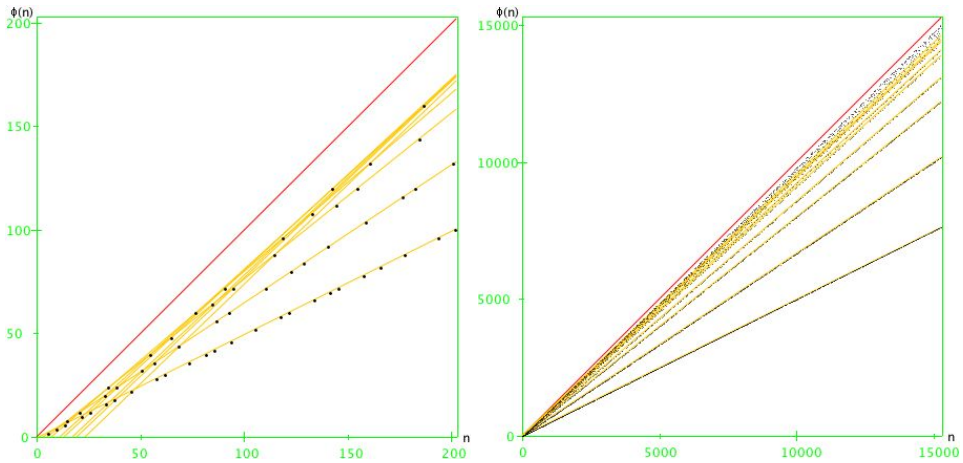

Figure 4: Plot of the Euler function $\varphi(pq)$, with p, q prime; also sketched are the rays f_p for $p = 2$, 3, 5, 7, 11, 13, 17, 19, 23.

Theorem 3.13 *Let be p, q two primes $p < q$, e an integer with $e > 1$, and $n = pq$. Moreover define for $a \in \mathbb{N}$ the exponents $\delta_{e,n}$, $\gamma_{e,a} \in \mathbb{N}$ by*

$$\delta_{e,n} = \max\{i \in \mathbb{N} : \ e^i \leqq n\} = \left\lfloor \frac{\ln n}{\ln e} \right\rfloor, \qquad \gamma_{e,a} = \max\{i \in \mathbb{N} : \ e^i \mid a\}, \tag{33}$$

as well as

$$r_\pm = \frac{1}{2}\left(\Delta \pm \sqrt{\Delta^2 - 4n}\right) \qquad \text{with } \Delta = p + q + \delta_{e,n}. \tag{34}$$

Then for any integers b, $r \in \mathbb{N}$, $r_- \leqq r \leqq p$ or $q \leqq r \leqq r_+$, satisfying

$$b e^{\lfloor f(r) \rfloor} = 1 \bmod n, \qquad \text{where} \quad f(r) = (r-1)\left(\frac{n}{r}-1\right), \tag{35}$$

the Euler function value $\varphi(n)$ can be computed by

$$\varphi(n) = \gamma_{e,b} + \lfloor f(r) \rfloor \tag{36}$$

Proof. Note first that real values for $r_\pm$ always exist since the term in the square root is positive, $\Delta^2 > (p+q)^2$, i.e. $\Delta^2 - 4n > (q-p)^2 > 0$. We see that $(p-1)(q-1) - (r-1)(n/r-1) = \frac{1}{r}(r^2 - \Delta r + n)$. Solving this quadratic equation with respect to r, straightforward calculation thus shows that the inequalities for r are equivalent to the inequalities

$$0 \leqq (p-1)(q-1) - (r-1)\left(\frac{n}{r} - 1\right) \leqq \delta_{e,n}, \tag{37}$$

which means that $0 \leqq \varphi(n) - (r-1)\left(\frac{n}{r} - 1\right) \leqq \delta_{e,n}$. On the other hand, b being the multiplicative inverse of e^r by the modular equation in (35), we have $b = e^j \bmod n$ for some $j \in \mathbb{N}$, in particular for $j = \varphi(n) - r$. But if $j < \delta_{e,n}$, we have $b = e^j$, and $j = \gamma_{e,b}$. □

