Security Analysis of MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$

Christian Tobias

Justus Liebig University Giessen, Department of Mathematics
Arndtstrasse 2, 35392, Germany

Abstract. This paper cryptanalyses the MOR cryptosystem [6] when the group $GL(2, R) \times_\theta \mathbb{Z}_n$ proposed in [7] is used.
We show generic attacks on the system that work with every ring R. For a concrete choice of R even stronger attacks may be possible.

Key words: MOR cryptosystem, cryptanalysis, conjugacy problem

1 Introduction

In 2001 Paeng, Ha, Kim, Chee and Park proposed a new cryptosystem based on the difficulty of the discrete logarithm problem in the inner automorphism group $Inn(G)$ of a non-abelian group G [6]. Later this system was named MOR cryptosystem [7].
The used non-abelian group G has to be chosen very carefully not to undermine the security of the system. The first proposal for G was the semi-direct product group $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_p$ (see [6]). The authors themselves showed the interrelation between MOR using $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_p$ and MOR using $SL(2, \mathbb{Z}_p)$. Since the conjugacy and the special conjugacy problem can be efficiently solved in $SL(2, \mathbb{Z}_p)$, the security of MOR using $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_p$ could be reduced to the hardness of the discrete logarithm problem in $SL(2, \mathbb{Z}_p)$ (see [7]).
In 2003 a detailed analysis of MOR using $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_p$ [8] was published. The efficient modes of MOR using $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_p$ proved to be extremely vulnerable to the presented attacks. In some cases an attacker is able to gain information equivalent to the secret key.
In [7] Paeng, Kwon, Ha and Kim described how to construct a semi-direct product group $GL(2, R) \times_\theta \mathbb{Z}_n$ from a given ring isomorphism $\Phi : R \to R$ and proposed to use this group for the MOR cryptosystem. The purpose of this article is to evaluate the level of security provided by MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$. Our analysis focusses on the impact of the hardness of the computational Diffie-Hellman and the discrete logarithm problem in $< \Phi >$ on the security of MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$. We show that if the computational Diffie-Hellman problem can be solved efficiently in $< \Phi >$, then the efficient modes of MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$ are vulnerable to chosen-ciphertext attacks. Furthermore, if even the discrete logarithm problem can be solved efficiently in $< \Phi >$, then the secret key can be (partly) calculated from the public parameters.
The rest of this paper is organized as follows. In section 2 needed notations

Tobias C. (2004).
Security Analysis of MOR using GL(2, R) ×Îÿ ZZ n.
In *Proceedings of the 2nd International Workshop on Security in Information Systems*, pages 170-179
DOI: 10.5220/0002671201700179

and definitions are described and the MOR cryptosystem is introduced. Section 3 shows how to construct a semi-direct product group $GL(2,R) \times_\theta \mathbb{Z}_n$ given a ring isomorphism $\Phi : R \to R$ and how to apply this group to the MOR cryptosystem. We further demonstrate that the discrete logarithm problem in $Inn(GL(2,R) \times_\theta \mathbb{Z}_n)$ can be reduced to the discrete logarithm problem in $< \Phi >$. In section 4 we show that MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$ is vulnerable to chosen ciphertext attacks if the computational Diffie-Hellman problem in $< \Phi >$ can be solved efficiently. In the final section 5 the impact of the presented attacks on the security of MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$ is discussed and directions for future research are pointed out. The appendix briefly describes how to solve the special conjugacy problem (SCP) in $GL(2,R)$ by solving simultaneous instances of the conjugacy problem (CP) in $GL(2,R)$.

Related Work: The conjugacy problem is considered a hard problem in braid groups. There is no known polynomial time algorithm which solves the decisional or the computational conjugacy problem in braid groups. For a detailed discussion of cryptography on braid groups we refer to [1, 3, 5]. Other cryptosystems using the conjugation map on matrix groups have been published by Yamamura [9, 10]. The systems later were broken by Blackburn and Galbraith [2].

2 Framework and Definitions

Definition 1 (Semi-Direct Product Group). *Let G and H be groups and $\theta : H \to Aut(G)$ be a homomorphism. The set $G \times H = \{(g,h) \mid g \in G, h \in H\}$ together with the multiplication map*

$$(g_1, h_1)(g_2, h_2) = (g_1\theta(h_1)(g_2), h_1h_2)$$

is a group, called the semi-direct product $G \times_\theta H$ of G and H with respect to θ.

Definition 2 (The mapping Inn). *Let G be a group. Then the mapping*

$$\begin{aligned} Inn : G &\to Aut(G) \\ g &\mapsto Inn(g) \end{aligned}$$

is given by $Inn(g)(h) = ghg^{-1}$.

We call $Inn(g)$ an inner automorphism and $Inn(G) = \{Inn(g) \mid g \in G\}$ the inner automorphism group. If G is an abelian group then $Inn(g)$ is the identity map for all $g \in G$ and $Inn(G)$ is trivial. Let $\{\gamma_i\}$ be a set of generators of G. Since $Inn(g)$ is a homomorphism, $Inn(g)$ is totally specified for all $m \in G$ if the values $\{Inn(g)(\gamma_i)\}$ are given.

