Cryptanalysis based on the theory of symmetric group representations

Romana Linkeová, Pavel Příhoda

***Abstract*—The key exchange Diffie-Hellman protocol originally works over the group $\mathbb{Z}_p^*$ where p is at least a 300-digit number. Even though this implementation is simple and secure, it makes the protocol unsuitable for devices with limited computational power. This fact led to a research of other algebraic structures which could be used as a platform for this protocol in order to decrease the computational and storage costs. Such attempt can be found in the work of D. Kahrobaei et al. posted in 2013. D. Kahrobaei et al. proposed a structure of small matrices over a group ring as a platform and claimed that this modification will not affect the security of the Diffie-Hellman protocol. We will attack this modification and prove that it is not secure with the help of the theory of symmetric group representations.**

***Index Terms*—Diffie-Hellman protocol, public key cryptography, symmetric group representations.**

I. Introduction

ONE of the requirements of symmetric cryptography is that two communicating parties are able to establish a secret shared key over a public channel without anyone else being able to retrieve their shared key from the communication as well. One of the cryptographic tools that solves this problem is the Diffie-Hellman protocol which was introduced by Witfield Diffie and Martin Hellman in 1976 in [2].

One of the drawbacks of this protocol is that it does not ensure the authentication of both parties. This fact makes the protocol vulnerable against the man-in-the-middle attack.

Another drawback is that using $\mathbb{Z}_p^*$ (the multiplicative group of integers modulo prime p, where p is suggested to have at least 300 digits), makes the protocol being unsuitable for devices with limited computational power. In order to decrease computational costs, we can exchange $\mathbb{Z}_p^*$ for another algebraic structure. One of the approaches that is trying to do so can be found in [6], where D. Kahrobaei et al. proposed a semigroup of $n \times n$ matrices over the group ring $\mathbb{F}_q[\boldsymbol{S}_m]$ as a platform. They proposed semigroups $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ and $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$ specifically.

Romana Linkeová is with the Department of Algebra, Faculty of Mathematics and Physics, Charles University in Prague, Sokolovská 83, 186 75 Prague, Czech Republic e-mail: linkeovaromana@gmail.com.

Pavel Příhoda is with the Department of Algebra, Faculty of Mathematics and Physics, Charles University in Prague, Sokolovská 83, 186 75 Prague, Czech Republic, e-mail: prihoda@karlin.mff.cuni.cz.

Manuscript received September 29, 2015; revised February 17, 2016.

The main advantage of this algebraic structure is that one can precompute a multiplicative table for elements from $\boldsymbol{S}_5$, which makes the computations in the semigroup very time-efficient. Another advantage is that this modification of the original protocol will not, according to D. Kahrobaei et al., decrease its security. In this paper, we will show that such modification will make the protocol insecure and it will be possible to retrieve the secret shared key within few hours using a common computer.

The security of the Diffie-Hellman key exchange protocol is based on the absence of an algorithm capable of solving the discrete logarithm problem in polynomial time. Nowadays, we are aware of multiple algorithms that are solving the discrete logarithm problem in non-polynomial time, such as baby-step giant-step, Pohlig-Hellman and Pollard's Rho (for more details see [5]). The authors of [6] claimed that those algorithms (together with the Shor's quantum algorithm) will not work for their modified protocol. In this paper, we will concentrate on the baby-step giant-step algorithm and show that it will be more effective than D. Kahrobaei et al. claimed.

Firstly, we will describe the necessary algebraic theory. Secondly, we will focus on the description of the original and the modified Diffie-Hellman protocol. Then, we will present the attack itself. After that, we will show that the baby-step giant-step algorithm will work on the modified protocol. The next section is focused on implementation of our attack. Lastly, we will present a list of papers that also proposed an attack on the modified protocol.

II. Definitions and notations

Definition 1 (Group ring)**.** *Let $G = (G, *, ^{-1}, e)$ be a finite group and let $R = (R, +, \cdot, -, 0_R, 1_R)$ be a ring with unity. Then a group ring $R[G]$ is the set of all formal sums*

$$\sum_{g \in G} r_g g,$$

where $r_g \in R$.

For $u = \sum_{g \in G} a_g g$, $v = \sum_{h \in G} b_h h$, $u, v \in R[G]$, $a_g, b_h \in R$. The addition $u \oplus v$ and multiplication $u \otimes v$ is defined as follows:

$$u \oplus v = \sum_{q \in G} (a_q + b_q) q,$$

$$u \otimes v = \sum_{q \in G} \left(\sum_{gh = q} a_g b_h \right) q.$$

Definition 2 (Period). *Let M be a square matrix. The least $k \in \mathbb{N}$ such that $M^i = M^{i+k}$, for some $i \in \mathbb{N}$ is called the period of M.*

Definition 3 (Pre-period). *Let M be a square matrix. The least $r \in \mathbb{N}$ such that there exists $k \in \mathbb{N}$ that $\forall i \geq r$, $M^{k+i} = M^i$ is called the pre-period of M.*