Example 3.14 Let be $p = 11$, $q = 13$, and $e = 7$. Then $\delta_{7,143} = 2$, and thus $\Delta = 26$, $r_\pm = 13 \pm \sqrt{26}$. So r shall satisfy $8 \leqq r \leqq 11$ or $13 \leqq r \leqq 18$. For $r = 8$, e.g., we have

$$(r-1)\left(\frac{n}{r} - 1\right) = 7 \cdot 16.875 = 118.125;$$

Since $7^{118} = 108 \bmod 143$, we achieve by the extended Euclidean algorithm $b = 49 = 7^2$ (because $1 = 49 \cdot 108 - 37 \cdot 143$), and with $\gamma_{7,49} = 2$ we obtain

$$\varphi(143) = 118 + \gamma_{7,49} = 120.$$

In fact, $\varphi(143) = 10 \cdot 12$. □

Example 3.15 Let be $p = 3\,336\,670\,033$, $q = 9\,876\,543\,211$, and $e = 2$. Then

$$n = 32\,954\,765\,761\,773\,295\,963,$$

$\delta_{2,n} = 64$, and thus

$$\Delta = 13\,213\,213\,308, \quad r_- = 3\,336\,670\,000.3, \quad r_+ = 9\,876\,543\,307.6.$$

For $r = 9\,876\,543\,308$, e.g., we have

$$i = (r-1)\left(\frac{n}{r} - 1\right) = 32\,954\,765\,748\,560\,082\,656.$$

Since

$$2^i = 7\,542\,048\,005\,965\,299\,043 \bmod n,$$

we achieve by the extended Euclidean algorithm

$$b = 18\,446\,744\,073\,709\,551\,616$$

and with $\gamma_{2,b} = 64$ we obtain

$$\varphi(n) = i + \gamma_{2,b} = 32\,954\,765\,748\,560\,082\,720.$$

□

The following lemma tells us the grade of "coarse graining," i.e., a step-width that a systematic and definite search for an appropriate Euler function ray factor r must use.

Lemma 3.16 *Let p, q be two primes, $p < q$, e an integer $e > 1$, and $n = pq$. Moreover let r_+ and $\delta_{e,n}$ be defined as in theorem 3.13 by equations (33) and (34). Then*

$$r_+ - q > \frac{\delta_{e,n}}{2}. \tag{38}$$

Moreover,

$$p - r_- > \frac{\delta_{e,n}}{2} \qquad \text{if } \delta_{e,pq} < \tfrac{2}{3}(3p-q). \tag{39}$$

Proof. By $\Delta^2 - 4pq = (q-p)^2 + 2(p+q)\delta_{e,pq} + \delta_{e,pq}^2$ we achieve for $\delta_{e,pq} > 0$

$$\begin{aligned} r_+ &= \frac{1}{2}\left(\Delta + \sqrt{\Delta^2 - 4pq}\right) = \frac{1}{2}\left(\Delta + \sqrt{(q-p)^2 + 2(p+q)\delta_{e,pq} + \delta_{e,pq}^2}\right) \\ &> \frac{1}{2}(\Delta + q - p) = \frac{1}{2}(2q + \delta_{e,pq}) = q + \frac{\delta_{e,pq}}{2}. \end{aligned}$$

Analogously, by (39) we have $2(q-p) + \frac{3}{2}\delta_{e,pq} < q + p$, i.e. $(q-p)^2 + 2(q+p)\delta_{e,pq} + \delta_{e,pq}^2 > (q-p)^2 + 4(q-p)\delta_{e,pq} + 4\delta_{e,pq}^2 = (q - p + 2\delta_{e,pq})^2$, i.e.

$$\begin{aligned} r_- &= \frac{1}{2}\left(\Delta - \sqrt{(q-p)^2 + 2(p+q)\delta_{e,pq} + \delta_{e,pq}^2}\right) \\ &< \frac{1}{2}\left(\Delta - \sqrt{(q-p)^2 + 4(p-q)\delta_{e,pq} + 4\delta_{e,pq}^2}\right) \\ &= \frac{1}{2}(\Delta - q + p - 2\delta_{e,pq}) = p - \frac{\delta_{e,pq}}{2}. \end{aligned}$$