Definition 3 (center, centralizer). *Let G be a group. The center $Z(G)$ of G is defined as $Z(G) := \{g \in G \mid xg = gx \ \forall x \in G\}$.*
Let $g \in G$. The centralizer $Z(g)$ of g is defined as $Z(g) := \{h \in G \mid hg = gh\}$.
Note that $Z(G) = \bigcap_{g \in G} Z(g)$.

In the appendix the terms "center" and "centralizer" are also used for rings resp. ring elements. For a ring R and ring elements $r \in R$ we define $Z(R) := \{r \in R \mid sr = rs \ \forall s \in R\}$ and $Z(r) := \{s \in R \mid rs = sr\}$.
In some cases it may not be clear from the context which structure is referred to, e.g. for $g \in GL(2,R) \subseteq M(2,R)$ the cenralizer $Z(g)$ in the ring $M(2,R)$ may be different from the centralizer $Z(g)$ in the multiplicative group $GL(2,R)$. In this case the corresponding structure is added as an index, e.g. $Z_{M(2,R)}(g) = \{h \in M(2,R) \mid gh = hg\}$ and $Z_{GL(2,R)}(g) = \{h \in GL(2,R) \mid gh = hg\}$.

Definition 4 (Conjugacy Problem). *Let G be a group. For arbitrary $x, y \in G$ the conjugacy problem (CP) is to find $w \in G$ such that $wxw^{-1} = y$.*

Let $w \in G$ be a solution of the instance (x, y) of the CP, i.e. $wxw^{-1} = y$. Then $w \cdot Z(x)$ is the solution set for instance (x, y).

Definition 5 (Special Conjugacy Problem). *For a given $\varphi \in Inn(G)$ the special conjugacy problem is to find an element $g \in G$ satisfying $Inn(g) = \varphi$.*

The solution set for instance $Inn(g)$ of the special conjugacy problem is $g \cdot Z(G)$. In $GL(2, \mathbb{Z}_p)$ the conjugacy problem is easy. To solve the special conjugacy problem in $GL(2, \mathbb{Z}_p)$ two pairs $(A_1, Inn(A_1))$ and $(A_2, Inn(A_2))$ with $A_1 \notin Z(A_2)$ are needed (see [8] for details). A similar result holds for the group $GL(2,R)$ of invertible matrices over a commutative ring with identity R (see appendix A).

The MOR cryptosystem: MOR is an asymmetric cryptosystem with a random value a as secret and the two mappings $Inn(g)$ and $Inn(g^a)$ (given as $\{Inn(g)(\gamma_i)\}$ and $\{Inn(g^a)(\gamma_i)\}$ for a generator set $\{\gamma_i\}$ of G) as corresponding public key. The encryption process works as follows:

1. Alice expresses the plaintext $m \in G$ as a product of the γ_i.
2. Alice chooses a random $b \in_R \mathbb{Z}_{\mathrm{ord}(Inn(g))}$ and computes $(Inn(g^a))^b$, i.e. $\{(Inn(g^a))^b(\gamma_i)\}$.
3. Alice computes $E = Inn(g^{ab})(m) = (Inn(g^a))^b(m)$.
4. Alice computes $\Phi = Inn(g)^b$, i.e. $\{Inn(g^b)(\gamma_i)\}$.
5. Alice sends the ciphertext $C = (E, \Phi)$ to Bob.

Decryption Process:

1. Bob expresses E as a product of the γ_i.
2. Bob computes Φ^{-a}, i.e. $\{\Phi^{-a}(\gamma_i)\}$.
3. Bob computes $m = \Phi^{-a}(E)$.

The MOR cryptosystem is very similar to the ElGamal cryptosystem [4]. The Diffie-Hellman key establishment protocol is used to fix a common inner automorphism $(Inn(g))^{ab}$. The ciphertext of a message $m \in G$ is the image of m under $Inn(g^{ab}) = (Inn(g))^{ab}$.
In [6] no formal proof of security is given for the MOR system. If the discrete logarithm problem is efficiently solvable in $< Inn(g) >$, then the secret key a can be calculated from $Inn(g), Inn(g^a)$ which are part of the public key. However, knowledge of the secret key is not necessary to attack the MOR cryptosystem for certain non-abelian groups G (see [8] for details).

