A Small Serving of Mash: (Quantum) Algorithms for SPDH-Sign* with Small Parameters

Andrew Mendelsohn[1], Edmund Dable-Heath[2], and Cong Ling[1]

[1] Department of EEE, Imperial College London, United Kingdom
[2] The Alan Turing Institute, United Kingdom
am3518@ic.ac.uk, edable-heath@turing.ac.uk, cling@ieee.org

Abstract. We find an efficient method to solve the semidirect discrete logarithm problem (SDLP) over finite nonabelian groups of order p^3 and exponent p^2 for certain exponentially large parameters. This implies an attack on SPDH-Sign, a signature scheme based on the SDLP, for such parameters. We also take a step toward proving the quantum polynomial time equivalence of SDLP and SCDH.

Keywords: semidirect product & discrete logarithm & signatures

1 Introduction

In [12], the authors introduced a key exchange protocol. The security of their scheme was based on a discrete logarithm problem: given a group element g which generates a finite group G, and the element g^x for some $x \in \mathbb{N}$, can one recover x? Efficient classical solutions to the general discrete logarithm problem remain elusive, but Shor [22] gave an efficient quantum algorithm to solve the above problem. Thus cryptography relying on the above assumption is not quantum-secure.

The field of post-quantum cryptography comprises several distinct topics: lattice, isogenies of elliptic curves, multivariate polynomials, and codes have all been used to develop cryptosystems believed no more vulnerable to attack by quantum adversaries than by classical adversaries. Another line of work refers back to the discrete logarithm problem above, asking: can the discrete logarithm problem be tweaked to yield a quantum-hard cryptographic problem?

One contribution to this is the semidirect discrete logarithm problem (SDLP). Informally, for some finite group element $g \in G$ and an element in the automorphism group of G, $\phi \in \mathrm{Aut}(G)$, given the element

$$s_{g,\phi}(x) := \phi^{x-1}(g) \cdot \phi^{x-2}(g) \cdot ... \cdot \phi(g) \cdot g$$

can an adversary recover x? Note when ϕ is the identity map we recover the standard discrete logarithm problem.

* Pronounced 'SPUD-Sign'.

This problem was recently analysed in [2] and [3]. In the former paper, the authors gave a subexponential- (but not polynomial-) time algorithm for SDLP. In the latter paper, the authors develop a signature scheme, SPDH-Sign, based on the hardness of the SDLP problem. In particular, the authors use the group

$$G = G_p := \left\{ \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix} : a, b \in \mathbb{Z}_{p^2}, a \equiv 1 \bmod p \right\}$$

to instantiate the SDLP problem. To ensure a suitable level of security, one takes p to be a 'cryptographic'-sized prime.

In the current paper, we contribute to the cryptanalysis of that scheme by performing further analysis on the SDLP problem.

Contributions In this work we provide four contributions to the study of SDLP. The first of these is to show that the structure of G_p enables an adversary to recover $x \bmod p-1$ from $s_{g,\phi}(x)$ in SDLP instances defined on elements of G_p. This allows one to recover x when x is defined modulo a small multiple of p. This is because of the semidirect product isomorphism

$$G_p \cong \mathbb{Z}/p^2\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}$$

which is efficiently computable. We obtain

Theorem 5. Let $(g, \phi) \in G_p \rtimes \mathrm{Aut}(G_p)$, where $g = (a, \varphi) \in G_p$. Then there is a quantum polynomial time algorithm to find $x \bmod p-1$.

We then show that one can recover $\phi^x(g)$ from the available information, and that this also leaks information on x due to the structure of the automorphisms of G_p. In both of the above cases, we can recover x only when it is defined modulo a small multiple of p. When the security parameter of a scheme is denoted by λ, one has $p = \exp(\lambda)$; so our attacks hold against exponentially large parameter sizes. However, since $|G_p| = (p-1)p^6$, one may take x to be defined at most modulo $(p-1)p^5$ (see below for more details) and in these larger parameter instances we do not currently see how to recover all of x.

