Cryptanalysis of an Algebraic Privacy Homomorphism

David Wagner*
University of California, Berkeley
daw@cs.berkeley.edu

Corrected version—September 24, 2003

Abstract. We use linear algebra to show that an algebraic privacy homomorphism proposed by Domingo-Ferrer is insecure for some parameter settings.

Keywords: Cryptanalysis, number-theoretic cryptosystems

1 The proposed scheme

Let Q, R denote two rings, with addition denoted by $+$ and multiplication by $\times$ in both cases. Let $E : \mathcal{K} \times Q \to R$ be a symmetric-key encryption scheme, with corresponding decryption $D : \mathcal{K} \times R \to Q$. We call E additively-homomorphic if $D_k(E_k(a) + E_k(b)) = a + b$ for all $a, b \in Q$ and all $k \in \mathcal{K}$. Similarly, we will call E multiplicatively-homomorphic if $D_k(E_k(a) \times E_k(b)) = a \times b$. Let $\mathbb{Z}/m\mathbb{Z}$ denote the ring of integers modulo m.

At *ISC 2002*, J. Domingo-Ferrer proposed an encryption scheme that is both additively- and multiplicatively-homomorphic [3]. Let us recall Domingo-Ferrer's scheme briefly here. First, we choose parameters:

Public values: a large integer m with many small divisors, and a small integer $d > 2$.
Private key: a small divisor m' of m, and a unit $r \in (\mathbb{Z}/m\mathbb{Z})^*$.

We form the ring $R \stackrel{\text{def}}{=} (\mathbb{Z}/m\mathbb{Z})[X]$ of polynomials in the indeterminate X with coefficients from $\mathbb{Z}/m\mathbb{Z}$. Ciphertexts are elements of R. The plaintext space is $Q \stackrel{\text{def}}{=} \mathbb{Z}/m'\mathbb{Z}$, hence the signature of Domingo-Ferrer's encryption algorithm is $E : \mathcal{K} \times \mathbb{Z}/m'\mathbb{Z} \to (\mathbb{Z}/m\mathbb{Z})[X]$ and $D : \mathcal{K} \times (\mathbb{Z}/m\mathbb{Z})[X] \to \mathbb{Z}/m'\mathbb{Z}$. The scheme is defined as follows:

Encryption: The encryption algorithm E is randomized. First, we pick uniformly at random a polynomial $p(X) \in (\mathbb{Z}/m\mathbb{Z})[X]$ of degree $\leq d$ satisfying $p(0) = 0$ and $p(1) \bmod m' = a$. Then, we calculate the polynomial $q(X) \stackrel{\text{def}}{=} p(rX) \in (\mathbb{Z}/m\mathbb{Z})[X]$, and this is our ciphertext, i.e., $E_{m',r}(a) \stackrel{\text{def}}{=} q(X)$. The ciphertext $q(X)$ is represented by its coefficients, so r is never revealed explicitly.

* This research was supported in part by NSF CAREER CCR-0093337.

Decryption: D is defined by $D_{r,m'}(q(X)) \stackrel{\text{def}}{=} q(r^{-1}) \bmod m'$, for ciphertexts $q(X) \in (\mathbb{Z}/m\mathbb{Z})[X]$.

Note that Domingo-Ferrer's scheme is both additively- and multiplicatively-homomorphic. However, the size of ciphertexts does grow each time we multiply ciphertexts.

For ease of notation, in this paper we define $r' \stackrel{\text{def}}{=} r \bmod m'$.

Corrections. An earlier version of this paper was published at *ISC'03* [5]. That version had serious flaws [2, 6]; this revision fixes them.

Provably secure? Domingo-Ferrer claims that, if we fix the number n of known plaintexts available to the cryptanalyst, then one can choose public parameters large enough so that the cryptanalyst is foiled [3, Corollary 13]. However, what his paper actually proves is that, under these conditions, there are many possible candidates for the private key that remain consistent with the known plaintexts [3, Theorem 7]. It follows that the cryptanalyst cannot recover the entire private key with certainty under these conditions.