□

3.3 The algorithm

An algorithm to break an RSA cryptosystem is shown below in pseudocode. It is invoked with the public key (e,n) and the estimate r for the Euler function ray as input parameters and returns a possible private RSA key parameter d corresponding to e. If it fails, $d \leqq 0$ is returned.

```
long rayAttack ( e, n, r ) {
  // store an array a such that a[i] = m^(2^i) < n:
  a[0] = e;
  j = 1;
  while ( a[j-1] < n ) {
     a[j] = a[j-1] * a[j-1];
     j++;
  }
  delta = 0;
  while ( e^(delta + 1) <= n )  delta++;
  step = delta / 2;
  d = 0; r = n^(1/2);
  while ( d == 0 && r > 0 ) {
    ord = omega(e,n,r);
    if ( ord > 0 )  d = euclid( e, ord )[0];
    else  r -= step;
  }
  return d;
}
```

The heart of algorithm *rayAttack* is the algorithm $\omega(m,n,r)$ determining an integer i being a multiple of $\mathrm{ord}_n(e)$ on the basis of corollary 3.9. Both algorithms use the extended Euclidean algorithm *euclid*. In detail:

```
/** returns minimum i >= (r - 1) * (n/r - 1) such that m^i = 1 mod n
 *  returns 0 if i is not computable, and -1 if the algorithm fails
 */
long omega( m, n, r ) {
  if ( gcd(m,n) != 1 )  return  0;
  else {
    i = (r - 1) * (n/r - 1);
    m = m % n;
    // determine b such that b * m^i = 1 mod n:
    b = euclid (n, ( m^i % n ) )[1] mod n;
    // determine maximum exponent gamma such that m^gamma divides b:
    gamma = 0;
    for ( k = a.length - 1; k >= 0; k-- ) {
      if ( b >= a[k] ) {
        if ( b % a[k] == 0 ) {
          gamma += 2^k;
          b /= a[k];
        }
        else break;  // not a power of e
      }
    }
    i += gamma;
    if ( i > 0 &&  b != 1 ) {
      i = - 1;  // algorithm fails!
    }
    return i;
  }
}
```

The classical Euclidean algorithm reads:

```
// euclid(m,n) = extended Euclidean algorithm
// returning x0, x1 s.t. gcd(m,n) = x0 * m + x1 * n:
long[] euclid( long m, long n) {
  x[] = {1,0};
  u = 0, v = 1;
  mNegative = false, nNegative = false;

  if ( m < 0 ) { m = -m; mNegative = true; }
  if ( n < 0 ) { n = -n; nNegative = true; }
  while ( n > 0 ) {
    // determine q and r such that m = qn + r:
    q = m / n; r = m % n;
    // replace:
    m = n; n = r;
    tmp = u; u = x[0] - q*u; x[0] = tmp;
    tmp = v; v = x[1] - q*v; x[1] = tmp;
  }
  if  ( mNegative ) x[0] = -x[0];
  if  ( nNegative ) x[1] = -x[1];
  return x;
}
```

3.3.1 Complexity analysis

First we note that the running time $T_{\text{euclid}}(m,n)$ of Euclid's algorithm for two input integers m,n is given by

$$T_{\text{euclid}}(m,n) = \log_\phi[(3-\phi)\cdot\max(m,n)], \tag{40}$$

where ϕ is the golden ratio $\phi = (1+\sqrt{5})/2$, see [5, §4.5.3, Corollary L (p.360)]. If we consider, to simplify, the running time as the number of loops to be performed, we therefore we achieve for the running time $T_\omega(m,n,r)$ of the ω-function $T_\omega(m,n,r) = T_{\text{pow}}(m,\lfloor f(r)\rfloor) + T_{\text{euclid}}(n, m^{\lfloor f(r)\rfloor} \bmod n) + \frac{1}{2}\log_m n + \frac{1}{2}\log_m n$, i.e. [2, §2.12]

$$T_\omega(m,n,r) = \log_2\lfloor f(r)\rfloor\cdot(\log_2 n)^2 + \log_\phi[(3-\phi)n] + \log_m n. \tag{41}$$