After this we turn to abstract properties of the SDLP problem, which we consider as a group action problem. We consider the 'linear hidden shift' (LHS) problem, and find that, as a corollary to our cryptanalytic attack, we can solve LHS in quantum polynomial time. The LHS problem, informally, is given (g_i, ϕ_i), $\mathbf{x}_i \in (\mathbb{Z}/n\mathbb{Z})^m$, and $s_{g,\phi}(\sum_j s_j x_{i_j})$ where $i = 1, ..., m$, to recover $\mathbf{s}$. We have

Theorem 6. Let $g \in G_p$ and $\phi \in \mathrm{Aut}(G_p)$, where $g = (a, \varphi) \in G_p$. Let $m \geq n$. Then there is a quantum polynomial time algorithm to solve $\mathrm{LHS}_{G,X,\mathbf{s}}$.

We then turn to an open problem from [3]. In addition to SDLP, another problem, SCDH, was considered. This is the problem, given $g, \phi, s_{g,\phi}(x)$ and $s_{g,\phi}(y)$, of computing $s_{g,\phi}(x+y)$. Of course, if one can solve SDLP, one may simply compute x from $s_{g,\phi}(x)$ and y from $s_{g,\phi}(y)$ and then compute $s_{g,\phi}(x+y)$ directly;

but it is unknown if a solution to SCDH implies a solution to SDLP. We partially resolve this problem by demonstrating a quantum algorithm which, given an oracle for a particular form of SCDH which given $s_{g,\phi}(a)$, returns $s_{g,\phi}(2a)$ for any a, reduces SDLP to a hidden subgroup problem instance which can be efficiently solved with Shor's period finding algorithm:

Theorem 7. There is a quantum polynomial-time reduction from $\mathrm{SDLP}_{g,\phi,n,x}$ to $\mathrm{SCDH}^2_{g,\phi,n}$.

We close by discussing the obstacles to a direct solution to SDLP via Shor's algorithm.

Prior Work There is a burgeoning literature on noncommutative variants of the semidirect product discrete logarithm problem, or schemes based on similar problems [19], [14], [16], [2], [3], [5]. Attacks on variants of this problem have been given in [21], [4], [7]. The literature on cryptographic group actions includes [10], [1], [13], [8], [11].

In a concurrent work (uploaded to the IACR Eprint server prior to our work), the authors Imran and Ivanyos [15] also provided cryptanalysis of the SDLP problem. We note the similarity to our work, and note the greater generality of their approach, which applies to a variety of finite groups. However, our paper includes results (on outer automorphisms, and relating SDLP and SCDH, for example) not covered by [15], and we consider our methods tailored to the choice of group suggested for SPDH-Sign a valuable contribution to the study of SDLP.

2 Preliminaries

Notations We may write $[n]$ to denote the set $\{1, ..., n\}$. The arrow '$\leftarrow$' may denote sampling from a set or sampling according to a distribution over a set; context will make which clear. If we write '$\xleftarrow{\$}$' we mean sampling uniformly at random. The identity element of a group G will be denoted by e.

Group Endomorphisms To any finite group are attached endomorphisms:

Definition 1. An endomorphism $\phi : G \to G$ is a homomorphism of groups from G to G.

If a group endomorphism ϕ is an isomorphism, we call ϕ an automorphism. The collection of all automorphisms of a finite group G forms a group, denoted $\mathrm{Aut}(G)$.

The Semidirect Product We define the semidirect prodcuct of two groups.

Definition 2. (semidirect product) Let G, H be finite groups. If there is an injective homomorphism

$$\rho : H \hookrightarrow \mathrm{Aut}(G)$$

then we can form a product of G and H, $G \rtimes_\rho H$, defined by the following multiplication rule: for $(g, \phi), (h, \psi) \in G \times H$,

$$(g, \phi) \cdot (h, \psi) = (\psi(g)h, \phi\psi)$$

Here $\psi(\cdot)$ is the action of the automorphism; this could be exponentiation (g^ψ) or conjugation ($\psi g \psi^{-1}$) or something more complicated. Note that this new group is noncommutative - that is, swapping the order of multiplication can change the resulting group element on the right hand side. If $H \subseteq \mathrm{Aut}(G)$, we can take ρ as the identity map and write $G \rtimes H$. In the literature, the product $G \rtimes \mathrm{Aut}(G)$ is sometimes called the holomorph of G, and denoted $\mathrm{Hol}(G)$. This construction is called the *external* semidirect product of G and H. It is a standard fact that $|G \rtimes H| = |G||H|$.

Group Actions We define and give properties of group actions.

