CRYPTANALYSIS OF THE DICKSON-SCHEME *

Winfried B. Müller and Rupert Nöbauer
Institut für Mathematik
Universität Klagenfurt
A - 9010 Klagenfurt, Austria

1. Introduction

In Müller and W. Nöbauer (1981) a new public-key cryptosystem was introduced. Similar to the well-known RSA-scheme, the plaintext alphabet and the code alphabet of this cryptosystem are given by Z/(n), the ring of residue classes of the integers Z modulo a natural number n. In contrast to the RSA-scheme, however, n need not be squarefree, but can be an arbitrary positive integer. The encryption polynomials x^k of the RSA-scheme are replaced by another class of polynomials, namely by the so-called Dickson-polynomials. We call this cryptosystem the Dickson-scheme.
So far, there is not known very much about the security of the Dickson-scheme. The goal of this paper is to perform a cryptanalysis of the Dickson-scheme. We start with some basic facts on Dickson-polynomials, outline a fast algorithm for the computation of function values for the Dickson-polynomials and then give a short description of the Dickson-scheme. Afterwards, several possible cryptanalytic attacks on the system are discussed and as a consequence requirements to the key parameters are formulated, which guarantee the system to be secure from the described attacks.

---

*) The work presented in this paper was supported by the Österreichischen Fonds zur Förderung der Wissenschaftlichen Forschung under FWF-Project No. P 5452.
The final version of this paper was prepared during a visiting appointment of the author W.B. Müller in the Department of Mathematics at Monash University, Clayton, Vic. 3168, Australia.

F. Pichler (Ed.): Advances in Cryptology - EUROCRYPT '85, LNCS 219, pp. 50-61, 1986.

2. Some basic facts

Let R be a commutative ring with identity, and let $a \in R$. The Dickson-polynomial $g_k(a,x) \in R[x]$ of degree k is given by

$$g_k(a,x) = \sum_{i=0}^{[k/2]} \frac{k}{k-i} \binom{k-i}{i} (-a)^i x^{k-2i},$$

where [k/2] denotes the greatest integer $i \leq k/2$.
If R_1 is an extension ring of R and if $u \in R_1$ is a unit, then the equation

$$(1) \qquad g_k(a, u + \tfrac{a}{u}) = u^k + (\tfrac{a}{u})^k$$

holds, as can be proved by using Waring's inversion formula (cf. Lidl and Niederreiter (1983)).
In this paper we restrict ourselves to the case $a = 1$ and write $g_k(1,x) =: g_k(x)$. Since for $a = 1$ from (1) the functional equation $g_k(x) \circ g_t(x) = g_{kt}(x)$ follows, the Dickson-polynomials $g_k(x)$ are closed under composition.

In order to use Dickson-polynomials in public-key cryptography, we put $R = Z/(n)$. The plaintext messages $m \in Z/(n)$ are encrypted by $m \to g_k(m) \bmod n$.
If the factorization of n is given by $n = \prod_{i=1}^{r} p_i^{e_i}$, then in the Dickson-scheme the number [1]

$$v(n) = [p_1^{e_1-1}(p_1^2-1), p_2^{e_2-1}(p_2^2-1), \ldots, p_r^{e_r-1}(p_r^2-1)]$$

plays the same role as the number $w(n) = [p_1-1, p_2-1, \ldots, p_r-1]$ for a squarefree n in the RSA-scheme. For example, whereas the power polynomial x^k induces a permutation of $Z/(n)$ for a squarefree n, iff[2] $(k,w(n)) = 1$, the Dickson-polynomial $g_k(x)$ induces a permutation of $Z/(n)$, n arbitrary, iff $(k,v(n)) = 1$ (cf. W. Nöbauer (1965)). Another obvious analogy to the RSA-scheme is given by the following fact:
If the permutation π of $Z/(n)$ is induced by a Dickson-polynomial $g_k(x)$, then π^{-1} is also induced by a Dickson-polynomial, namely by $g_t(x)$, where $kt \equiv 1 \bmod v(n)$ (cf. Lausch, Müller and W. Nöbauer (1973)).