Definition 4 (Representation). *a representation of a group G of degree n is a homomorphism $\varphi : G \to GL(n,T)$; $\varphi(g) = \varphi_g$ for $g \in G$.*

Definition 5 (Equivalent representations). *Two representations $\varphi : G \to GL(n,T)$ and $\psi : G \to GL(m,T)$ are equivalent if $m = n$ and if $F \in GL(n,T)$ such that $\psi_g = F\varphi_g F^{-1}$, $\forall g \in \boldsymbol{G}$ exists.*

Definition 6 (φ-invariant subspace). *For a representation $\varphi : G \to GL(n,T)$ a subspace $S \leq T^n$ is φ-invariant if $\varphi_g s \in S$, $\forall g \in G$, $s \in S$.*

Definition 7 (Irreducible representation). *a representation $\varphi : G \to GL(n,T)$ is irreducible if and only if φ-invariant subspaces of T^n are $\{0\}$ and T^n.*

Definition 8 (Partition of number n). *Let $n \in \mathbb{N}$, then the partition λ of number n is defined as a non-increasing sequence of m positive integers $\lambda = (\lambda_1, \ldots, \lambda_m)$ such that $\lambda_1 + \ldots + \lambda_m = n$. We denote $\lambda \vdash n$.*

Theorem 1. *For $n \in \mathbb{N}$ and a field T of characteristics 0 or p, where p is a prime and $p > n$:*

- *each $\lambda \vdash n$ gives representation $\varphi^\lambda : S_n \to GL(n_\lambda, T)$ (for more details see [3, Theorem 4.12]),*
- *φ^λ is irreducible representation for all $\lambda \vdash n$,*
- *$\lambda \vdash n$, $\eta \vdash n$, $\eta \neq \lambda$, then φ^λ and φ^η are not equivalent,*
- *each irreducible representation of S_n over T is equivalent to some representation φ^λ,*
- *$TS_n \simeq \prod_{\lambda \vdash n} M_{n_\lambda}(T)$.*

III. Diffie-Hellman protocol

A. Discrete logarithm

Let $G = \langle g \rangle$ be a finite cyclic group of order n. Then for all elements $b \in G$ exists one and only one x in interval $(0, \ldots, n-1)$ such that $b = g^x$. The number x is called the *discrete logarithm* of element b in G. The task to compute x when G, g and b are given is called the *discrete logarithm problem*. We are not aware of any general method that could solve the discrete logarithm problem on a common computer in sub-exponential time.

B. Original Diffie-Hellman protocol

The requirement that two parties should be able to construct a secret shared key over a public channel resulted in the introduction of the Diffie-Hellman protocol in 1976. This protocol describes an exchange between two parties A and B leading to establishment of a secret shared key. Only A and B possess the key and it can not be retrieved by anyone who is listening to their conversation. The security of this protocol is based on the difficulty of the discrete logarithm problem.

The protocol works as follows:

- A and B decide on a finite cyclic group G and its generating element g,
- A picks a secret number $a \in (0, \ldots, |\, G \,| - 1)$ and sends $u = g^a$ to B,
- B picks a secret number $b \in (0, \ldots, |\, G \,| - 1)$ and sends $v = g^b$ to A,
- A computes $v^a = (g^b)^a = g^{ab}$,
- B computes $u^a = (g^a)^b = g^{ba}$,
- both A and B are in possession of the secret shared key g^{ab}.

Both A and B are using the algorithm *square and multiply* when computing g^a, g^b and g^{ab}.

An eavesdropper E, who is trying to retrieve the secret shared key g^{ab} from the knowledge of G, g, g^a, g^b, g^{ab}, is trying to solve the so called *Diffie-Hellman problem*.

The simplest and original implementation of the Diffie-Hellman protocol uses the class of groups $\mathbb{Z}_p^*$ as a platform. Working over groups from this class is convenient since we can easily calculate powers of its elements. Moreover, no fast algorithm that could solve the discrete logarithm problem in those groups is known. Nowadays, the protocol is considered secure, if p is at least a 300-digit number and a and b are at least 100-digit numbers. Unfortunately, these sizes of the parameters do not make this protocol suitable for devices with limited computational power. D. Kahrobaei et al. proposed a semigroup of small matrices as a platform for the protocol. Multiplication is fast in this structure hence the protocol is not that time consuming.

C. Modified Diffie-Hellman protocol

The structure proposed to work with is a semigroup of small matrices over the group ring $\mathbb{Z}_p[S_m]$ where $\mathbb{Z}_p$ is the ring of integers modulo p and S_m is the symmetric group of order $m!$. Parameters proposed in [6] are 3×3 matrices over $\mathbb{Z}_7[S_5]$ or over $\mathbb{Z}_2[S_5]$.

The main advantage of this structure is that we can precompute a multiplicative table for elements from S_5; hence multiplying two elements from $\mathbb{Z}_p[S_5]$ requires only multiplying elements from $\mathbb{Z}_p$ and searching in the multiplicative table.