Definition 3. (group action) A group action of a finite group G on a set X (sometimes called a G-set) is a map $\star : G \times X \to X$ satisfying

1. for any $x \in X$, $e \star x = x$, and
2. for any $g, h \in G$ and any $x \in X$, $(gh) \star x = g \star (h \star x)$.

A group action is effective if $|G| < \infty$ and standard group-theoretic operations can be performed in polynomial time. The following are standard properties of group actions:

Definition 4. A group action of G on X is

1. transitive, if for any $x_1, x_2 \in X$ there exists a $g \in G$ satisfying $x_2 = g \star x_1$;
2. faithful, if one has $g \star x = x$ for all $x \in X$ if and only if $g = e$;
3. free, if one has $g = e$ if and only if there exists an $x \in X$ such that $x = g \star x$.

A free and transitive group action is called regular.

SDLP and SCDH Recall the discrete logarithm problem in a finite abelian group G. Fix $g \in G$, which we will consider to be public. A challenger selects an integer x, computes $h = g^x$, and gives h to an adversary. The adversary has to recover x, which is defined modulo the order of $g \in G$. This can be solved in quantum polynomial time via Shor's algorithm [22], but is classically only solvable in subexponential time.

The authors of [3] use a version when G is replaced with $G \rtimes H$. Let $(g, \phi) \in G \rtimes H$. Select, for instance, $x = 2$, and compute

$$(g, \phi)^2 = (g, \phi) \cdot (g, \phi) = (\phi(g)g, \phi^2)$$

If a challenger gave an adversary the resulting group element, they could take the second component ϕ^2, solve the (abelian) discrete logarithm problem in H,

and find that $x = 2$. Alternatively, they could solve a discrete logarithm problem in the cyclic group generated by (g, ϕ), denoted $\langle (g, \phi) \rangle$. More generally, for an arbitrary choice of x, we have

$$(g, \phi)^x = (\phi^{x-1}(g) \cdot ... \cdot \phi(g)g, \phi^x)$$

Clearly if $x < |H|$, an adversary could always solve an abelian discrete logarithm problem to find x. If $x \geq |H|$, they could solve an abelian discrete logarithm to find $x \bmod |H|$. So one cannot release the second coordinate of $(g, \phi)^x$ and maintain secrecy of x. This leads to

Definition 5. (SDLP) The semidirect product discrete logarithm problem, $\mathrm{SDLP}_{g,\phi,x}$, is, given

$$s_{g,\phi}(x) := \phi^{x-1}(g) \cdot ... \cdot \phi(g)g$$

for some $x \in \mathbb{Z}^+$ and $(g, \phi) \in G \rtimes H$, to find x.

One can see that x is only defined modulo $|G \rtimes H| = |G| \times |H|$. Moreover, it is in fact only defined modulo the order of the group element chosen, $o(g, \phi)$, since if $x > o(g, \phi)$ then $(g, \phi)^x = (g, \phi)^{x \bmod o(g,\phi)}$. As a consequence, we may take $x \in \mathbb{Z}/n\mathbb{Z}$ for some $n \mid o(g, \phi)$. We denote such a problem instance by $\mathrm{SDLP}_{g,\phi,n,x}$.

A related problem to SDLP is the semidirect computational Diffie-Hellman (SCDH) problem:

Definition 6. (SCDH). Let G be a finite group, and let $(g, \phi) \in G \rtimes \mathrm{Aut}(G)$. Let $x, y \in \mathbb{N}$ and suppose we are given $(g, \phi), s_{g,\phi}(x)$, and $s_{g,\phi}(y)$. The Semidirect Computational Diffie-Hellman problem, $\mathrm{SCDH}_{g,\phi,n,x,y}$, is to compute $s_{g,\phi}(x+y)$.

In [2], a subexponential quantum algorithm was given for SDLP over semigroups. In the following, a family of (semi)groups indexed by κ is 'easy' if for a fixed κ, pairs $(g, \phi), (g', \phi') \in G_\kappa \times \mathrm{End}\,(G_\kappa)$, and values $f(\kappa), f'(\kappa)$ (resp. $g(\kappa), g'(\kappa)$) denoting the number of operations required to solve SDLP (resp. SCDH) for (g, ϕ) and (g', ϕ') respectively, then we have $f(\kappa) = \mathcal{O}\,(f'(\kappa))$ (resp. $g(\kappa) = \mathcal{O}\,(g'(p))$). Then:

Theorem 1. *[2, Theorem 10] Let $\{G_\kappa\}_\kappa$ be an easy family of semigroups, and fix κ. For any pair $(g, \phi) \in G_\kappa \times \mathrm{End}\,(G_\kappa)$, there is a quantum algorithm solving SDLP with respect to (g, ϕ) with time and query complexity $2^{\mathcal{O}(\sqrt{\log \kappa})}$.*

We also note a group action interpretation of SDLP. Define

$$\mathcal{X}_{g,\phi} := \{s_{g,\phi}(i) : i \in \mathbb{Z}/n\mathbb{Z}\}.$$

Then

Definition 7. Let $(g, \phi) \in G \rtimes H$ and $n \mid o(g, \phi)$. Define a group action of $\mathbb{Z}/n\mathbb{Z}$ on $\mathcal{X}_{g,\phi}$ by

$$\mathbb{Z}/n\mathbb{Z} \circlearrowright \mathcal{X}_{g,\phi} : x * (g, \phi) = \phi^{x-1}(g) \cdot ... \cdot \phi(g)g$$

This group action is free and transitive. We call this group action the semidirect product group action (SDPGA).

SPDH-Sign In [3], a signature scheme was designed based on SDLP. The key generation and signing algorithms require multiple instances of SDLP to be published; we denote the number of samples by N, and refer to SPDH $-\text{Sign}_{g,\phi}(N)$ below. The key generation and signing algorithms are given by:

Algorithm 1 KeyGen Algorithm

KeyGen(N):
for $i \leftarrow 1, N$ **do**
  $X_i \xleftarrow{\$} \mathcal{X}_{g,\phi}$
  $s_i \xleftarrow{\$} \mathbb{Z}_n$
  $Y_i \leftarrow s_i \star X_i$
end for
$sk \leftarrow (s_1, ..., s_N)$
$pk \leftarrow ((X_1, \ldots, X_N), (Y_1, \ldots, Y_N))$
return (sk, pk)

Algorithm 2 Signing Algorithm

$\text{Sg}(m, (sk, pk))$:
for $i \leftarrow 1, N$ **do**
  $t_i \xleftarrow{\$} \mathbb{Z}_n$
  $I_i \leftarrow t_i \star X_i$
end for
$I \leftarrow (I_1, \ldots, I_N)$
$c \leftarrow H(I, m)$
for $i \leftarrow 1, N$ **do**
  if $c_i = 0$ **then**
    $p_i \leftarrow t_i$
  else
    $p_i \leftarrow t_i - s_i$
  end if
end for
$p \leftarrow (p_1, \ldots, p_N)$
$(\sigma_1, \sigma_2) \leftarrow (I, p)$
return (σ_1, σ_2)

Note that it suffices to solve SDLP to break the scheme. Before we state the hardness result of SPDH-Sign, we require some definitions. For the syntax of signature schemes, see [3] or [17].

Definition 8. [3, Definition 8] (Chosen Message Attack) Let $\text{S} = (\text{KeyGen}, \text{Sg}, \text{Vf})$ be a signature scheme and $\mathcal{A}$ an adversary. Consider the following game:

1. The challenger obtains $(sk, pk) \leftarrow \text{KeyGen}$ and passes pk to $\mathcal{A}$.
2. The adversary enters into a 'querying' phase, whereby they can obtain signatures $\sigma_i = \text{Sg}(sk, m_i)$ from the challenger, for the adversary's choice of message m_i. The total number of messages queried is denoted Q.
3. The adversary submits their attempted forgery - a message-signature pair (m^*, σ^*) - to the challenger. The challenger outputs $\text{Vf}(pk, (m^*, \sigma^*))$; the adversary wins if this output is 'Accept'.

Denote the advantage of the adversary in this game with S as the challenger by $\text{cma-adv}(\mathcal{A}, \text{S})$.