---

1) By $[a_1,\ldots,a_r]$ we denote the least common multiple of the integers $a_1,\ldots,a_r$.
2) By $(a_1,\ldots,a_r)$ we denote the greatest common divisor of the integers $a_1,\ldots,a_r$.

Thus, exactly like in the RSA-scheme, the trapdoor information of the Dickson-scheme consists in the factorization of n: All known methods for computing the inverse of an encryption function $x \to g_k(x) \bmod n$ need the prime factor decomposition of n.

3. A fast evaluation algorithm for Dickson-polynomials

We now give an evaluation algorithm of complexity $O(\mathrm{ld}(k))$, which permits to calculate function values of $g_k(x)$ (cf. also R. Nöbauer (1985/86)). Given $b \in Z/(n)$, we want to compute $g_k(b) \bmod n$. For doing this, we have to solve

(2) $$u + \frac{1}{u} = b,$$

or equivalently

(3) $$u^2 - bu + 1 = 0$$

in some extension ring of $Z/(n)$.
As can be seen easily, the factor ring $R_b = Z/(n)[u]/(u^2-bu+1)$ is an extension ring of $Z/(n)$, and every element $s \in R_b$ can be represented uniquely in the form

$$s = a_1 u + a_0, \quad a_0, a_1 \in Z/(n).$$

Multiplication in R_b can be implemented by using the formula

(4) $$(a_1u+a_0)(b_1u+b_0) = (a_1b_0+a_0b_1+a_1b_1b)u + a_0b_0 - a_1b_1.$$

Obviously, the element $u \in R_b$ is a solution of (3). Since $u(b-u) = 1$, u is always invertible.
Now, for the evaluation of $g_k(b)$ just calculate the power u^k in the ring R_b by using the "square- and multiply-technique": That is, first compute

$$u,\ u^2,\ (u^2)^2,\ \ldots$$

and then multiply together the appropriate factors, thus finding elements $a_0, a_1 \in Z/(n)$ with

$$u^k = a_1 u + a_0.$$

Since u^{-1} also satisfies (3), the equation

$$\frac{1}{u^k} = a_1 \frac{1}{u} + a_0$$

holds, and therefore

$$g_k(b) = g_k(u + \frac{1}{u}) = u^k + \frac{1}{u^k} = a_1(u + \frac{1}{u}) + 2a_0 = a_1 b + 2a_0.$$

The number of required steps is $O(\mathrm{ld}(k))$.

We summarize our procedure in the following

Algorithm 1:

Input n, k, b

Compute $a_0, a_1 \in Z/(n)$ with $u^k \equiv a_1 u + a_0 \bmod u^2 - bu + 1$.

Comment [use the square-and multiply-technique].

Compute $g_k(b) \equiv a_1 b + 2a_0 \bmod n$.

End.

4. The Dickson-scheme

Every participant C of the communication network chooses a positive integer $r_C := r$, r odd prime powers $p_i^{e_i}$ (if also a power 2^e is chosen, the following formulas have to be modified slightly), and an encryption key $k_C := k$ with $(k, p_i^{e_i - 1}(p_1^2 - 1)) = 1$ for $i = 1, 2, \ldots, r$.

Then C calculates the numbers

$n_C := n = \prod_{i=1}^{r} p_i^{e_i}$, $v(n) = [p_1^{e_1 - 1}(p_1^2 - 1), \ldots, p_r^{e_r - 1}(p_1^2 - 1)]$, and computes a decryption key $t_C := t$, that is a natural number satisfying the linear congruence

(5) $$kt \equiv 1 \mod v(n).$$

The public key of C consists in the parameters n and k, and the secret key is given by the prime factorization of n and by t.

If A intends to send the secret message $m \in Z/(n_B)$ to B, he has to encrypt m by calculating $c \equiv g_{k_B}(m) \bmod n_B$ and then he sends c to B. The receiver B decrypts c by calculating $g_{t_B}(c) \equiv g_{t_B}(g_{k_B}(m)) \equiv m \bmod n_B$.