3 MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$

Let R be a commutative ring with identity and $\Phi : R \to R$ be a (non-trivial) ring isomorphism. Then $GL(2,R) = \{ \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \in M(2,R) \mid a_1a_4 - a_2a_3 \text{ is invertible}\}$ is a (multiplicative) group. A group automorphism ϕ is induced by Φ:

$$\phi : GL(2,R) \to GL(2,R),$$

$$\begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \mapsto \begin{pmatrix} \Phi(a_1) & \Phi(a_2) \\ \Phi(a_3) & \Phi(a_4) \end{pmatrix}$$

By setting $\theta(1) = \phi$ we get a homomorphism $\theta : \mathbb{Z}_n \to Aut(GL(2,R))$, i.e.

$$\theta(k) = \phi^k : \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \to \begin{pmatrix} \Phi^k(a_1) & \Phi^k(a_2) \\ \Phi^k(a_3) & \Phi^k(a_4) \end{pmatrix}$$

We now examine MOR using the semi-direct product $GL(2,R) \times_\theta \mathbb{Z}_n$.

The conjugation map in $GL(2,R) \times_\theta \mathbb{Z}_n$:
Let $(x,y),(m_1,m_2) \in G \times_\theta H$. Then:

$$(x,y)(m_1,m_2)(x,y)^{-1} = (x\theta(y)(m_1)\theta(m_2)(x^{-1}), m_2)$$

Applied to the group $G = GL(2,R) \times_\theta \mathbb{Z}_n$ and homomorphism θ we get for $(x,y),(m_1,m_2) \in GL(2,R) \times_\theta \mathbb{Z}_n$:

$$(x,y)(m_1,m_2)(x,y)^{-1} = (x \cdot \phi^y(m_1) \cdot \phi^{m_2}(x^{-1}), m_2)$$

The choice of Φ:
Let $G = GL(2,R) \times_\theta \mathbb{Z}_n$ and Φ, ϕ and θ as defined above. Then

1. $\text{ord}(\Phi) = \text{ord}(\phi)$
2. $\theta(n) = Id_{GL(2,R)} \Leftrightarrow n \equiv 0 \pmod{\text{ord}(\Phi)}$
3. If $(x,y),(x,\hat{y}) \in G$, then $Inn((x,y)) = Inn((x,\hat{y})) \Leftrightarrow y \equiv \hat{y} \pmod{\text{ord}(\Phi)}$
4. The homomorphism θ is well-defined if and only if $\text{ord}(\Phi) \mid n$.

Let $(x,y) \in GL(2,R) \times_\theta \mathbb{Z}_n$ and $(x,y)^{ab} = (\hat{x}, aby \pmod n)$ for some $\hat{x} \in GL(2,R)$. Then a ciphertext of a message $(m_1,m_2) \in GL(2,R) \times_\theta \mathbb{Z}_n$ looks as follows:

$$Inn((x,y)^{ab})(m_1,m_2) = (\hat{x}\phi^{aby}(m_1)\phi^{m_2}(\hat{x}^{-1}), m_2)$$

The values $a,b,y \in \mathbb{Z}_n$ should have no common divisor with the order of homomorphism ϕ. Otherwise ϕ^{aby} is no generator of the cyclic group $< \phi >$. This reduces the number of possible ciphertexts for a plaintext message $(m_1,m_2) \in GL(2,R) \times_\theta \mathbb{Z}_n$. To avoid this problem, we suggest to choose n prime.

Extracting ϕ^y from $Inn(g)$:
We now show that given an inner automorphism $Inn(g)$ for some $g = (x, y) \in GL(2, R) \times_\theta \mathbb{Z}_n$ the group automorphism ϕ^y can be calculated efficiently.

Step 1: To calculate ϕ^y we make use of the fact that $\Phi^y(0) = 0$ and $\Phi^y(1) = 1$. For a unimodular matrix $m \in GL(2, R)$ (i.e. a matrix with entries only 0 and 1) it follows that $\phi^y(m) = m$ and we get

$$\begin{aligned} Inn(g)(m, 0) = (x, y)(m, 0)(x, y)^{-1} &= (x \cdot \phi^x(m) \cdot \phi^0(x^{-1}), 0) \\ &= (x \cdot m \cdot x^{-1}, 0) \end{aligned}$$

This leads to an instance m, xmx^{-1} of the conjugacy problem in $GL(2, R)$. By solving the two instances $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $x \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x^{-1}$ and $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$, $x \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} x^{-1}$ of the conjugacy problem in $GL(2, R)$ simultaneously the special conjugacy problem can be solved and an element $\hat{x} \in GL(2, R)$ with $Inn(x) = Inn(\hat{x})$ can be calculated (see appendix A).

Step 2: For arbitrary $m \in GL(2, R)$ we get

$$Inn(g)(m, 0) = (x, y)(m, 0)(x, y)^{-1} = (x \cdot \phi^y(m) \cdot x^{-1}, 0)$$

Since $Inn(x) = Inn(\hat{x})$ we know that $\hat{x}^{-1} \cdot x \in Z(GL(2, R))$. The image of martix m under ϕ^y can be calculated as follows:

$$Inn(\hat{x}^{-1})(x \cdot \phi^y(m) \cdot x^{-1}) = (\hat{x}^{-1}x) \cdot \phi^y(m) \cdot (\hat{x}^{-1}x)^{-1} = \phi^y(m)$$

Using the same technique the homomorphism ϕ^{ay} can be calculated given $Inn(g^a)$. Since $Inn(g)$ and $Inn(g^a)$ are part of the public key, the two ring homomorphisms ϕ^y and ϕ^{ay} can be calculated efficiently. For the security of MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$ it is necessary that the discrete logarithm problem is hard in $< \phi >$. Otherwise $a \pmod{\text{ord}(\phi)}$ can be calculated which gives partial information of the secret key a.