We first note that the size of the parameters in Domingo-Ferrer's scheme grows with n. It is traditional in the cryptographic community to seek a scheme that takes only polynomial work for legitimate parties to evaluate, while requiring the cryptanalyst to spend super-polynomial work to break the scheme. To avoid confusion, we should emphasize that Domingo-Ferrer's scheme does not attain this goal, nor does it attempt to; the envisioned applications are chosen so that the number of known texts available to the adversary can be bounded in advance.

However, there is a bigger issue lurking in the wings. The paper's provable security result is unsatisfying, in that it does not rule out all possible attacks. Domingo-Ferrer's security theorem leaves open the possibility that, though the cryptanalyst cannot recover the entire private key, maybe the cryptanalyst can recover enough information to decrypt any other ciphertext of interest.

Indeed, in what follows we show that something like this does happen: in Domingo-Ferrer's scheme, (m', r') is all you need to know to decrypt, and we give an attack that recovers (m', r') efficiently, even though the attacker never learns the remainder of the private key (i.e., r). This shows that Domingo-Ferrer's Corollary 13 is false and demonstrates that the scheme is insecure.

By way of analogy, imagine an encryption scheme that completely ignores its key and simply acts as the identity transformation. So long as the key is long enough, it would be straightforward to show that no attack can recover the key with certainty. Nonetheless, such an encryption scheme is obviously insecure. Domingo-Ferrer's scheme behaves similarly; there is a portion of the key that is completely ignored by both encryption and decryption, so it is not surprising that there is no way to recover the entire key—but this does not promise anything about the security of Domingo-Ferrer's scheme. This illustrates the importance of using the right definitions when following the provable security paradigm.

We should clarify that our attack does not work for all parameter settings. However, it does work for many choices of parameters, and in particular, it breaks all the specific instantiations suggested in Domingo-Ferrer's original paper. Along the way, we show that the proof of security for Domingo-Ferrer's scheme is flawed. Consequently, Domingo-Ferrer's original proposal could reasonably be regarded as insecure.

2 Cryptanalysis

In the remainder of the paper, we describe a simple attack on Domingo-Ferrer's scheme.

Equivalent keys. The secret key consists of the pair (m', r), but it is easy to see that the decryption procedure needs only m' and r'. In other words, the Domingo-Ferrer scheme has many equivalent secret keys[1]: (m', r) is equivalent to (m', r^*) whenever $r \equiv r^* \pmod{m'}$. Put yet another way, the part of the key represented by the value $\frac{r-r'}{m'}$ is completely ignored by both the encryption and decryption procedures. Thus, it suffices to recover $r \bmod m'$, i.e., to recover r'.

In our attack, we recover the key (up to equivalences) one element at a time: first, we show how to find m'; then, we show how to recover r'.

Preliminaries. When analyzing the Domingo-Ferrer scheme, it will be useful to recall the properties of the resultant. Given two polynomials $f(X), g(X) \in R[X]$, their resultant, denoted $\mathrm{Res}(f, g)$, is an element of R that can be efficiently computed from the coefficients of f and g. Also, when f, g have a common root $f(z) = g(z) = 0$ with $z \in R$, then $\mathrm{Res}(f, g) = 0$. Finally, when these polynomials have integer coefficients, the resultant respects modular reduction: $\mathrm{Res}(f(X) \bmod n, g(X) \bmod n) = \mathrm{Res}(f(X), g(X)) \bmod n$.

Recovering m'. Assume we have a collection of 400 known plaintexts and their encryptions, $(a_i, q_i(X))$ where $q_i(X) = E_{m',r}(a_i)$. Define the polynomials $f_1(X)$, $\ldots$, $f_{400}(X) \in \mathbb{Z}/m\mathbb{Z}[X]$ by $f_i(X) \stackrel{\text{def}}{=} q_i(X) - a_i$. Note that we have $f_i(r^{-1}) \equiv 0 \pmod{m'}$, hence if we reduce these polynomials modulo m', they will all share a common root r^{-1}.