5. Cryptanalysis

Since unlike to B a spy does not know the factorization of n_B, he cannot compute a decryption key t_B in the same way as B does. However, he might try to use other methods of decryption, especially to do partial decryption, that is to decrypt certain ciphertexts without knowing a decryption key t_B.

In the following we discuss several procedures of partial decryption. We show, that in some cases these attacks can be used also for factoring n. All discussed attacks are analogues to well-known attacks on the RSA-scheme (cf. Schnorr (1981), Simmons and Norris (1977), Berkowitz (1982), Herlestam (1978), Rivest (1978)). For a more algebraic discussion of superenciphering attacks on variants of the RSA-scheme see also W. Nöbauer (1985).

In the following we restrict ourselves to the cryptographically most important case where n is the product of two distinct odd prime numbers, that is $n = p_1 p_2$. We show that the Dickson-scheme is secure from the described attacks, if $p_i - 1$ $(i = 1,2)$ contains a large prime factor p_i', if $p_i + 1$ $(i = 1,2)$ contains a large prime factor p_i^*, and if as well the order of $k \bmod p_i'$ as the order of $k \bmod p_i^*$ $(i = 1,2)$ is large. These requirements are fulfilled, if e.g. for $i = 1,2$

$$(6) \qquad \begin{cases} p_i - 1 = a_i p_i', & a_i < 10^5, \quad p_i' > 10^{80} \\ p_i + 1 = b_i p_i^*, & b_i < 10^5, \quad p_i^* > 10^{80}, \end{cases}$$

$$(7) \qquad \begin{cases} \operatorname{ord} p_i' (k) > 10^{11} \\ \operatorname{ord} p_i^* (k) > 10^{11}. \end{cases}$$

5.1. Attacks by finding an s with $g_s(c) \equiv 2 \bmod n$

5.1.1. Partial decryption

Let $c \in Z/(n)$ be a given ciphertext. Suppose, the cryptanalyst succeeds in finding a natural number s with $g_s(c) \equiv 2 \bmod n$. Let $s = s_1 s_2$, where s_1 contains all those prime factors of s which divide k, and s_2 contains the remaining prime factors. The numbers s_1 and s_2 can be computed without the knowledge of the prime factorization of s, by using the following

Algorithm 2:

> Input $\quad k, s$.
> Initialize $\quad s_1 = 1; \; s_2 = s$.
> While $\quad (s_2, k) > 1$ do $s_1 = s_1 (s_2, k); \; s_2 = \frac{s_2}{(s_2, k)}$.
> End.

Let $u_i \in GF(p_i^2)$, $i = 1,2$, be solutions of $u + \frac{1}{u} = c$. (Such solutions always exist.) From $g_s(c) \equiv 2 \bmod n$ we obtain $g_s(c) \equiv 2 \bmod p_i$ for $i = 1,2$, and using (1) it follows, that in $GF(p_i^2)$ the equation $g_s(c) = g_s(u_i + \frac{1}{u_i}) = u_i^s + \frac{1}{u_i^s} = 2$ holds. This is equivalent with $u_i^s = 1$, hence with $u_i^{s_1 s_2} = 1$. Since $(k, p_i^2 - 1) = 1$, we have also $(s_1, p_i^2 - 1) = 1$.

Let o_i be the order of u_i in $GF(p_i^2)^*$, the multiplicative group of

$GF(p_i^2)$. As $o_i | p_i^2-1$, there holds

(8) $$(s_1, o_i) = 1.$$

From $u_i^{s_1 s_2} = 1$ we get $o_i | s_1 s_2$, hence $o_i | s_2$ by (8), that is $u_i^{s_2} = 1$.

By definition of s_2 we have $(k, s_2) = 1$. Thus there exists a natural number $\bar{k}$ such that $k\bar{k} \equiv 1 \bmod s_2$. Suppose that $k\bar{k} = s_2 r + 1$.