Since the complexity $T_{\text{ray}}(e,pq,r)$ of the ray Attack algorithm (with $n=pq$) then is given by

$$T_{\text{ray}}(e,pq,r) = \frac{r-p}{\log_e pq}T_\omega(e,pq,r) + T_{\text{euclid}}(e,\omega(e,pq,r)),$$

and since by $\omega(e,pq,r) < n$ we have $T_{\text{euclid}}(e,\omega(e,pq,r)) < T_{\text{euclid}}(e,pq)$, we obtain

$$\begin{aligned} T_{\text{ray}}(e,pq,r) &< \left(\frac{r-p}{\log_e pq}+1\right)\log_\phi[(3-\phi)\,pq] \\ &\quad + (r-p)\left(1+\frac{\log_2\lfloor f(r)\rfloor\cdot(\log_2 pq)^2}{\log_e pq}\right) \\ &= O((r-p)\ln r\cdot\ln e\cdot\ln pq)\,. \end{aligned} \tag{42}$$

(Note that $f(r) = O(r)$.)

4 Discussion

In this article a new ansatz to attack RSA cryptosystems is described, basing on geometric properties of the Euler functions, the *Euler function rays*. However, a resulting algorithm turns out to be inefficient. It essentially consists of a loop with starting value determined by the Euler function ray and with step width given by a function $\omega_e(n)$ being a multiple of the order $\text{ord}_n(e)$, where e denotes the public key exponent and n the RSA modulus. For $n=pq$ and an estimate $r < \sqrt{pq}$ for the smaller prime factor p, the running time is given by $T(e,n,r) = O((r-p)\ln e\ln n\ln r)$.

In other words, this attack is queuing up into a long series of failed attacks on RSA. So, what is gained in the end? First, we achieved a small mathematical novelty, the Euler function rays, i.e. geometrical properties of the Euler function. To my knowledge they have never been mentioned before. Second, the ω-function has been introduced, being closely related to the order of a number but being more appropriate for practical purposes. Finally, this trial as another failure in fact is good news. It seems that e-commerce basing on RSA can go on.

A Appendix

A.1 Euler's Theorem

If n is a prime, the set of all numbers (more exactly: of all residue classes) modulo n is a field with respect to addition and multiplication, as is well known. However, if n is a composite

integer, the ring of all numbers modulo n is not a field, because the cancellation of a number (more exactly: a congruence) modulo n by any divisor d of n also requires the corresponding cancellation of n, and thus carries us from the ring modulo n to another ring, namely modulo n/d. In this case, d is said to be a zero divisor of the ring, since $d|n$ and $n = n/d = 0 \bmod n/d$. For instance, for $n = 9$ the congruence

$$15 = 6 \bmod 9$$

is cancelled by $d = 3$ through

$$\frac{15}{d} = \frac{6}{d} \bmod \frac{9}{\gcd(d,9)}, \qquad \text{or} \qquad 5 = 2 \bmod 3.$$

However, if we avoid the zero divisors of n and consider only the those numbers (more exactly: primitive residue classes) a mod n with $\gcd(a,n) = 1$, then all divisions by *these* elements can be uniquely performed. For example, by $\gcd(5,12) = 1$

$$5x = 10 \bmod 12 \qquad \Longleftrightarrow \qquad x = 2 \bmod 12.$$

These numbers actually constitute a multiplicative group of order $\varphi(n)$:

Definition A.1 For $n \in \mathbb{N}$, $n > 1$, *Euler's φ-function* or *totient function* assigns to n the number $\varphi(n)$ of positive integers $k < n$ relatively prime to n, i.e.

$$\varphi(n) = \#\mathbb{Z}_n^*, \qquad \text{where} \qquad \mathbb{Z}_n^* = \{k \in \mathbb{N} : k < n \text{ and } \gcd(k,n) = 1\}. \tag{43}$$