4 Analysis of MOR using $GL(2, R) \times_\theta \mathbb{Z}_n$

The most time consuming operations in the encryption and decryption process of the MOR cryptosystem are the exponentiations in $< Inn(g) >$. The inner automorphisms are given by the images of the generators $\gamma_1, \ldots \gamma_n$ of the used group G. To calculate $Inn(g^2)(\gamma_i)$, two steps are needed. In the first step $Inn(g)(\gamma_i)$ has to be expressed as a product of the generators γ_i and in the second step the corresponding images $Inn(g)(\gamma_i)$ have to be multiplied. Since 2 (resp. 1) exponentiations in $< Inn(g) >$ have to be calculated during the encryption (resp. decryption) process, the MOR cryptosystem in its basic form is much too inefficient to be of practical interest.
Therefore a variant of MOR has been proposed [6] where the encryption exponent b is used for multiple encryptions. Since the resulting encryption scheme

is deterministic, the authors of [6] recommend to use a probabilistic padding scheme when fixing the encryption exponent.
We now show that MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$ with fixed encryption exponent (even when the probabilistic padding scheme is used) is vulnerable to chosen ciphertext attacks if the computational Diffie-Hellman Problem in $< \phi >$ can be solved (efficiently). From $Inn(g^a)$ (which is part of the public key) and $Inn(g^b)$ (which is part of the ciphertext) the homomorphisms ϕ^{ay} and ϕ^{by} can be computed. Solving the computational Diffie-Hellman problem yields ϕ^{aby}.

Let $c = (c_1, c_2) \in GL(2,R)$ be a given challenge ciphertext of MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$. In a chosen ciphertext attack the attacker is assumed to have access to a decryption oracle. He is allowed to send ciphertexts $\hat{c} \neq c$ to the oracle and gets the corresponding plaintext messages. A cryptosystem is secure against chosen ciphertext attacks if such an attacker is not able to compute the plaintext corresponding to c efficiently.
In our attack we make use of the fact that the encryption function $Inn(g^{ab})$ is an automorphism, i.e. every $d = (d_1, d_2) \in GL(2,R) \times_\theta \mathbb{Z}_n$ is a valid ciphertext of a (maybe unknown) message $m = (m_1, m_2) \in GL(2,R) \times_\theta \mathbb{Z}_n$.
Let $g = (x, y) \in GL(2,R) \times_\theta \mathbb{Z}_n$. Then $(x,y)^{ab} = (\hat{x}, aby \pmod n)$ for some $\hat{x} \in GL(2,R)$. Ciphertexts of MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$ are of the form

$$d = (d_1, d_2) = (\hat{x} \cdot \phi^{aby}(m_1) \cdot \phi^{m_2}(\hat{x}^{-1}), m_2)$$

The attack consists of two steps. In the first step an $\bar{x} \in GL(2,R)$ with $Inn(\hat{x}) = Inn(\bar{x})$ is computed. This element $\bar{x}$ is used in the second step to decipher the challenge ciphertext c.

Step 1: For every $d_1 \in GL(2,R)$ the value $(d_1, 0) \in GL(2,R) \times_\theta \mathbb{Z}_n$ is a valid ciphertext of the (unknown) message $(m_1, 0) \in GL(2,R) \times_\theta \mathbb{Z}_n$:

$$(d_1, 0) = (\hat{x} \cdot \phi^{aby}(m_1) \cdot \hat{x}^{-1}, 0)$$

Sending $(d_1, 0)$ to the decryption oracle, the attacker gets the corresponding plaintext message $(m_1, 0)$. Since we assumed that the attacker knows ϕ^{aby} he is able to compute $\phi^{aby}(m_1)$. The values $\phi^{aby}(m_1), d_1 = \hat{x} \cdot \phi^{aby}(m_1) \cdot \hat{x}^{-1}$ form an instance of the conjugacy problem in $GL(2,R)$. Repeating this process generates multiple simultaneous instances of the conjugacy problem in $GL(2,R)$ which can be used to solve the special conjugacy problem in $GL(2,R)$ and get a group element $\bar{x} \in GL(2,R)$ with $Inn(\hat{x}) = Inn(\bar{x})$ (see appendix A for details).