If $m \equiv g_k^{-1}(c) \equiv g_t(c) \bmod n$ is the plaintext corresponding to c, then the equation $m = g_t(c) = g_t(u_i + \frac{1}{u_i}) = u_i^t + \frac{1}{u_i^t}$ holds in $GF(p_i^2)$ for $i = 1,2$. Therefore we have

$$g_{\bar{k}}(c) = g_{\bar{k}}(g_k(m)) = g_{\bar{k}k}(m) = g_{\bar{k}k}(u_i^t + \frac{1}{u_i^t}) = u_i^{t\bar{k}k} + \frac{1}{u_i^{t\bar{k}k}} =$$
$$= u_i^{ts_2 r+t} + \frac{1}{u_i^{ts_2 r+t}} = u_i^t + \frac{1}{u_i^t} = m$$

in $GF(p_i^2)$. By the Chinese remainder theorem we obtain $g_{\bar{k}}(c) = m \bmod n$.

If we assume that the search for an s such that $g_s(c) \equiv 2 \bmod n$ is done by trial and error, and more concretely by testing all s between 1 and 10^5, we can summarize our attack in the following

Algorithm 3 (Deciphering the cryptogram $c \in Z/(n)$):

Input n,k,c.
Initialize $s = 1$.
While $s < 10^5$ and $g_s(c) \not\equiv 2 \bmod n$ do $s = s+1$.
If $g_s(c) \not\equiv 2 \bmod n$ then stop; comment [algorithm unsuccessful].
Else
  Compute $s = s_1 s_2$, where s_1 contains all those prime factors of s which divide k, and s_2 consists of the remaining prime factors of s; comment [use algorithm 2].
  Compute a natural number $\bar{k}$ such that $k\bar{k} \equiv 1 \bmod s_2$.
  Decipher c by calculating $g_{\bar{k}}(c) \equiv m \bmod n$.
End.

Now we will show that the Dickson-scheme is secure from attack 5.1.1., if the key parameters satisfy (6). For $i = 1,2$, we consider the p_i equations

(9) $$z + \frac{1}{z} = q, \quad q \in GF(p_i),$$

or equivalently, the p_i quadratic equations $z^2 - qz + 1 = 0$. Let M_i be the set of elements of $GF(p_i^2)$, which are solutions of anyone of the equations (9). In W. Nöbauer (1968) it is shown that $M_i = K_i \cup L_i$,

where $K_i = \{u \in GF(p_i^2) : u^{p_i - 1} = 1\}$ and $L_i = \{u \in GF(p_i^2) : u^{p_i+1} = 1\}$.
Obviously, K_i and L_i are subgroups of $GF(p_i^2)^*$. If w is a generator of $GF(p_i^2)^*$, then $K_i = \{w^{(p_i+1)r_1} : r_1 = 0,1,\dots,p_i-2\}$ and $L_i = \{w^{(p_i-1)r_2} : r_2 = 0,1,\dots,p_i\}$.
For $q \neq \pm 2$, the equations (9) have exactly two solutions $u,v \in GF(p_i^2)$, which are either both elements of K_i or of L_i (cf. W. Nöbauer (1968)). For $q = \pm 2$, these equations have exactly one solution $u \in GF(p_i^2)$, namely $u = 1$ or $u = -1$ respectively.
The groups K_i and L_i are cyclic, and by (6) the orders of K_i and L_i are given by $|K_i| = p_i - 1 = a_i p_i'$ and by $|L_i| = p_i + 1 = b_i p_i^*$. If $u \in K_i$, then $\mathrm{ord}(u) \le 10^5$ holds if and only if $\mathrm{ord}(u) | a_i$. If $d | a_i$, then the number of elements $u \in K_i$ with $\mathrm{ord}_{K_i}(u) = d$ is given by $\varphi(d)$, and therefore the number of elements $u \in K_i$ with $\mathrm{ord}_{K_i}(u) \le 10^5$ is given by $\sum_{d|a_i} \varphi(d) = a_i$.
Thus we have proved