$\mathbb{Z}_n^*$ is the multiplicative group modulo n. For instance, the set of numbers less than 12 and relatively prime to 12 are $\{1, 5, 7, 11\}$, and thus $\varphi(12) = 4$. An explicit formula denotes

$$\varphi(n) = p_1^{\alpha_1 - 1} \cdots p_r^{\alpha_r - 1} \cdot (p_1 - 1) \cdots (p_r - 1) = n \cdot \prod_{p|n} \left(1 - \frac{1}{p}\right), \tag{44}$$

if the prime factorization of n is given by $n = p_1^{\alpha_1} \cdots p_r^{\alpha_r}$. E.g., $12 = 2^2 \cdot 3$, and

$$\varphi(12) = 2 \cdot 2 = 12 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{3}\right) = 4.$$

Theorem A.2 (Euler's Theorem) *If* $\gcd(m,n) = 1$, *then*

$$m^{\varphi(n)} = 1 \bmod n. \tag{45}$$

For a proof see, e.g., [7, §4.1].

A.2 The Carmichael function and Carmichael's Theorem

Euler's Theorem can be strengthened. As we will see, this will yield an efficient determination of key pairs of a RSA public key cryptosystem, much more efficient than the originally (and yet nowadays in many textbooks) proposed procedure based on Euler's Theorem.

Definition A.3 For $n \in \mathbb{N}$ let $n = \prod_{i=1}^{r} p^{\alpha_i}$ be its prime factorisation. Then the *Carmichael*[1] *function* λ is given by $\lambda(n) = \operatorname{lcm}[\lambda(p_i^{\alpha_i})]_i$, where for each $i = 1, \ldots, r$,

$$\lambda(p_i^{\alpha_i}) = \begin{cases} 2^{\alpha_i - 2} & \text{if } p_i = 2 \text{ and } \alpha_i \geqq 3, \\ p_i^{\alpha_i - 1}(p_i - 1) & \text{otherwise.} \end{cases} \tag{46}$$

For $n > 2$, $\lambda(n)$ is even (since $p_i - 1$ as an even integer divides $\lambda(n)$); for $n = 2$, we have simply $\lambda(2) = \varphi(2) = 1$. Moreover, since $\lambda(n)$ is the least common multiple of factors of $\varphi(n)$, it divides the Euler totient function:

$$2 \mid \lambda(n) \mid \varphi(n) \qquad \text{for } n > 2. \tag{47}$$

Theorem A.4 (Carmichael's Theorem) *If $m, n \in \mathbb{N}$ and $\gcd(m,n) = 1$, then*

$$m^{\lambda(n)} = 1 \bmod n. \tag{48}$$

Moreover, $\lambda(n)$ is the smallest exponent with this property.

Using Carmichael's Theorem, we have a way of explicitly writing down the quotient of two residue classes $a/b \bmod n$. The formula is

$$\frac{a}{b} = ab^{-1} = ab^{\lambda(n)-1} \bmod n, \qquad \text{if } \gcd(b,n) = 1, \tag{49}$$

i.e. $b^{-1} = b^{\lambda(n)-1} \bmod n$.

Example A.5 For $n = 65\,520 = 2^4 \cdot 3^2 \cdot 5 \cdot 7 \cdot 13$, Euler's function assumes the value $\varphi(n) = 8 \cdot 6 \cdot 4 \cdot 6 \cdot 12 = 13\,824$, while $\lambda(n) = \operatorname{lcm}(4, 6, 4, 6, 12) = 12$. For all m with $\gcd(m,n) = 1$ we thus have

$$m^{12} = 1 \bmod 65\,520.$$

For each m with $\gcd(b,n) = 1$ we have $m^{-1} = m^{11} \bmod 65\,520$. For instance,

$$\frac{1}{11} = 11^{11} = 47\,651 \bmod 65\,520.$$

Theorem A.6 *If $n \in \mathbb{N}$ is a product of distinct primes, i.e. $n = \prod_i p_i$, then*

$$m^{\lambda(n)+1} = m \bmod n \qquad \textit{for all } m \in \mathbb{Z}. \tag{50}$$

For a proof see, e.g., [8, §A2].