The modified protocol works in the case $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ as follows:

- A and B decides on a matrix $M \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$,
- A picks a secret number $a \in \mathbb{N}$ and sends M^a to B,
- B picks a secret number $b \in \mathbb{N}$ and sends M^b to A,
- A computes $(M^b)^a = M^{ba}$,
- B computes $(M^a)^b = M^{ab}$,
- both A and B are in possession of the secret shared key M^{ab}.

It is important to note that M has to be chosen properly, i.e. that it has period larger than 10^{10}. Otherwise the attacker E could retrieve the secret shared key M^{ab} by

means of exhaustive search. The method to construct a suitable matrix $M \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ can be found in [6].

IV. Attack

The goal of our method is to retrieve the secret shared key M^{ab} with only the knowledge of $M_n(\mathbb{F}_q[\boldsymbol{S}_m])$, M, M^a and M^b. To do so, we do not have to find both a and b. We can find a' such that $M^{a'} = M^a$. Then, the secret shared key will be $(M^{a'})^b = (M^a)^b = M^{ab}$. Let us denote $N = M^a$.

The core idea of our method is that we used the representation theory of symmetric groups, which allows us to reduce the work an attacker has to do. We know that in order to break the Diffie-Hellman problem in $M_n(\mathbb{Z}_p[\boldsymbol{S}_m])$ we would have to solve the discrete logarithm problem in this semigroup in a reasonable amount of time. Since the order of $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ is approximately 10^{913}, we see it is not possible. However, the representation theory of the symmetric groups allows us to transform the problem onto a structure in which we are able to calculate the discrete logarithms in feasible time.

Firstly, we will describe our method for the case of $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$. Secondly, we will introduce the approach that solved the challenge given in [6] when using $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$ as a platform.

A. Case $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$

The characteristics of $\mathbb{Z}_7$ does not divide the order of S_5, so the theory of symmetric group representations gives us 7 irreducible representations φ_i, $i \in \{1, \ldots, 7\}$, i.e. homomorphisms

$$\varphi_i : \boldsymbol{S}_5 \to \mathrm{GL}(d_i, \mathbb{Z}_7),$$

where $d_i \in \{1, 4, 5, 6, 5, 4, 1\}$.

We can extend the homomorphisms in two steps as follows:

$$\varphi_i' : \mathbb{Z}_7[\boldsymbol{S}_5] \to M_{d_i}(\mathbb{Z}_7)$$

and

$$\psi_i : M_3(\mathbb{Z}_7[\boldsymbol{S}_5]) \to M_{3d_i}(\mathbb{Z}_7).$$

Then, according to [11, Theorem 3.9] and [4, Theorem 2.1.12], we obtain an algebra isomorphism $\psi = (\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6, \psi_7)$:

$$\begin{aligned}\psi : M_3(\mathbb{Z}_7[S_5]) \to \\ M_3(\mathbb{Z}_7) \times M_{12}(\mathbb{Z}_7) \times M_{15}(\mathbb{Z}_7) \times M_{18}(\mathbb{Z}_7) \\ \times M_{15}(\mathbb{Z}_7) \times M_{12}(\mathbb{Z}_7) \times M_3(\mathbb{Z}_7).\end{aligned} \tag{1}$$

We can see that the maximum order of matrices we will work with is 18 which is very small.

Note that the homomorphisms φ_i', $i \in \{1, \ldots, 7\}$ can be efficiently computed (for more details see [3, Chapter 8]). Now, we can map matrices M and N using the isomorphism ψ. We get two 7-tuples

$$\psi(M) = (M_{(1)}, \ldots, M_{(7)})$$
$$\psi(N) = (N_{(1)}, \ldots, N_{(7)}).$$

To construct a', we need to find numbers $a_i \in \mathbb{Z}$ such that $M_{(i)}^{a_i} = N_{(i)}$ for all $i \in \{1, \ldots, 7\}$.

To obtain a_i, $i \in \{1, \ldots, 7\}$ we will use the Menezes-Wu algorithm which can be found in [8].

To simplify the situation, we assume that 0 is not an eigen value of any of matrices $M_{(i)}$, $i \in \{1, \ldots, 7\}$. In fact, large powers of Jordan blocks with eigenvalue 0 are zero matrices, so the simplification has no essential effect regarding the attack.

In the following part, we will demonstrate the Menezes-Wu algorithm. We fix $i \in \{1, \ldots, 7\}$ and find one a_i, since all a_i, $i \in \{1, \ldots, 7\}$ can be obtained in the same manner. Also, we will show this method for both the cases of diagonal matrices $M_{(i)}$ and $N_{(i)}$ and non-diagonal matrices $M_{(i)}$ and $N_{(i)}$.

a) Diagonalizable matrices: Matrices $M_{(i)}$ and $N_{(i)}$ are diagonalizable if and only if their characteristic polynomials decompose into the product of linear factors and algebraic multiplicity of each eigen value is equal to its geometric multiplicity. In order to ensure that the characteristic polynomial will decompose to the product of linear factors we will work over the splitting field $\mathbb{F}$ of that polynomial. Let us denote k the rank of matrices $M_{(i)}$ and $N_{(i)}$, $\lambda_1, \ldots, \lambda_k$ eigen values of $M_{(i)}$ and $u_1, \ldots, u_k$ a basis of $\mathbb{F}^k$ composed of eigen vectors of $M_{(i)}$.