$$(10) \qquad |\{u \in K_i : \mathrm{ord}_{K_i}(u) \le 10^5\}| = a_i,$$

and similarly, we obtain

$$(11) \qquad |\{u \in L_i : \mathrm{ord}_{L_i}(u) \le 10^5\}| = b_i.$$

For a given ciphertext $c \in Z/(n)$, algorithm 3 is successful, if and only if there exists an s with $1 \le s \le 10^5$, such that $g_s(c) \equiv 2 \bmod n$, or equivalently, such that $g_s(c) \equiv 2 \bmod p_i$, $i = 1,2$. If $u \in K_i \cup L_i$ is a solution of $u + \frac{1}{u} = c$, then $g_s(c) \equiv 2 \bmod p_i$ holds if and only if $u^s + \frac{1}{u^s} = 2$, that is, if and only if $u^s = 1$. Using the Chinese remainder theorem and the equations (10) and (11), we obtain

$$\begin{aligned}
&|\{c \in Z/(n) : \exists\, s \text{ with } 1 \le s \le 10^5 \text{ such that } g_s(c) \equiv 2 \bmod n\}| \le \\
&\le \prod_{i=1}^{2} |\{c \in Z/(p_i) : \exists\, s \text{ with } 1 \le s \le 10^5 \text{ such that } g_s(c) \equiv 2 \bmod p_i\}| = \\
&= \prod_{i=1}^{2} [\tfrac{1}{2}|\{u \in K_i \setminus \{\pm 1\} : \mathrm{ord}_{K_i}(u) \le 10^5\}| + \tfrac{1}{2}|\{u \in L_i \setminus \{\pm 1\} : \mathrm{ord}_{L_i}(u) \le 10^5\}| + 2] = \\
&= \prod_{i=1}^{2} [\tfrac{1}{2}(a_i - 2) + \tfrac{1}{2}(b_i - 2) + 2] = \frac{1}{4} \prod_{i=1}^{2} (a_i + b_i) < 10^{10}.
\end{aligned}$$

Therefore, if (6) holds and if c is uniformly distributed on $Z/(n)$, then the probability that c can be decrypted by algorithm 3 is bounded by $10^{10}/10^{160} = 10^{-150}$.

5.1.2. Factoring of n

In certain cases, knowing an s such that $g_s(c) \equiv 2 \bmod n$ not only allows to decipher c, but also to factorize n.
For the following considerations we put $v_2(s) := \max\{e \in \mathbb{N} : 2^e | s\}$.
Suppose that a cryptanalyst succeeds in finding an even s such that $g_s(c) \equiv 2 \bmod n$. Let $u_i \in GF(p_i^2)$, $i = 1,2$, be a solution of $u_i + \frac{1}{u_i} = c$.

Then we have $u_i^s = 1$ for $i = 1,2$.

Let $j := \max\{r \in \{0,1,\dots,v_2(s)\} : u_i^{s/2^r} = 1,\ i = 1,2\} =$

$= \max\{r \in \{0,1,\dots,v_2(s)\} : g_{s/2^r}(c) \equiv 2 \bmod n\}$.

Since the equation $x^2 = 1$ has just the two solutions 1 and -1 in the cyclic group $GF(p_i^2)^*$, $i = 1,2$, one of the following four cases holds:

(i) $j = v_2(s)$

(ii) $j < v_2(s)$, $u_1^{s/2^{j+1}} = 1$, $u_2^{s/2^{j+1}} = -1$

(iii) $j < v_2(s)$, $u_1^{s/2^{j+1}} = -1$, $u_2^{s/2^{j+1}} = 1$

(iv) $j < v_2(s)$, $u_1^{s/2^{j+1}} = -1$, $u_2^{s/2^{j+1}} = -1$.

Case (i) is equivalent to $g_{s/2^{v_2(s)}}(c) \equiv 2 \bmod n$, case (iv) is equivalent to $g_{s/2^{j+1}}(c) \equiv -2 \bmod n$, and in these cases our procedure does not provide the factorization of n.
If case (ii) holds, then $g_{s/2^{j+1}}(c) \equiv 2 \bmod p_1$ and $g_{s/2^{j+1}}(c) \not\equiv 2 \bmod p_2$, and therefore $(g_{s/2^{j+1}}(c)-2, n) = p_1$. Similarly, in case (iii) there holds $(g_{s/2^{j+1}}(c)-2, n) = p_2$.