[1] Incidentally, this shows that Domingo-Ferrer's Theorem 11 is false. We can easily find two private keys (m', r_1) and (m', r_2) that both decrypt a random ciphertext the same way with certainty; simply take $r_2 = r_1 + m'$. The flaw in the proof of Theorem 11 is in the second of the three cases considered, where Lemma 6 is misapplied. Lemma 6 is applied to the quantity $p(r_1^{-1}) - p(r_2^{-1}) \in \mathbb{Z}/m\mathbb{Z}$, where $p(X)$ is a randomly chosen ciphertext (recall that r_1, r_2 are held fixed throughout). Lemma 6 only applies to integers chosen uniformly at random from $\{0, 1, \ldots, m-1\}$. However, the quantity $p(r_1^{-1}) - p(r_2^{-1})$ does not necessarily behave like a uniformly random integer, so Lemma 6 does not apply here. In addition, the proof implicitly assumes that the divisors of this quantity are distributed uniformly on $\{0, 1, \ldots, m-1\}$, but this assumption is not justified.

Let us consider the quantity $\mathrm{Res}(f_1, f_2)$. This is an element of $\mathbb{Z}/m\mathbb{Z}$. However, we also know that $f_i(X) \bmod m'$ and $f_j(X) \bmod m'$ share a common root, hence $\mathrm{Res}(f_1, f_2) \equiv 0 \pmod{m'}$. This suggests that $\mathrm{Res}(f_1, f_2)$ should behave like a random multiple of m', i.e., it is likely to be uniformly distributed on the set $\{0, m', 2m', \ldots, m - m'\}$.

It is known that if we take the greatest common denominator of a pair of random, sufficiently-large integers, the greatest common denominator is 1 with probability about $6/\pi^2$. Hence, the greatest common denominator of a pair of random multiples of m' will be m' with probability about $6/\pi^2$.

This suggests a simple method for recovering m'. Let $x_i \stackrel{\text{def}}{=} \mathrm{Res}(f_{2i-1}, f_{2i})$. From 400 known plaintexts, we can compute 200 known x values. We can use $\gcd(x_1, x_2)$ as an initial guess at m'; this guess will be correct with probability $6/\pi^2$. If we repeat this procedure 100 times with 100 different pairs of x values, then with probability $1 - (1 - 6/\pi^2) \approx 1 - 2^{-135}$, the correct value of m' will appear somewhere in these 100 results, so we could imagine repeating the rest of the attack with each of these 100 candidates for m'. Better still is to note that the most common result among these 100 is almost certain to be m', so taking a majority vote should suffice to recover m'. Even better is to compute $\gcd(x_1, x_2, \ldots, x_{200})$.

This gives an efficient algorithm for recovering part of the secret key given only a modest supply of known plaintext. All that remains is to learn r'.

Recovering r'. If m' is small, finding r' is easy: we can simply enumerate all possibilities for r', and test which ones correctly decrypt all 400 of our known plaintexts. 3 of the 6 parameter sets suggested by Domingo-Ferrer use a value of m' that is 5 decimal digits long, so brute-force enumeration would break these schemes with complexity about 2^{17}. However, it is easy to choose larger values of m' that would make brute-force search infeasible, and hence we describe next a more clever attack that drastically reduces the complexity of finding m', namely, from exponential time to polynomial time.

Assume that we have a collection of d known plaintexts and their encryptions, $(a_i, q_i(X))$ where $q_i(X) = \sum_{j=0}^{d} q_{i,j} X^j = E_{m',r}(a_i)$. Note that if we view the terms $X, X^2, \ldots, X^d$ as d formal unknowns, then the identity $q_i(r^{-1}) \equiv a_i \pmod{m'}$ gives d linear equations in d unknowns over $\mathbb{Z}/m'\mathbb{Z}$.