Let U be an invertible matrix that has eigen vectors $u_1, \ldots, u_k$ as columns. It holds that

$$U^{-1} M_{(i)} U = \begin{pmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_k \end{pmatrix}.$$

Then

$$U^{-1} M_{(i)}^a U = \begin{pmatrix} \lambda_1^a & 0 & \ldots & 0 \\ 0 & \lambda_2^a & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_k^a \end{pmatrix} = U^{-1} N_{(i)} U.$$

If we find $c \in \mathbb{N}_0$ such that $\lambda_j^c = \lambda_j^a$ for all $j \in \{1, \ldots, k\}$, it will hold that $M_{(i)}^c = N_{(i)}$ and c will be the required a_i for $M_{(i)}$ and $N_{(i)}$.

To obtain c we have to:

- find the characteristic polynomial q, eigen values $\lambda_1, \ldots, \lambda_k$ and eigen vectors $u_1, \ldots, u_k$ of $M_{(i)}$,
- equations

  $$N_{(i)} u_j = \lambda_j^a u_j, \forall j \in \{1, \ldots, k\}$$

  lead to finding $c_j \in \mathbb{N}_0$ such that $\lambda_j^{c_j} = \lambda_j^a$ for all λ_j, $j \in \{1, \ldots, k\}$. In order to find c_j, $j \in \{1, \ldots, k\}$, we have to solve the discrete logarithm problem in groups of order $\mathrm{ord}(\lambda_j)$, $j \in \{1, \ldots, 7\}$ (note that for every irreducible factor q_j of q we work in $\mathbb{Z}_7[x]/(q_j)$ where $x + (q_j)$ represents the eigen value λ_j),

- the fact that $\text{ord}(\lambda_j) \mid (c_j - c)$,$\forall j \in \{1,\dots,k\}$ allows us to put together a system of diophantine equations

$$c = c_j - \text{ord}(\lambda_j) \cdot h_j, \forall j \in \{1,\dots,k\},$$

where $h_j \in \mathbb{Z}$. By solving this system of equations (see [1, Algorithm 2.4.10]) we will get c such that $\lambda_j^c = \lambda_j^a$, for all $j \in \{1,\dots,k\}$.

So, in order to find c, we need to know the orders of eigen values λ_j, $j \in \{1,\dots,k\}$ and we have to be able to solve the discrete logarithm problem in groups of orders $\text{ord}(\lambda_j)$, $j \in \{1,\dots,k\}$.
For a fixed $j \in \{1,\dots,7\}$ we find the order of eigen value λ_j using the fact that $\text{ord}(\lambda_j)$ divides $\mid T_r^* \mid$, where T_r denotes field $\mathbb{Z}_7(\lambda_j) \simeq \mathbb{Z}_7[x]/(q_j)$, where q_j is the minimal polynomial of λ_j in $\mathbb{Z}_7$. Computing $\text{ord}(x)$ in $(\mathbb{Z}_7[x]/(q_j))^*$ gives us orders of all roots of polynomial q_j.
When computing the discrete logarithm in groups of orders $\text{ord}(\lambda_j)$, consider $\text{ord}(\lambda_j) =\mid T_r^* \mid$ represents the worst case. Denote

$$|T_r^*| = s_1^{l_1} \cdot s_2^{l_2} \cdot \ldots \cdot s_n^{l_n}$$

the factorization of the order of T_r^*. Now we can use the Pohlig-Hellman reduction and reduce the computations into group orders of which will be at most $s_m^{l_m}$ for some $m \in \{1,\dots,n\}$. Also, for a polynomial q of degree d we have $\mid T_r^* \mid = 7^d - 1$. Since $M_{(i)}, i \in \{1,\dots,7\}$ have degrees at most 18, then $d \leq 18$ and prime factors of $7^d - 1$ for $d \leq 18$ are small enough for us to be able to solve the discrete logarithm problems in a reasonable amount of time when using a common computer.

b) Non-diagonalizable matrices: We will outline the method for non-diagonalizable matrices in this section.
Let us fix $i \in \{1,\dots,7\}$. Assume that we have a basis B such that $[H]_B$ is a matrix H expressed in terms of the basis B which has a Jordan canonical form.
Let us have an invertible matrix U which has vectors of basis B as columns. Then it holds that

$$U^{-1}M_{(i)}U = \begin{pmatrix} J_1 & 0 & \dots & 0 \\ 0 & J_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & J_k \end{pmatrix}$$

is a block diagonal matrix with Jordan blocks J_j, $j \in \{1,\dots,k\}$ on diagonal and

$$U^{-1}N_{(i)}U = \begin{pmatrix} J_1^a & 0 & \dots & 0 \\ 0 & J_2^a & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & J_k^a \end{pmatrix}.$$

Now, we can find $c_j \in \mathbb{N}_0$ for each Jordan block J_j, $j \in \{1,\dots,k\}$ such that $J_j^{c_j} = J_j^a$.