If we assume that searching for an s such that $g_s(c) \equiv 2 \bmod n$ is done by testing all even s between 1 and 10^5, we can summarize the attack in the following

<u>Algorithm 4:</u>

<u>Input</u> n,c.
<u>Initialize</u> s = 2.
<u>While</u> $s < 10^5$ <u>and</u> $g_s(c) \not\equiv 2 \bmod n$ <u>do</u> s = s+2.
<u>If</u> $g_s(c) \not\equiv 2 \bmod n$ <u>then</u> <u>goto</u> 10.
<u>Compute</u> $v_2(s)$.
<u>Compute</u> $j = \max\{r \in \{0,1,\dots,v_2(s)\} : g_{s/2^r}(c) \equiv 2 \bmod n\}$.

<u>If</u> $j = v_2(s)$ <u>goto</u> 10; <u>comment</u> [case (i)].

<u>Else</u> <u>if</u> $g_{s/2^{j+1}}(c) \equiv -2 \bmod n$ <u>goto</u> 10; <u>comment</u> [case (iv)].

<u>Else</u> <u>compute</u> $d = (g_{s/2^{j+1}}(c) - 2, n)$; <u>comment</u> [d is a non-trivial factor of n].

10 <u>Comment</u> [algorithm unsuccessful].

Since algorithm 4 is successful only with ciphertexts c which can be decrypted by algorithm 3, this algorithm does not represent a real threat to the Dickson-scheme: If condition (6) holds and if c is uniformly distributed on $Z/(n)$, then the probability that algorithm 4 provides a nontrivial factor of n is bounded by 10^{-150}.

5.2 Factoring by means of fixed points

Let s be an odd natural number, and let $c \not\equiv \pm 2 \bmod n$ be a fixed point of $g_s(x) \bmod n$. Clearly c is also a fixed point of $g_s(x) \bmod p_i$ for $i = 1,2$. Let $u_i \in GF(p_i^2)$ be a solution of $u_i + \frac{1}{u_i} = c$, $i = 1,2$. Then we have

$g_s(u_i + \frac{1}{u_i}) = u_i^s + \frac{1}{u_i^s} = u_i + \frac{1}{u_i}$, hence $(u_i^{s+1}-1)(u_i^{s-1}-1) = 0$, and therefore one of the equations $u_i^{s+1} = 1$ or $u_i^{s-1} = 1$ holds. Clearly, $u_i^{s+1} = 1$ is equivalent to $g_{s+1}(c) \equiv 2 \bmod p_i$, and $u_i^{s-1} = 1$ is equivalent to $g_{s-1}(c) \equiv 2 \bmod p_i$.

If $u_1^{s+1} = 1$ and $u_2^{s-1} = 1$, but not $u_2^{s+1} = 1$,
or $u_1^{s-1} = 1$ and $u_2^{s+1} = 1$, but not $u_2^{s-1} = 1$,
then $(g_{s+1}(c)-2,n) \in \{p_1,p_2\}$, and a factor of n is found. However, if $u_1^{s+1} = 1$ and $u_2^{s+1} = 1$ or $u_1^{s-1} = 1$ and $u_2^{s-1} = 1$, then we have found an even number $\bar{s}$ with $g_{\bar{s}}(c) \equiv 2 \bmod n$, and therefore attack 5.1.2. can be applied.

A special case of this attack is given, when $s = k$. Then c is a fixed point of the enciphering polynomial $g_k(x) \bmod n$.

As there is not known any systematic algorithm for the search for fixed points of $g_s(x) \bmod n$, only trial and error methods can be used. Therefore, the Dickson-scheme is secure from attack 5.2., if the number $\mathrm{fix}(n,s)$ of fixed points of $g_s(x) \bmod n$ is small. By the Chinese remainder theorem $\mathrm{fix}(n,s) = \prod_{i=1}^{2} \mathrm{fix}(p_i,s)$, and according to R. Nöbauer (1985) $\mathrm{fix}(p_i,s) = \frac{1}{2}[(s-1,p_i-1) + (s+1,p_i-1) + (s-1,p_i+1) + (s+1,p_i+1)] - 2$.