The oracle may not answer queries with zero as second component, because $GL(2,R) \times_\theta \{0\}$ is isomorphic to $GL(2,R)$ and the conjugacy problem is efficiently solvable in $GL(2,R)$. In this case the attacker sends queries $(d_1, i), (\hat{d_1}, i) \in GL(2,R) \times_\theta \mathbb{Z}_n$ with the same second component to the decryption oracle:

$$(d_1, i) = (\hat{x} \cdot \phi^{aby}(m_1) \cdot \phi^i(\hat{x}^{-1}), i)$$
$$(\hat{d_1}, i) = (\hat{x} \cdot \phi^{aby}(\hat{m_1}) \cdot \phi^i(\hat{x}^{-1}), i)$$

With the plaintext messages $(m_1, i), (\hat{m_1}, i) \in GL(2,R) \times_\theta \mathbb{Z}_n$ and homomorphism ϕ^{aby} the attacker can compute $\phi^{aby}(m_1) \cdot (\phi^{aby}(\hat{m_1}))^{-1} = \phi^{aby}(m_1 \cdot \hat{m_1}^{-1})$ and $d_1 \cdot (\hat{d_1})^{-1} = \hat{x} \cdot \phi^{aby}(m_1 \cdot \hat{m_1}^{-1}) \cdot \hat{x}^{-1}$ to get an instance of the CP in $GL(2,R)$.

Step 2: Let (p_1, p_2) be the plaintext message encrypted in the challenge ciphertext $c = (c_1, c_2)$. Since $\bar{x} = \hat{x} \cdot z$ for a $z \in Z(GL(2,R))$ we get:

$$\begin{aligned} \bar{x}^{-1} \cdot c_1 \cdot \phi^{c_2}(\bar{x}) &= \bar{x}^{-1} \cdot (\hat{x} \cdot \phi^{aby}(p_1) \cdot \phi^{c_2}(\hat{x})) \cdot \phi^{c_2}(\bar{x}) \\ &= \phi^{aby}(p_1) \cdot z^{-1} \cdot \phi^{c_2}(z) \end{aligned}$$

Only one oracle query is necessary to calculate $z^{-1} \cdot \phi^{c_2}(z)$. The attacker chooses a $c_3 \neq c_1 \in GL(2,R)$ and sends (c_3, c_2) to the oracle. If $\hat{m}$ is the answer of the oracle, the attacker gets $z^{-1} \cdot \phi^{c_2}(z)$ as follows:

$$\begin{aligned} c_3 \cdot (\phi^{c_2}(\bar{x}) \phi^{aby}(\hat{m}^{-1}) \bar{x}^{-1}) &= (\hat{x} \phi^{aby}(\hat{m}) \phi^{c_2}(\hat{x}^{-1})) \cdot (\phi^{c_2}(\bar{x}) \phi^{aby}(\hat{m}^{-1}) \bar{x}^{-1}) \\ &= \hat{x} \phi^{aby}(\hat{m}) \phi^{c_2}(\hat{x}^{-1}) \phi^{c_2}(\hat{x}z) \phi^{aby}(\hat{m}^{-1}) (\hat{x}z)^{-1} \\ &= z^{-1} \cdot \phi^{c_2}(z) \end{aligned}$$

Now the attacker can compute $\phi^{aby}(p_1)$.

Step 3: If the knowledge of ϕ^{aby} is not sufficient to compute p_1 from $\phi^{aby}(p_1)$, the decryption oracle is used to compute preimages under ϕ^{aby}. To obtain the preimage of $\phi^{aby}(p_1)$ the attacker sends

$$(d_1, 0) = (\bar{x} \cdot \phi^{aby}(p_1) \cdot \bar{x}^{-1}, 0) = (\hat{x} \cdot \phi^{aby}(p_1) \cdot \hat{x}^{-1}, 0)$$

as query to the decryption oracle. The oracle reply equals the wanted preimage. If the oracle does not answer queries with zero as second component the value $\bar{x} \cdot \phi^{aby}(p_1) \cdot \bar{x}^{-1}$ can be expressed as $\bar{x} \cdot \phi^{aby}(p_1) \cdot \bar{x}^{-1} = e_1 \cdot \hat{e_1}^{-1}$ for $e_1, \hat{e_1} \in GL(2,R)$ and (e_1, i) and $(\hat{e_1}, i)$ can be sent to the oracle. If a_1 and $\hat{a_1}$ are the oracle's answers, the desired preimage is $p_1 = a_1 \cdot \hat{a_1}^{-1}$ (see also step 1 for a similar argument).

Using a randomised padding scheme: In [6] the authors propose to use a probabilistic padding scheme when fixing the encryption exponent. The plaintext message $m \in R$ is embedded in $GL(2,R)$ by choosing a random matrix $M = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} \in GL(2,R)$ with $m_1 = m$. After that the encryption function $Inn(g^{ab})$ is applied to M.
In [8] it has been shown that MOR using $SL(2, \mathbb{Z}_p) \times_\theta \mathbb{Z}_n$ is insecure even if the randomised padding scheme is used: Two pairs consisting of plaintext and corresponding ciphertext are sufficient to calculate $Inn(g^{ab})$. The same techniques can be applied to step 1 of our attack to calculate an element $\bar{x} \in GL(2,R)$ with $Inn(\hat{x}) = Inn(\bar{x})$.