Since for Jordan block

$$J_j = \begin{pmatrix} \lambda & 1 & 0 & & 0 & 0 \\ 0 & \lambda & 1 & \dots & 0 & 0 \\ 0 & 0 & \lambda & & 0 & 0 \\ & \vdots & & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \lambda & 1 \\ 0 & 0 & 0 & \dots & 0 & \lambda \end{pmatrix}$$

we have

$$J_j^a = \begin{pmatrix} \lambda^a & \binom{a}{1}\lambda^{a-1} & \binom{a}{2}\lambda^{a-2} & \dots & \binom{a}{k-1}\lambda^{a-k+1} \\ 0 & \lambda^a & \binom{a}{1}\lambda^{a-1} & \dots & \binom{a}{k-2}\lambda^{a-k+2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda^a & \binom{a}{1}\lambda^{a-1} \\ 0 & 0 & \dots & 0 & \lambda^a \end{pmatrix},$$

and c_j has to ensure equality of all elements in upper triangular matrices J_j^a and $J_j^{c_j}$. Note that if J_j^a and $J_j^{c_j'}$ have same values on the diagonal, we may find c_j of the form $c_j' + z \cdot \text{ord}(\lambda)$ for $z \in (0,1,...,6)$.
Then, we can compute $c \in \mathbb{N}_0$ such that $J_j^c = J_j^a$, $j \in \{1,\dots,k\}$ and c will be the required a_i for $M_{(i)}$ and $N_{(i)}$.
Before we proceed to the construction of a', we will show how to compute the period of $M_{(i)}$ for some $i \in \{1,\dots,7\}$, first.

c) Matrix period: Assuming that $M_{(i)}$ is diagonalizable we have its Jordan canonical form

$$C = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_k \end{pmatrix}.$$

We can see that all Jordan blocks are of degree 1 and the Jordan canonical form of an m^{th} power of $M_{(i)}$ is

$$C^m = \begin{pmatrix} \lambda_1^m & 0 & \dots & 0 \\ 0 & \lambda_2^m & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_k^m \end{pmatrix}.$$

This means that for $M_{(i)}$ it holds that

$$\text{per}(M_{(i)}) = \text{LCM}(\text{ord}(\lambda_1),\dots,\text{ord}(\lambda_k)).$$

Assuming that $M_{(i)}$ is non-diagonalizable, it holds that

$$\text{per}(M_{(i)}) = \text{LCM}(\text{per}(J_1),\dots,\text{per}(J_k)),$$

where J_j, $j \in \{1,\dots,k\}$ are Jordan blocks of non-zero eigen values of $M_{(i)}$. The periods of the Jordan blocks can be found using the following method.
Fixing $j \in \{1,\dots,k\}$, we denote p_j the period of the Jordan block J_j and λ_j the element on its diagonal. Then, it has to hold that

$$p_j = k \cdot \text{ord}(\lambda_j)$$

for $k \in \mathbb{N}$.
Having p_j as a multiple of $\text{ord}(\lambda_j)$ will ensure that the elements on the diagonal of J_j will be the same as the

elements on the diagonal of $J_j^{p_j+1}$. Finding a fitting $k \in \mathbb{N}$ will ensure that all elements above the diagonal of $J_j^{p_j}$ will be zero and we will get $J_j = J_j^{p_j+1}$.
Initialize $p_j = \mathrm{ord}(\lambda_j)$ and split the calculation of k in a few cases. It is important to keep in mind that we are computing over a field with characteristic 7, so that all operations with integers are modulo 7.

1) 7 divides p_j and the rank of J_j is at most 7:
   in this case 7 divides all binomial coefficients in $J_j^{p_j}$, hence $k = 1$ and $\mathrm{per}(J_j) = p_j$;
2) 7 divides p_j and the rank of J_j is greater than 7:
   denote $c_j = p_j/7$. A problem can appear when working with binomial coefficient $\binom{p_j}{7}$. We know that 7 divides p_j, hence

$$\binom{p_j}{7} = \frac{p_j \cdot (p_j-1)(p_j-2) \cdot \ldots \cdot (p_j-6)}{7 \cdot 6 \cdot 5 \cdot \ldots \cdot 1} = \frac{c_j \cdot (p_j-1)(p_j-2) \cdot \ldots \cdot (p_j-6)}{6 \cdot 5 \cdot \ldots \cdot 1}.$$

   For $\binom{p_j}{7} = 0$, we need $7 \mid c_j$. If 7 does not divide c_j, we set $k = 7$ which will lead to $\mathrm{per}(J_j)$ being a multiple of $7p_j$.
   It is easy to see that $7 \mid \binom{7p_j}{14}$ therefore $\mathrm{per}(J_j) = kp_j$.
3) 7 does not divide $\mathrm{ord}(\lambda_j)$ and the rank of J_j is at least 2:
   initialize $p_j = 7 \cdot \mathrm{ord}(\lambda_j)$. This case is described in cases 1 and 2.