If the key parameters satisfy (6), then

$$\mathrm{fix}\,(p_i,s) = \tfrac{1}{2}\,[(s-1,a_i)(s-1,p_i') + (s+1,a_i)(s+1,p_i') + (s-1,b_i)(s-1,p_i^*) + (s+1,b_i)(s+1,p_i^*)] - 2.$$

If for $i = 1,2$

(12)[1] $$p_i' \nmid s-1,\ p_i' \nmid s+1,\ p_i^* \nmid s-1,\ p_i^* \nmid s+1,$$

we have $\mathrm{fix}(p_i,s) < 10^6$, and consequently $\mathrm{fix}(n,s) < 10^{12}$. In this case, the probability that a uniformly distributed $c \in Z(n)$ is a fixed point of $g_s(x) \bmod n$ is bounded by $10^{12}/10^{160} = 10^{-148}$, and the task of finding any fixed point is computationally infeasible.

Let us assume that the number s itself is chosen according to a uniform distribution on $M = \{1,2,\ldots,r\}$, where r is a large positive integer, e.g. $r = 10^{100}$. In the following we write $[x]$ for the greatest integer which is less or equal than the real number x. There are exactly $[\frac{r-1}{p_i'}]+1$ numbers $s \in M$ such that $p_i' | s-1$, namely the numbers $1,\ 1+p_i',\ 1+2p_i',\ \ldots,\ 1+[\frac{r-1}{p_i'}]p_i'$. Similarly, there are exactly $[\frac{r-1}{p_i^*}]+1$ numbers $s \in M$ such that $p_i^* | s-1$, there are exactly $[\frac{r+1}{p_i'}]$ numbers $s \in M$ sucht that $p_i' | s+1$, and there are exactly $[\frac{r+1}{p_i^*}]$ numbers $s \in M$ such that $p_i^* | s+1$. Since $p_i' > 10^{80}$, we obtain

$$[\frac{r-1}{p_i'}]+1 \le [\frac{r}{p_i'}]+1 \le [\frac{r}{10^{80}}]+1,$$

$$[\frac{r+1}{p_i'}] \le [\frac{r}{p_i'}]+1 \le [\frac{r}{10^{80}}]+1,$$

and the same inequalities hold also with p_i^* instead of p_i'. Therefore, an upper bound for the number of elements $s \in M$ with

$$p_i' | s-1 \quad \text{or} \quad p_i' | s+1 \quad \text{or} \quad p_i^* | s-1 \quad \text{or} \quad p_i^* | s+1$$

is given by $4([\frac{r}{10^{80}}]+1)$. Consequently, a lower bound for the probability that a uniformly distributed $s \in M$ satisfies (12), is given by

$$(r - \frac{4r}{10^{80}} - 4)/r = 1 - \frac{4}{10^{80}} - \frac{4}{r}\ .$$

Therefore, a uniformly distributed $s \in \{1,2,\ldots r\}$ satisfies (12) almost certainly.

---

1) We write $a \nmid b$ for "a does not divide b".

Altogether we obtain: If the key parameters satisfy (6), then the task of finding an $s \in \mathbb{N}$ and a $c \in Z/(n)$ such that c is a fixed point of $g_s(x) \bmod n$ is computationally infeasible.

5.3 Superenciphering

Let $c \in Z/(n)$ be a given ciphertext. We consider $g_k(c)$, $g_k^2(c)$, $g_k^3(c),\dots,$ where $g_k^r(x)$ denotes the function $g_k(x)$ iterated r times. Since $Z/(n)$ is finite, there are two exponents r and s such that $g_k^r(c) \equiv g_k^s(c) \bmod n$. This implies the existence of a positive integer t such that $g_k^t(c) \equiv c \bmod n$, or equivalently, $g_{k^t}(c) \equiv c \bmod n$. If m denotes the plaintext corresponding to c, it follows from $c \equiv g_k(m) \bmod n$ that $g_k^{t+1}(m) \equiv g_k(m) \bmod n$. Hence $g_k^t(m) \equiv m \bmod n$, and therefore $g_k^{t-1}(c) \equiv m \bmod n$, and the plaintext is obtained.