Suppose an adversary $\mathcal{A}$ is given $(g, \phi) \in \text{Hol}(G)$ and $s_{g,\phi}(x)$ for some $x \in \mathbb{N}$. Denote the advantage of $\mathcal{A}$ against SDLP by $\text{sdlp} - \text{adv}(\mathcal{A}, (g, \phi))$. The authors of [3] prove

Theorem 2. *[3, Theorem 7] Let G be a finite non-abelian group; $(g, \phi) \in G \ltimes Aut(G)$; and $n \in \mathbb{N}$ be the smallest integer such that $s_{g,\phi}(n) = 1$. Consider*

the chosen message attack game in the random oracle model, where Q_s is the number of signing queries made and Q_{ro} is the number of random oracle queries. For any efficient adversary $\mathcal{A}$ and $N \in \mathbb{N}$, there exists an efficient adversary $\mathcal{B}$ running $\mathcal{A}$ as a subroutine such that $\mathrm{SPDH-Sign}_{g,\phi}(N)$ *has*

$$\delta \leq \frac{Q_s}{n}(Q_s + Q_{ro} + 1) + \frac{Q_{ro}}{2^N} + \sqrt{(Q_{ro} + 1)\,\mathrm{sdlp-adv}(\mathcal{B}, (g, \phi))}$$

where $\delta = \text{cma-adv}^{\mathrm{ro}}\left(\mathrm{SPDH-Sign}_{g,\phi}(N), \mathcal{A}\right)$ *is the advantage of the signature scheme in the random oracle model version of the chosen message attack game.*

For the use of SPDH-Sign, one has to pick a particular group with which the scheme will be instantiated; the authors propose the use of the group

$$G = G_p := \left\{ \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix} : a, b \in \mathbb{Z}_{p^2}, a \equiv 1 \mod p \right\}$$

We note that we have $G_p \cong \mathbb{Z}/p^2\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}$, where $\mathbb{Z}/p\mathbb{Z}$ acts on $\mathbb{Z}/p^2\mathbb{Z}$ via $a \star b = b^{1+pa}$. This isomorphism and its inverse are plainly efficiently computable.

When using such a group, p would be chosen to be a cryptographic prime, that is, $p = \exp(\lambda)$ where λ is the security parameter of a SLDP-based scheme, such as SPDH-Sign.

Finally in this section, we note an incorrect statement in [3]. The authors write:

Theorem 3. *[3, Theorem 9] Let $(g, \phi) \in G_p \rtimes \mathrm{Aut}(G_p)$, where p is an odd prime. Suppose n is the smallest integer for which $s_{g,\phi}(n) = 1$. Then*

$$n \in \left\{p, p^2, p^3, p^4, p^5, p^6, (p-1), p(p-1), p^2(p-1), p^3(p-1), p^4(p-1), p^5(p-1)\right\}$$

The reasoning runs as follows. Since $n \mid \mathrm{ord}((g, \phi))$, and $\mathrm{ord}((g, \phi)) \mid G_p \ltimes \mathrm{Aut}(G)$, we must have $n \mid (p-1)p^6$ for some odd prime p, and $n \neq (p-1)p^6$ since this would imply $G_p \rtimes \mathrm{Aut}(G_p)$ were cyclic.

The reasoning is sound; the conclusion of the theorem statement, however, is false when $p \neq 3$: since p is prime, $p - 1$ is not prime, and thus the set of possibilities for n includes all elements of the set of divisors of $p - 1$ multiplied by powers of p, up to p^6 - not just the twelve values stated above. For instance, $n = 2p$ is a possibility for all p. The statement should read:

Theorem 4. *Let $(g, \phi) \in G_p \rtimes \mathrm{Aut}(G_p)$, where p is an odd prime. Suppose n is the smallest integer for which $s_{g,\phi}(n) = 1$. Let $\{p_1, ..., p_t\}$ be a set of prime divisors of $p - 1$. Then*

$$n \in \left\{ p^j \prod_{i \in S} p_i \right\}_{j,S},$$

where $S \subset [t]$ runs over multisets S such that $\prod_{i \in S} p_i$ denotes the products of the p_i indexed by a subset of possible indices such that $\prod_{i \in S} p_i \mid p - 1$, and j satisfies $j \in [5]$ if $S \neq \emptyset$ or $j \in [6]$ if $S = \emptyset$.

We point this out for its relevance to our results in Section 4.

3 On G_p and its Automorphisms

In this setion we discuss properties of G_p which we will exploit below, and in particular give an explicit form for its automorphisms. Any finite group has a set of automorphisms, denoted $\mathrm{Aut}(G)$, which form a group under composition. The structure of $\mathrm{Aut}(G)$ comprises two factors: the inner and outer automorphisms. These each form a subgroup of $\mathrm{Aut}(G)$.