The maximum rank of Jordan blocks is 18, so this method includes all possibilities.

d) Finding a': At this point, we have obtained $a_1, \ldots, a_7$ such that

$$(M_{(1)}^{a_1}, \ldots, M_{(7)}^{a_7}) = (N_{(1)}, \ldots, N_{(7)}).$$

We denote $p_i = \mathrm{per}(M_{(i)})$, $i \in \{1, \ldots, 7\}$. We may assume $a_1, \ldots, a_7 \geq$ pre-period(M). Then there exist constants $l_i \in \mathbb{N}_0$, $i \in \{1, \ldots, 7\}$, such that

$$a' = a_1 + l_1 p_1 = a_2 + l_2 p_2 = \ldots = a_7 + l_7 p_7. \quad (2)$$

Equation (2) can be written as a system of diophantine equations

$$a_1 + l_1 p_1 = a_2 + l_2 p_2 = \ldots = a_7 + l_7 p_7,$$

where $l_1, \ldots, l_7$ are calculated. After substituting any l_i, $i \in \{1, \ldots, 7\}$ in (2), we get a' as

$$a' = x + my,$$

where $x, y, m \in \mathbb{Z}$, and x and y are computed and m is a parameter. Choosing m such that $a' \geq a_1 \ldots, a_7$ we find the secret shared key $(M^b)^{a'} = (M^b)^a = M^{ab}$ as was required.

B. Case $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$

In this case, we can again find homomorphisms φ_i, $i \in \{1, \ldots, 7\}$

$$\varphi_i : \boldsymbol{S}_5 \to \mathrm{GL}(d_i, \mathbb{Z}_2),$$

where $d_i \in \{1, 4, 5, 6, 5, 4, 1\}$
and extend them as before

$$\varphi_i' : \mathbb{Z}_2[\boldsymbol{S}_5] \to M_{d_i}(\mathbb{Z}_2)$$

and

$$\psi_i : M_3(\mathbb{Z}_2[\boldsymbol{S}_5]) \to M_{3d_i}(\mathbb{Z}_2).$$

We get

$$\psi : M_3(\mathbb{Z}_2[S_5]) \to M_3(\mathbb{Z}_2) \times M_{12}(\mathbb{Z}_2) \times M_{15}(\mathbb{Z}_2) \times M_{18}(\mathbb{Z}_2) \times M_{15}(\mathbb{Z}_2) \times M_{12}(\mathbb{Z}_2) \times M_3(\mathbb{Z}_2).$$

However, since $\mathrm{char}(\mathbb{Z}_2) \mid \mathrm{ord}(S_5)$, the representations φ_i, $i \in \{1, \ldots, 7\}$ will not be irreducible and ψ will not be an isomorphism. Because of that we can not use the method mentioned above.
We will present a method of how to solve the challenge given in appendix of [6]. In this challenge, authors presented matrices M, M^a, M^b and asked a reader to find the secret shared key M^{ab}. To do so, we will again search a' such that $M^{a'} = M^a$.
The method works as follows:

- we calculate $\dim(\mathrm{Ker}(\psi)) = 78$ and denote $l = 128$, the smallest power or 2 greater than 78,
- using the method described in [10] we embed $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$ into $M_{360}(\mathbb{Z}_2)$ in order to calculate the pre-period y of M; note that we do not need to compute the period of M,
- we find b such that $\psi(M)^y = \psi(M)^{y+b}$ using the method mentioned in the previous section,
- this gives us a nilpotent matrix $C = M^y - M^{y+b}$,
- equation

$$0 = C^{128} = M^{128y} - M^{128y+128b}$$

  leads to finding $\mathrm{per}(M) \mid 128b$ of M,
- since $b = 75565$ then $\mathrm{per}(M) = 9672320$ which is period small enough for us to be able to find $a' = 217183$ by means of exhaustive search.

C. Implementation

To support our result, we implemented the attack in both cases $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ and $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$. We used Microsoft Visual C++ 2012 with NTL and MPIR libraries and Wolfram Mathematica 8.
We followed the method presented in [6] and constructed $M \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ as a product $M = M_1 \cdot S$, where $M_1 \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ is an invertible matrix and S is a scalar matrix with an element $s = (3+g_1)(3+g_2)(3+g_3)(3+g_4)(3+g_5)(3+g_6)(5+h)$ on its diagonal. Elements $g_i \in S_5$, $i \in \{1, \ldots, 6\}$ generate different subgroups of order 5 in S_5 and the element

h is a product of two independent cycles of lengths 2 and 3. For our particular choice of the parameters see [7, Page 15].
We picked $a = 387060858948298925004416 5641$ and obtained matrices $M_{(i)}$ and $N_{(i)}$ diagonalizable for $i \in \{1, \ldots, 7\}$ therefore we followed the method presented in IV-A0a and we got

$$a' = 350310065731473567845307248788 2159$$
$$935195562645858532491241278585 04$$
$$+414872873390037779882720801600m$$

as a result.