Sometimes superciphering also yields the factorization of n. Namely, from $g_k^t(c) \equiv c \bmod n$ follows $g_{k^t}(c) \equiv c \bmod n$. That means, c is a fixed point of $g_{k^t}(x) \bmod n$. Since k^t is odd, attack 5.2. can be applied. Superenciphering is only successful if there exists a small t - say $t \le 10^{10}$ - such that c is a fixed point of $g_{k^t}(x) \bmod n$. Thus the Dickson-scheme is secure from superenciphering, if for all $t \le 10^{10}$ the mapping $x \to g_{k^t}(x) \bmod n$ has only a small number of fixed points. Let us assume that the conditions (6) and (7) are satisfied. Then all t between 1 and 10^{10} fulfil $k^t \not\equiv \pm 1 \bmod p_i'$ and $k^t \not\equiv \pm 1 \bmod p_i^*$. Hence

$$\begin{aligned} fix(p_i,k^t) &= \tfrac{1}{2}\,[(k^t-1,a_ip_i') + (k^t+1,a_ip_i') + (k^t-1,b_ip_i^*) + \\ &\quad + (k^t+1,b_ip_i^*)] - 2 \le \\ &\le a_i+b_i-2 < 10^6, \end{aligned}$$

and therefore $fix(n,k^t) < 10^{12}$.

This yields

$$|\{c \in Z/(n) : \exists\, t \text{ with } 1 \le t \le 10^{10} \text{ and } g_{k^t}(c) \equiv c \bmod n\}| < \sum_{t=1}^{10^{10}} fix(n,k^t) < 10^{10} \cdot 10^{12} = 10^{22}.$$

Therefore, if the conditions (6) and (7) hold, then the fraction of ciphertexts $c \in Z/(n)$ which can be decrypted by superenciphering is bounded by $10^{22}/10^{160} = 10^{-138}$.

References

Berkowitz, S. (1982): Factoring via superencryption. Cryptologia 6, 229-237.

Herlestam, T. (1978): Critical remarks on some public-key cryptosystems. BIT 18, 493-496.

Lausch, H., Müller, W.B. and Nöbauer, W. (1973): Über die Struktur einer durch Dicksonpolynome dargestellten Permutationsgruppe des Restklassenringes modulo n. J. reine angew. Math. 261, 88-99.

Lidl, R. and Niederreiter, H. (1983): Finite Fields. Vol. 20 of the Encyclopedia of Mathematics and Its Applications. Addison-Wesley, Reading, Massachusetts.

Müller, W.B. and Nöbauer, W. (1981): Some remarks on public-key cryptosystems. Studia Sci. Math. Hungar. 16, 71-76.

Nöbauer, R. (1985): Über die Fixpunkte von durch Dicksonpolynome dargestellten Permutationen. Acta Arithmetica 45, 91-99.

Nöbauer, R. (1985/86): Key distribution systems based on polynomial functions and on Rédei-functions. To appear in Problems of Control and Information Theory.

Nöbauer, W. (1965): Über Permutationspolynome und Permutationsfunktionen für Primzahlpotenzen. Monatsh. Math. 69, 230-238.

Nöbauer, W. (1968): Über eine Klasse von Permutationspolynomen und die dadurch dargestellten Gruppen. J. reine angew. Math. 231, 215-219.

Nöbauer, W. (1985): On the length of cycles of polynomial permutations. To appear in Contributions to General Algebra 3, Verlag B.G. Teubner, Stuttgart.

Rivest, R. L. (1978): Remarks on a proposed cryptanalytic attack on the M.I.T. public-key cryptosystem. Cryptologia 2, 62-65.

Schnorr, C.P. (1981): Zur Analyse des RSA-Schemas. Preprint. Fachbereich Mathematik, Universität Frankfurt.

Simmons, G.J. and Norris, N.J. (1977): Preliminary comments on the M.I.T. public-key cryptosystem. Cryptologia 1, 406-414.