The first part of step 2 also works if the described padding scheme is used, i.e. $\phi^{aby}(p_1) \cdot z^{-1} \cdot \phi^{c_2}(z)$ can be calculated. The second part of step 2 has to be

changed slightly: On input (c_3, c_2) the decryption oracle outputs only the $(1,1)$-component of $\hat{m}$. The other entries of matrix $\hat{m}$ are not known to the attacker. Since $Z(GL(2,R)) = \{c \cdot Id \mid c \in R, c \text{ invertible}\}$, the value $z^{-1} \cdot \phi^{c_2}(z)$ is of the form $z^{-1} \cdot \phi^{c_2}(z) = \begin{pmatrix} r & 0 \\ 0 & r \end{pmatrix}$ for an invertible $r \in R$. In particular $z^{-1} \cdot \phi^{c_2}(z) \in Z(GL(2,R))$. For $\hat{m} = \begin{pmatrix} \hat{m_1} & \hat{m_2} \\ \hat{m_3} & \hat{m_4} \end{pmatrix}$ we get

$$\begin{aligned} \bar{x}^{-1} \cdot c_3 \cdot \phi^{aby}(\bar{x}) &= (z^{-1}\hat{x}^{-1}) \cdot (\hat{x}\phi^{aby}(\hat{m})\phi^{c_2}(\hat{x}^{-1})) \cdot (\phi^{c_2}(\hat{x}z)) \\ &= \phi^{aby}(\hat{m}) \cdot z^{-1} \cdot \phi^{c_2}(z) \\ &= \begin{pmatrix} r \cdot \hat{m_1} & r \cdot \hat{m_2} \\ r \cdot \hat{m_3} & r \cdot \hat{m_4} \end{pmatrix} \end{aligned}$$

The value $\hat{m_1}$ can be obtained by sending (c_3, c_2) to the decryption oracle. If r cannot be calculated given $\hat{m_1}$ and $r \cdot \hat{m_1}$ this process has to be repeated with a different value c_3.

Step 3 also works when the randomised padding scheme is used but has to be carried out for every single component, i.e. to compute the preimage of $\phi^{aby}(p_1) = \begin{pmatrix} \Phi^{aby}(p_{11}) & \Phi^{aby}(p_{12}) \\ \Phi^{aby}(p_{13}) & \Phi^{aby}(p_{14}) \end{pmatrix}$ step 3 is used to find preimages of $d_i \in GL(2,R)$, $1 \leq i \leq 4$, where the $(1,1)$-component of d_i equals $\Phi^{aby}(p_{1i})$.

5 Conclusion

We showed that MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$ with fixed encryption exponent is vulnerable to chosen ciphertext attacks if the computational Diffie-Hellman Problem is easy in $< \Phi >$. The presented attacks still work if the randomised padding scheme of [6] is used. They do not work if the encryption exponent b is randomly chosen for every plaintext to be encrypted. However, in this case two exponentiations in $< Inn(g) >$ have to be calculated during the encryption and one during the decryption process. The resulting cryptosystem is too inefficient to be of practical interest.

Our results show that the hardness of the discrete logarithm problem (DLP) in $< \Phi >$ is essential for the security of all modes of MOR (even when the encryption exponent b is chosen randomly and independently for every plaintext to be encrypted). The DLP in $< \Phi >$ is much easier than the DLP in $< Inn(g) >$ (which has to be solved to calculate the secret key given the public key). It may be more appropriate to use a variant of the ElGamal cryptosystem [4] using the cyclic group $< \Phi >$. The resulting cryptosystem would be provable secure and more efficient than MOR using $GL(2,R) \times_\theta \mathbb{Z}_n$.

All attacks are generic attacks, i.e. they work for every ring R and every homomorphism Φ. For certain choices of R and Φ there may be even stronger attacks. It is a task for future reserach to find a non-abelian group suitable for the use with the MOR cryptosystem.