In other words, let $Y_1, \ldots, Y_d$ denote d formal indeterminates. To aid the intuition, we should think informally of associating the indeterminate Y_j with the term X^j. Consider the d linear equations

$$\sum_{j=1}^{d} q_{i,j} Y_j \equiv a_i \pmod{m'}$$

over $\mathbb{Z}/m'\mathbb{Z}$ in the d unknowns $Y_1, \ldots, Y_d$. Note that the assignment $Y_j \stackrel{\text{def}}{=} (r')^{-j} \equiv r^{-j} \pmod{m'}$ satisfies all of these linear equations, since by assumption $q_i(r^{-1}) \equiv a_i \pmod{m'}$ and $q_{i,0} \equiv 0 \pmod{m'}$.

A standard fact[2] of linear algebra says that if we are given a system of d random linear equations over $\mathbb{Z}/m'\mathbb{Z}$ in d unknowns, then with non-negligible probability there is a unique solution to these equations, and this solution can be found in $O(d^3(\lg m')^2)$ time using Gaussian elimination. Hence, by the above comments, for a random choice of plaintexts a_i, we expect to be able to solve the system of equations for a unique solution $Y_1, \ldots, Y_d \in \mathbb{Z}/m'\mathbb{Z}$. Subsequently, $Y_1^{-1} \bmod m'$ will be a good candidate for $r \bmod m'$. This attack succeeds with constant probability, so after re-trying this procedure a constant number of times, we expect to learn the true value of $r \bmod m'$. Once $r \bmod m' = r'$ is known, we have everything we need to know to be able to decrypt any ciphertext we like. This attack is applicable so long as $n > d$.

Combining these two observations, we see that the Domingo-Ferrer proposal is broken when $n > d$. For instance, the suggested parameter sets in the original paper all had $d = 3$ and $n = 5$, 10, or 50, so we have $n > d$ in all 6 of the 6 parameter choices suggested in the original paper, and hence all 6 of those instantiations are all insecure.

Another way to recover r': polynomial root-finding. Suppose m' has known factorization. Then there is a totally different way to recover r', using algorithms for factoring polynomials over finite fields rather than linear algebra. The starting point is the following observation: each known plaintext gives an equation of the form $q(r^{-1}) \equiv 0 \pmod{m'}$ where the polynomial q and the modulus m' are known, but where the root r^{-1} is unknown. Since several polynomial-time algorithms for finding roots of polynomials are known, this suggesting using them to learn $r^{-1} \bmod m' = (r')^{-1}$.

The method is clearest in the case where m' is prime. In this case, we can take any one known plaintext, derive an equation $q(r^{-1}) \equiv 0 \pmod{m'}$ and find its roots (of which there are at most d, and which can be found in polynomial time using standard methods for univariate polynomial factorization in polynomial time). Each root of q is a candidate value of $(r')^{-1}$, and we may test it against other known plaintexts for validity. In this way, we recover r' in the case where m' is prime.

If m' is prime power, say $m' = p^e$, then we obtain the equation $q(r^{-1}) \equiv 0 \pmod{p^e}$. The roots of this equation may also be found efficiently [4]: first find all roots modulo p, and then use Hensel lifting to find the roots modulo p^2, then