V. Baby-step giant-step

Nowadays, we are aware of multiple algorithms that speed up solving the discrete logarithm problem. One of them is the baby-step giant-step algorithm.
In this method, for a cyclic group $G = \langle g \rangle$ of order n and an element $b \in \langle g \rangle$, we try to find $x \in \mathbb{N}$ such that $b = g^x$ using the fact that x can be expressed as

$$x = im + j, \tag{3}$$

where $m = \lceil \sqrt{n} \rceil$, $i, j \in \{0, \ldots, m-1\}$.
Equation 3 leads to

$$b = g^{im+j} \Leftrightarrow bg^{-im} = g^j.$$

The algorithm baby-step giant-step then proceeds to so called baby-steps where it computes and stores values (j,g^j) for $j \in \{0, \ldots, m-1\}$. Baby-steps are followed by so called giant-steps which calculate values bg^{-im} for $i \in \{0, \ldots, m-1\}$ and also compare those values to stored g^j. When we hit equality

$$bx^{-im} = g^j$$

for some $i, j \in \{0, \ldots, m-1\}$, we have found $x = im + j$ such that $g^x = b$.
We can transform the situation according to [6] and work in $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$. In this case, we have $M, N \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$, $n = |M_3(\mathbb{Z}_7[\boldsymbol{S}_5])|$ such that $M^x = N$ and $x \in \mathbb{N}$ can again be expressed as in (3).
In [6], the authors used an analogy of the baby-step giant-step algorithm that is based on equation

$$N = M^{im+j} \Leftrightarrow NM^{-j} = M^{im},$$

where M is a regular matrix.
Baby-steps then calculate and store values (J,NM^j) for $j \in \{1, \ldots, m-1\}$ and giant-steps calculate M^{im} for $i \in \{1, \ldots, \lceil n/m \rceil\}$ and compare them to values NM^j. When we hit equality

$$NM^j = M^{im}$$

for some $i \in \{1, \ldots, \lceil n/m \rceil\}$, $j \in \{1, \ldots, m-1\}$, we get $x = im - j$ for which $M^x = N$ holds.
Algorithm 1 shows how the baby-step giant-step works for the modified Diffie-Hellman protocol according to [6].
The algorithm requires that

Algorithm 1: Baby-step giant-step

Input: M, $N \in M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$, $n = |M_3(\mathbb{Z}_7[\boldsymbol{S}_5])|$
Output: $x \in \mathbb{N}$, such that $M^x = N$

$m = \lceil \sqrt{n} \rceil$;
$t = \lceil n/m \rceil$;
for $j = 1, \ldots, m-1$ **do**
  Compute NM^j;
  Store (j, NM^j);
for $i = 0, \ldots, t$ **do**
  Compute $M_i = M^{im}$;
  if there exists j such that $M_i = NM^j$ **then**
    return $im - j$.

$$M^{im} = NM^j \Rightarrow N = M^{im-j}$$

holds.
Since M does not have to be regular, hence invertible, it seems that this implication is not obvious. However, if we consider Jordan canonical forms, we get

$$(U^{-1}MU)^{im} = U^{-1}NU\left(U^{-1}MU\right)^j \tag{4}$$

for basis U. This can be illustrated as

$$\begin{pmatrix} \boxed{\textbf{sing}} & 0 \\ 0 & \boxed{\textbf{reg}} \end{pmatrix}^{im} =$$
$$\begin{pmatrix} \boxed{\textbf{sing}} & 0 \\ 0 & \boxed{\textbf{reg}} \end{pmatrix}^{x} \begin{pmatrix} \boxed{\textbf{sing}} & 0 \\ 0 & \boxed{\textbf{reg}} \end{pmatrix}^{j},$$

where **reg** denotes a section that appertains to nonzero eigen values and **sing** denotes a section that appertains to zero eigen values. We can see that if $N = M^x$ is large enough power of M and if m is large enough, we can illustrate (4) as follows:

$$\begin{pmatrix} \boxed{0} & \\ & \boxed{\textbf{reg}^{im}} \end{pmatrix} = \begin{pmatrix} \boxed{0} & \\ & \boxed{\textbf{reg}^{x}} \end{pmatrix} \begin{pmatrix} \boxed{?} & \\ & \boxed{\textbf{reg}^{j}} \end{pmatrix}.$$

Then

$$\begin{array}{c} M^{im} = NM^j \\ \Updownarrow \\ \textbf{reg}^{im} = \textbf{reg}^x\textbf{reg}^j \\ \Updownarrow \\ \textbf{reg}^{im-j} = \textbf{reg}^x. \end{array}$$

This means that if $im - j$ is large enough and $\textbf{sing}^{im-j} = 0$, then $M^{im-j} = N$.
According to [6], the baby-step giant-step algorithm is not usable for the modified Diffie-Hellman protocol. The main reason is that this algorithm has huge memory requirements whilst working over $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$.
It is obvious that the knowledge of a period of M would

significantly simplify the situation. Instead of searching in the whole semigroup $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$, we could just search in space of a size $\mathrm{per}(M)$. We have shown a method for computing the period of M in IV-A0c therefore the baby-step giant-step will be more effective than the authors of [6] claimed.