If the multiplicative group $\mathbb{Z}_n^* = \{m : 1 \leqq m, \gcd(m,n) = 1\}$ decomposes into the subgroups G_i,

$$\mathbb{Z}_n^* = G_1 \times G_2 \times \ldots \times G_k, \tag{51}$$

and if d_i is the order of the group G_i, then each element $m \in \mathbb{Z}_n^*$ can be written in the form

$$m = g_1^{e_1} g_2^{e_2} \cdots g_k^{e_k} \qquad \text{with } 1 \leqq e_i \leqq d_i. \tag{52}$$

[1] Robert D. Carmichael (1879 – 1967), U.S. mathematician

Furthermore, for each i,

$$g_i^{d_i} = 1 \bmod n, \quad \text{with } d_i | \lambda(n). \tag{53}$$

For instance, $\mathbb{Z}_{15} = \{1,2,4,7,8,11,13,14\}$. We see that $\varphi(15) = 8 = \#\mathbb{Z}_{15}$. All possible subgroups G_i of $\mathbb{Z}_{15}$ are the following ones.

$$G_1 = \{1\},\ G_2 = \{1,4\},\ G_3 = \{1,11\},\ G_4 = \{1,14\},$$

$$G_5 = \{1,2,4,8\},\ G_6 = \{1,4,7,13\}.$$

Hence $d_1 = 1$, $d_2 = d_3 = d_4 = 2$, and $d_5 = d_6 = 4$. They all divide $\lambda(15) = 4$.

Corollary A.7 *Let be $e, m, n \in \mathbb{N}$, $n > 1$, and either n a product of distinct primes, or* $\gcd(m,n) = 1$. *Then for all $e \in \mathbb{N}$*

$$m^e = m^{e \bmod \lambda(n)} \bmod n. \tag{54}$$

Lemma A.8 *For $n \in \mathbb{N}$,*

$$\lambda(n) \leqq n - 1. \tag{55}$$

Proof. Because $\lambda(p) < p$ for every prime, $\lambda(n) < n$ as the least common multiple of the Carmichael function values of the prime factors of n. □

References

[1] F. L. Bauer. *Decrypted Secrets. Methods and Maxims of Cryptology*. 2nd edition, Springer-Verlag, Heidelberg und Berlin 2000.

[2] J. A. Buchmann. *Introduction to Cryptography*. Springer-Verlag, New York 2001.

[3] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. McGraw-Hill, New York 1990.

[4] W. Diffie and M. E. Hellman. 'New directions in cryptography'. *IEEE Trans. Inform. Theory*, **22** (6), 644–654, 1976.

[5] D. E. Knuth. *The Art of Computer Programming. 3rd Sorting and Searching*. 3rd edition, Addison-Wesley, Reading, 1998.

[6] N. Koblitz. *A Course inn Number Theory and Cryptography*. 2nd edition, Springer-Verlag, New York 1994.

[7] F. Padberg. *Elementare Zahlentheorie*. Spektrum Akademischer Verlag, Heidelberg Berlin, 2nd edition, 1996.

[8] H. Riesel. *Prime Numbers and Computer Methods for Factorization*. Birkhäuser, Boston 1994.

[9] R. L. Rivest, A. Shamir, and L. M. Adleman. 'A method for obtaining digital signatures and public-key cryptosystems'. *Comm. ACM*, **21**, 120–126, 1978.

[10] A. de Vries. 'The ray attack on RSA cryptosystems', in R. Muno (ed.), *Jahresschrift der Bochumer Interdisziplinren Gesellschaft eV 2002*. ibidem-Verlag, Stuttgart 2003

WebLinks

1. http://math-it.org/mathematics : Learn more about RSA and number theory interactively
2. http://www.rsasecurity.com/rsalabs/rsa_algorithm : Homepage of RSA labs; contains the latest RSA challenge numbers