References

1. I. Anshel, M. Anshel, D. Goldfeld, "An Algebraic Method for Public-Key Cryptography", Mathematical Research Letters, 6 (1999), pp. 287-291
2. S. Blackburn, S. Galbraith, "Cryptanalysis of two cryptosystems based on group action", Advances in Cryptology - Asiacrypt 1999, LNCS 1716, pp. 52-61
3. P. Dehornoy, "Braid-based cryptography", Preprint, University of Caen, 2003, http://matin.math.unicaen.fr/∼dehornoy/papers.html
4. T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, Volume 31, 1985, pp. 469-472
5. K. H. Koo, S. J. Lee, J. H. Cheon, J. W. Han, J. Kang, C. Park, "New Public-Key Cryptosystem Using Braid Groups", Advances in Cryptology - Crypto 2000, LNCS 1880, pp. 166-183
6. S.-H. Paeng, K.-C. Ha, J. H. Kim, S. Chee, C. Park, "New Public Key Cryptosystem Using Finite Non Abelian Groups", Advances in Cryptology - Crypto 2001, LNCS 2139, pp. 470-485
7. S.-H. Paeng, D. Kwon, K.-C. Ha, J. H. Kim, "Improved public key cryptosystem using finite non abelian groups", IACR EPrint-Server, Report 2001/066, http://eprint.iacr.org/2001/066
8. C. Tobias, "Security Analysis of the MOR Cryptosystem", 6th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2003, LNCS 2567, pp. 175-186
9. A. Yamamura, "Public key cryptosystems using the modular group", 1st International Public Key Cryptography Conference PKC 1998, LNCS 1431, pp. 203-216
10. A. Yamamura, "A functional cryptosystem using a group action", 4th Australian Information Security and Privacy Conference, ACISP 1999, LNCS 1587, pp. 314-325

A The Special Conjugacy Problem in $GL(2, R)$

Let $Inn(g) : GL(2, R) \to GL(2, R)$ be a public inner automorphism. We assume that $Inn(g)$ is given as a black box, i.e. an attacker is able to calculate images under $Inn(g)$ but does not know the used $g \in GL(2, R)$. This approach assures that our calculations are independent of the presentation of $Inn(g)$. We now show that the special conjugacy problem is efficiently solvable in $GL(2, R)$.

Let $B, C, X \in GL(2, R)$ and $B, XBX^{-1} = \hat{B} = \begin{pmatrix} \hat{b_1} & \hat{b_2} \\ \hat{b_3} & \hat{b_4} \end{pmatrix}$ and $C, XCX^{-1} = \hat{C} = \begin{pmatrix} \hat{c_1} & \hat{c_2} \\ \hat{c_3} & \hat{c_4} \end{pmatrix}$ be two simultaneous instances of the conjugacy problem in $GL(2, R)$.

Let $\hat{X} \in GL(2, R)$ be a solution of these two instances. Then $\hat{X} = Z \cdot X$ with $\begin{pmatrix} z_1 & z_2 \\ z_3 & z_4 \end{pmatrix} = Z \in Z(\hat{B}) \cap Z(\hat{C})$. By comparing the components of $Z \cdot \hat{B}$, $\hat{B} \cdot Z$ and $Z \cdot \hat{C}$, $\hat{C} \cdot Z$ we get:[1]

[1] Since $\hat{X}$ could also be expressed as $\hat{X} = X \cdot \hat{Z}$ for a $\hat{Z} \in Z(B) \cap Z(C)$, the following paragraph is also true if $\hat{b_i}$ and $\hat{c_i}$ are replaced by b_i and c_i. In particular $B \in Z(C) \Leftrightarrow \hat{B} \in Z(\hat{C})$.

- $z_2(\hat{c_3}\hat{b_2} - \hat{b_3}\hat{c_2}) = 0$ and $z_3(\hat{c_3}\hat{b_2} - \hat{b_3}\hat{c_2}) = 0$
- $z_2(\hat{c_2}(\hat{b_1} - \hat{b_4}) - \hat{b_2}(\hat{c_1} - \hat{c_4})) = 0$ and $z_3(\hat{c_2}(\hat{b_1} - \hat{b_4}) - \hat{b_2}(\hat{c_1} - \hat{c_4})) = 0$
- $z_2(\hat{c_3}(\hat{b_1} - \hat{b_4}) - \hat{b_3}(\hat{c_1} - \hat{c_4})) = 0$ and $z_3(\hat{c_3}(\hat{b_1} - \hat{b_4}) - \hat{b_3}(\hat{c_1} - \hat{c_4})) = 0$

If $\hat{c_3}\hat{b_2} = \hat{b_3}\hat{c_2}$, $\hat{c_2}(\hat{b_1} - \hat{b_4}) = \hat{b_2}(\hat{c_1} - \hat{c_4})$ and $\hat{c_3}(\hat{b_1} - \hat{b_4}) = \hat{b_3}(\hat{c_1} - \hat{c_4})$, then $\hat{B} \in Z(\hat{C})$. Therefore, were $\hat{B}, \hat{C} \in GL(2, R)$ chosen such that $\hat{B} \notin Z(\hat{C})$, one of the equations has to be false and z_2 and z_3 are zero divisors.

If $\hat{B}, \hat{C} \in GL(2, R)$ where chosen such that $\hat{c_3}\hat{b_2} - \hat{b_3}\hat{c_2}$, $\hat{c_2}(\hat{b_1} - \hat{b_4}) - \hat{b_2}(\hat{c_1} - \hat{c_4})$ or $\hat{c_3}(\hat{b_1} - \hat{b_4}) - \hat{b_3}(\hat{c_1} - \hat{c_4})$ is no zero divisors it further follows that $z_2 = z_3 = 0$. If one of the ring elements $\hat{b_2}$, $\hat{b_3}$, $\hat{c_2}$ or $\hat{c_3}$ is no zero divisor, then $Z = \begin{pmatrix} z_1 & 0 \\ 0 & z_1 \end{pmatrix}$ for a $z_1 \in R$. Since $Z \cdot M = M \cdot Z$ for all $M \in M(2, R)$, we get that $Inn(X) = Inn(\hat{X})$, i.e. $\hat{X} \in GL(2, R)$ is a solution of the instance $Inn(X)$ of the special conjugacy problem in $GL(2, R)$.