VI. Related work

The security of the modified Diffie-Hellman protocol proposed in [6] was also analysed in [10] and [9].

In [10], A. Myasnikov and A. Ushakov proposed an embedding of $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ into $M_{360}(\mathbb{Z}_7)$. This embedding, together with the Menezes-Wu algorithm, allowed the authors to find the secret shared key using a quantum computer in polynomial time. This paper proves that the modified Diffie-Hellman protocol does not belong to the realm of post-quantum cryptography.

In [9] can be found a method for attacking the modified protocol which is based on the same core idea as our method. The authors first constructed an embedding ψ of $M_3(\mathbb{Z}_7[\boldsymbol{S}_5])$ into $M_{360}(\mathbb{Z}_7)$ as proposed in [10] and then they constructed an isomorphism between $\mathrm{Im}(\psi)$ and $M_3(\mathbb{Z}_7) \times M_{12}(\mathbb{Z}_7) \times M_{15}(\mathbb{Z}_7) \times M_{18}(\mathbb{Z}_7) \times M_{15}(\mathbb{Z}_7) \times M_{12}(\mathbb{Z}_7) \times M_3(\mathbb{Z}_7)$. Having this isomorphism they were able to retrieve the secret shared by computing the minimal polynomial of $A \in \mathrm{Im}(\psi)$. The authors also worked with $M_3(\mathbb{Z}_2[\boldsymbol{S}_5])$ and solved the challenge given in appendix of [6]. However, our work is in scope of [7] and we worked independently from [9].

VII. Conclusion

We have recalled the modified Diffie-Hellman protocol proposed in [6] which is trying to make the original Diffie-Hellman protocol suitable for devices with limited computational power. To do so, the authors of [6] proposed $M_n(\mathbb{Z}_p[\boldsymbol{S}_m])$ as a platform. However, this modification met the computational costs requirements, it decreased the security level of the key exchange itself. We have shown that with help of the theory of symmetric group representations we can exploit the algebraic properties of $M_n(\mathbb{Z}_p[\boldsymbol{S}_m])$ and construct the secret shared key on a common computer in feasible time. The same result was presented in [9]. Consequently, the modified protocol is not as secure as is claimed in [6] when $p > m$. Any improvement of this modification to resist this attack is not clear. Our brief calculation for $m = 5$ and $p = 2$ indicates that choosing the parameters $p < m$ is probably not sufficient to make the protocol secure.

References

[1] H. Cohen, "A Course in Computational Algebraic Number Theory", 1st ed., Springer, Berlin, 1996.

[2] W. Diffie, M. E. Hellman, "New directions in cryptography", in *IEEE Transaction on Information Theory*, vol. IT-22, no. 6, pp 644-654, Nov. 1976.

[3] G. D. James, "The Representation Theory of the Symmetric Group", in *Lecture Notes in Mathematics 682*, Springer, 1978.

[4] G. D. James, A. Kerber, "The Representation Theory of Symmetric group", Cambridge University Press, 2009.

[5] A. Joux, A. Odlyzko, C. Pierrot, "The past, evolving present, and future of the discrete logarithm", in *Open Problems in Mathematics and Computational Science*, pp 5-36, 2014.

[6] D. Kahrobaei, C. Kouppari, V. Shpilrain, "Public key exchange using matrices over group rings", in *Groups, Complexity and Cryptology*, vol. 5, pp 97-115, 2013.

[7] R. Linkeová (2014, May), "Diffie a Hellman si vyměňují matice nad grupovým okruhem". [Online]. Available: https://is.cuni.cz/webapps/zzp/detail/141169/

[8] A. J. Menezes, Y. Wu, "The discrete logarithm problem in $\mathrm{Gl}_n(\mathbb{F}_q)$" in *Ars Combinatoria*, vol. 47, pp 23-32, 1997.

[9] Ch. Monico, M. D. Neusel, "Cryptanalysis of a system using matrices over group rings", in *Groups, Complexity, Cryptology*, vol. 7, pp 175-182, 2015.

[10] A. Myasnikov, A. Ushakov (2012, Oct.), "Quantum algorithm for discrete logarithm problem for matrices over finite group rings". [Online]. Available: http://eprint.iacr.org/2012/574

[11] S. H. Weintraub, "Representation Theory of Finite Groups: Algebra and Arithmetic (Graduate Studies in Mathematics)", in *Amer Mathematical Society*, vol. 59, 2003.

Romana Linkeová is a master student at Charles University in Prague. She graduated her bachelor studies in 2014. Her study field is Mathematical Methods in Information Security. You can contact her: linkeovaromana@gmail.com

Pavel Příhoda is an associate professor at Department of Algebra, Charles University in Prague where he also got his PhD in mathematics. In 2006 - 2007 he was a PostDoc researcher at Centre de Reserca Matematica, Barcelona. His research field is algebra, in particular module theory. You can contact him: prihoda@karlin.mff.cuni.cz