Inner automorphisms are defined by conjugation: if $g \in G$ is an arbitrary group element, the map $c_h : g \mapsto hgh^{-1}$ can be checked to be an automorphism. The group formed by such maps is denoted $\mathrm{Inn}(G)$. Clearly if h commutes with all other group elements, c_h is the trivial map; thus when counting the number of inner automorphisms, we find that there are $|\mathrm{Inn}(G)| = \frac{|G|}{|\mathcal{Z}(G)|}$ of them, where $\mathcal{Z}(G)$ denotes the center of the group.

The group of outer automorphisms, $\mathrm{Out}(G)$, is defined as

$$\mathrm{Out}(G) := \mathrm{Aut}(G)/\mathrm{Inn}(G)$$

Hence there are $|\mathrm{Out}(G)| = |\mathrm{Aut}(G)|/|\mathrm{Inn}(G)|$ outer automorphisms. We are interested in determining explicit forms of elements of these groups, for our subsequent cryptanalysis of SPDH-Sign. Below, we let $g \in G_p$ and write $g = \begin{pmatrix} 1+pm & b \\ 0 & 1 \end{pmatrix}$ for some $m \in \mathbb{Z}/p\mathbb{Z}$ and $b \in \mathbb{Z}/p^2\mathbb{Z}$. As in [9], G_p is generated by elements r, s where

$$r = \begin{pmatrix} 1+p & 0 \\ 0 & 1 \end{pmatrix} \text{ and } s = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

So a generic group element $g = \begin{pmatrix} 1+pm & b \\ 0 & 1 \end{pmatrix}$ may be written $g = s^b r^m$, and group multiplication can be expressed

$$s^b r^m \cdot s^{b'} r^{m'} = s^{b+b'+pmb'} r^{m+m'}$$

Inner Automorphisms of G_p We first consider inner automorphisms. Note that $(s^c r^n)^{-1} = r^{n^{-1}} s^{c^{-1}} = r^{-n} s^{-c} = s^{pcn} s^{-c} r^{-n}$, since $s^{pcn} s^{-c} r^{-n} \cdot s^c r^n = s^{pcn} s^{-pcn} r^0 = 1$. The inner automorphisms act on $s^b r^m$ by conjugation; that is, if $\phi \in \mathrm{Inn}(G_p)$, then

$$\begin{aligned} \phi(s^b r^m) &= (s^c r^n)^{-1} s^b r^m (s^c r^n) = (s^c r^n)^{-1} s^{b+c+pmc} r^{m+n} \\ &= s^{pcn} y^{-c} r^{-n} s^{b+c+pmc} r^{m+n} = s^{pcn} s^{b+pmc-pn(b+c)} r^m \\ &= s^{pnc+b+pmc-pnb-pnc} r^m = s^{b+p(mc-nb)} r^m \end{aligned}$$

We summarise this as

Lemma 1. *Let ϕ be an inner automorphism of G_p. Then the action of ϕ on a generic group element $g = s^b r^m$ is given by*

$$\phi(g) = s^{b+p(mc-nb)} r^m$$

We note that there are $|\operatorname{Inn}(G_p)| = \frac{|G_p|}{|\mathcal{Z}(G_p)|} = \frac{p^3}{p} = p^2$ inner automorphisms, since the center of G_p is

$$\mathcal{Z}(G_p) = \left\{ \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix} : b \in \mathbb{Z}/p^2\mathbb{Z}, b \equiv 0 \bmod p \right\} = \left\langle \begin{pmatrix} 1 & p \\ 0 & 1 \end{pmatrix} \bmod p^2 \right\rangle$$

Outer Automorphisms of G_p The form of the outer automorphisms is less obvious than that of the inner automorphsims; we have

Proposition 1. *The outer automorphisms of G_p are given by the maps*

$$\phi(s^b r^m) = s^{bw+pmu} r^m$$

where ϕ corresponds to a pair $(u, w) \in \mathbb{Z}/p\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}^\times$.