We now show that a simultaneous solution of these two instances can be calculated efficiently. The equations $XBX^{-1} = \hat{B}$ and $XCX^{-1} = \hat{C}$ are equivalent to $XB = \hat{B}X$ and $XC = \hat{C}X$. If $B \notin Z(C)$ this yields to a system of three linear equations. In the presented attack in section 4 the elements $\hat{B}, \hat{C} \in GL(2, R)$ can be chosen freely. If $\hat{b_3}$ is invertible, the obtained system of linear equations is equivalent to:

$$\begin{aligned} x_1 + \quad & \tfrac{\hat{b_4} - b_1}{\hat{b_3}} \cdot x_3 \quad - \quad \tfrac{b_3}{\hat{b_3}} \cdot x_4 \quad = 0 \\ x_2 - \quad & \tfrac{b_2}{\hat{b_3}} \cdot x_3 \quad + \quad \tfrac{\hat{b_4} - b_4}{\hat{b_3}} \cdot x_4 \quad = 0 \\ & (\hat{c_4} - c_1 - \hat{c_3} \cdot \tfrac{\hat{b_4} - b_1}{\hat{b_3}}) \cdot x_3 - (c_3 - \hat{c_3} \cdot \tfrac{b_3}{\hat{b_3}}) \cdot x_4 = 0 \end{aligned}$$

For arbitrary $r \in R$ this system is solved by $x_1 = k_1 \cdot r$, $x_2 = k_2 \cdot r$, $x_3 = k_3 \cdot r$ and $x_4 = k_4 \cdot r$ where $k_4 = \hat{c_4} - c_1 - \hat{c_3} \cdot \frac{\hat{b_4} - b_1}{\hat{b_3}}$, $k_3 = (c_3 - \hat{c_3}\frac{b_3}{\hat{b_3}}) \cdot k$, $k_2 = \frac{b_2}{\hat{b_3}} \cdot k_3 - \frac{\hat{b_4} - b_4}{\hat{b_3}} \cdot k_4$ and $k_1 = \frac{b_3}{\hat{b_3}} \cdot k_4 - \frac{\hat{b_4} - b_1}{\hat{b_3}} \cdot k_3$.

If either $\hat{b_3}c_3 - \hat{c_3}b_3$ or $\hat{b_3}(\hat{c_4} - c_1) - \hat{c_3}(\hat{b_4} - b_1)$ is no zero divisor, $\begin{pmatrix} k_1 & k_2 \\ k_3 & k_4 \end{pmatrix} \in GL(2, R)$ and $\begin{pmatrix} rk_1 & rk_2 \\ rk_3 & rk_4 \end{pmatrix} \neq \begin{pmatrix} \hat{r}k_1 & \hat{r}k_2 \\ \hat{r}k_3 & \hat{r}k_4 \end{pmatrix}$ for $r, \hat{r} \in R$ with $r \neq \hat{r}$, i.e. we get $\mid R \mid$ distinct solutions. In this case we further know that $\begin{pmatrix} rk_1 & rk_2 \\ rk_3 & rk_4 \end{pmatrix} \in GL(2, R)$ if and only if $r \in R$ is no zero divisor.

Since $X \in GL(2, R)$, the equation $XB = \hat{B}X$ is equivalent to $\hat{B} = XBX^{-1}$. For an element $\hat{X} \in M(2, R)$ with $\hat{X}B = \hat{B}\hat{X}$ we get that $(X^{-1}\hat{X})B = B(X^{-1}\hat{X})$ holds, i.e. $\hat{X} = X \cdot Z$ with $Z \in Z_{M(2,R)}(B)$.

Thus, the simultaneous solutions (in $M(2, R)$) of the equations $XB = \hat{B}X$ and $XC = \hat{C}X$ are of the form $Z \cdot X$ where $Z \in Z_{M(2,R)}(\hat{B}) \cap Z_{M(2,R)}(\hat{C})$. If $\hat{B}, \hat{C} \in GL(2, R)$ were chosen such that $Z_{M(2,R)}(\hat{B}) \cap Z_{M(2,R)}(\hat{C}) = Z(M(2, R))$, there are $\mid Z(M(2, R)) \mid = \mid R \mid$ many solutions, i.e. all solutions are given by $x_1 = k_1 \cdot r$, $x_2 = k_2 \cdot r$, $x_3 = k_3 \cdot r$ and $x_4 = k_4 \cdot r$ with $r \in R$.