[2] Let $f(n)$ denote the probability that a random, large matrix over $\mathbb{Z}/n\mathbb{Z}$ is invertible. It is well-known that $f(p)$ is non-negligible when p is prime; for instance, we have $f(2) \geq 0.288$, $f(3) \geq 0.560$, $f(5) \geq 0.760$, and so on [1]. Also, $f(p) = (1 - \frac{1}{p})(1 - \frac{1}{p^2}) \cdots (1 - \frac{1}{p^n}) \geq \exp\{-2/(p-1)\} \geq 1 - 2/(p-1)$, so random matrices modulo a large prime have an overwhelming probability of being invertible. Next, a matrix is invertible modulo p^e if and only if it is invertible modulo p, so $f(p^e) = f(p)$. (See also [1, Corollary 2.2].) Finally, by the Chinese remainder theorem, if $n = p_1^{e_1} \cdots p_k^{e_k}$, then $f(n) = f(p_1) \times \cdots \times f(p_k)$. In particular, $f(n) \geq e^{-2h(n)}$ where $h(n) = \sum_{p|n} 1/(p-1)$, so in most cases, there is a very good probability that a system of d random linear equations in d unknowns will be solvable uniquely.

modulo p^3, and so on. There are at most d roots, and each such candidate for $(r')^{-1}$ may then be tested against other known plaintexts as before.

In general, m' may be an arbitrary composite. If m' has known factorization $m' = p_1^{e_1} \cdots p_k^{e_k}$, we may find $r' \bmod p_i^{e_i}$ for each i using the methods described above, finally applying the Chinese remainder theorem to recover $r' \bmod m'$. Heuristically, it seems likely that this will succeed with high probability so long as we have a small number of known plaintexts, though we have no proof of this claim, and so this prediction should be viewed with some skepticism.

Summary. First, we showed an efficient way to recover m' whenever we have a small pool of known plaintexts. Then, we described several ways to recover r'. One possibility is exhaustive search; this works whenever m' is small (e.g., for 3 of the 6 parameter settings originally suggested). Another possibility is an attack based on linear algebra, which works with reasonable success probability whenever $n \geq d$ (e.g., for all 6 of 6 parameter settings). A third possibility is an attack based on polynomial root-finding, which applies m' can be factored and we have a few known plaintexts. In each case, these attacks are conjectured to work, but we have no formal proof of this conjecture.

These known attacks break all schemes proposed in the original paper. To repair the scheme, one might hope that other class of parameter choices might provide better security. Thus, we can ask: Is there any hope for Domingo-Ferrer's privacy homomorphism to remain secure in the case $n < d$? We do not know.

There is one case where no attack strategy is known to us: namely, where m' is chosen to be hard to factor and the number of texts encrypted is restricted so that $n < d$. One might hope for this to be secure. Unfortunately, there is some bad news: if we choose from this restricted class of parameters, then the performance of the scheme suffers dramatically. For instance, if we want to securely encrypt up to $n = 100$ messages, we might reasonably take m' to be a random 1024-bit RSA modulus, d and s to be slightly larger than n (say, $d = s = 105$), and m to be a random $1024s$-bit multiple of m'. However, then we find that each ciphertext is 11 million bits long, so the scheme is likely to be cumbersome in practice. For this reason, we do not see any obvious way to rescue Domingo-Ferrer's proposal.

3 Acknowledgements

I thank Dr. Josep Domingo-Ferrer and the anonymous reviewers for several useful comments on an earlier version of this paper. I thank Rob Johnson for showing me the content of Footnote 2. I am very grateful to Koji Chida for pointing out serious flaws in the proceedings version of this paper.

References

1. R.P. Brent, B.D. McKay, "Determinants and Ranks of Random Matrices Over $\mathbb{Z}_m$," *Discrete Mathematics* 66, pp.35–49, 1987.
2. K. Chida, personal communication, August 7, 2003.

3. J. Domingo-Ferrer, "A Provably Secure Additive and Multiplicative Privacy Homomorphism," *ISC 2002*, LNCS 2433, pp. 471–483, 2002.
4. G. McGuire, "An Approach to Hensel's Lemma," *Irish Math Soc. Bulletin* 47, pp.15–21, 2001.
5. D. Wagner, "Cryptanalysis of an Algebraic Privacy Homomorphism," *ISC 2003*, LNCS 2851, 2003.
6. D. Wagner, "Erratum Concerning 'Cryptanalysis of an Algebraic Privacy Homomorphism'," September 24, 2003.