Proof. Clearly $\phi : G_p \to G_p$ such that $\phi(e) = e$. Observe

$$\begin{aligned} \phi(gg') &= \phi(s^b r^m \cdot s^{b'} r^{m'}) = \phi(s^{b+b'+pmb'} r^{m+m'}) \\ &= s^{w(b+b'+pmb')+p(m+m')u} r^{m+m'} \end{aligned}$$

and

$$\begin{aligned} \phi(g)\phi(g') &= s^{bw+pmu} r^m s^{b'w+pm'u} r^{m'} \\ &= s^{bw+pmu+b'w+pm'u+pm(b'w+pm'u)} r^{m+m'} \\ &= s^{bw+pmu+b'w+pm'u+pmb'w} r^{m+m'} \end{aligned}$$

So ϕ is indeed multiplicative. Moreover, these are not inner automorphisms, which be seen by inspecting the 'twist' of b in the exponent by w. Note that there are $|\operatorname{Out}(G_p)| = |\operatorname{Aut}(G_p)|/|\operatorname{Inn}(G_p)| = (p-1)p^3/p^2 = (p-1)p$ outer automorphisms, and since the automorphisms above are obtained by pairs from $\mathbb{Z}/p\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}^\times$, and $|\mathbb{Z}/p\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}^\times| = p(p-1)$, we conclude we have found all the outer automorphisms. □

We conclude this section with the important observation

Corollary 1. *Let $\phi \in \operatorname{Aut}(G_p)$. Then for any $g = s^b r^m$, we have $\phi(g) = s^{b'} r^m$; that is, ϕ leaves r^m unchanged.*

Proof. Observation of the results of Lemma 1 and Proposition 1. □

4 Making 'Mash' when $n \leq \operatorname{poly}(\log p)p$

Here we outline an attack on SPDH when n is 'small' (though still exponential in the security parameter). The attack uses the structure of G_p to extract information on x from $g, \phi, s_{g,\phi}(x)$. We begin with a proposition:

Proposition 2. *Let $G = M \rtimes N$ be a semidirect product of finite groups with N acting on M via automorphisms. Consider the holomorph of G, $(M \rtimes N) \rtimes \mathrm{Aut}(G)$. Then if N is simple, the maps induced on N by elements of $\mathrm{Aut}(G)$ are either the constant map $N \to \{e\}$ or automorphisms.*

Proof. Let $\phi \in \mathrm{Aut}(G)$. Writing $\phi(m, n) = (m', n')$, consider the induced map $\psi : N \to N, n \mapsto n'$. Since

$$\phi((m,n))\phi((m',n')) = \phi((m,n)(m',n')) = \phi((n'(m)m', nn'))$$

we have $\psi(n)\phi'(n') = \psi(nn')$. Moreover,

$$\phi((m,e))\phi((m',e)) = \phi((m,e)(m',e)) = \phi(mm', e)$$

so $\psi(e)\psi(e) = \psi(e)^2 = \psi(e)$ and $\psi(e)$ is an idempotent in a finite group, hence $\psi(e) = e$. Thus ψ is an endomorphism of N.

Since the image of a group under an endomorphism is a subgroup, we find that either $\psi(N) = N$ or $\psi(N) = \{e\}$. In the latter case every element is mapped to e, and in the former we have a homomorphism between finite groups of trivial kernel, and thus an automorphism. □

We note that when $N = \mathbb{Z}/p\mathbb{Z}$, $\mathrm{End}(\mathbb{Z}/p\mathbb{Z}) = \mathbb{Z}/p\mathbb{Z}$.

We now give a general method to recover x when n is at most a small multiple of $|\mathrm{Aut}(N)|$, subject to a constraint on the group element $(g, \phi) \in G \rtimes \mathrm{Aut}(G)$ where $G = M \rtimes N$ is a semidirect product with M and N finite abelian, and N simple as in the previous proposition, and $g = (a, \varphi) \in G$. We then specialise to the particular case of G_p.

Theorem 5. *Let $G = M \rtimes N$ be a semidirect product with M and N finite abelian, and N simple. Suppose $|\mathrm{Aut}(N)| = \prod_i p_i$ for distinct primes p_i. Let $(g, \phi) \in G \rtimes \mathrm{Aut}(G)$, where $g = (a, \varphi) \in G$. Suppose that ϕ acts on φ as an automorphism ψ, sending $\varphi \mapsto \varphi^\alpha$ for some $\alpha \neq 0$. Then there is a quantum polynomial time algorithm to find $x \bmod |\mathrm{Aut}(N)|$.*

Proof. The SDLP instance is to recover x from $s_{g,\phi}(g)$, which we may write $s_{g,\phi}(x) = \phi^{x-1}((a,\varphi))\phi^{x-2}((a,\varphi))...\phi(a,\varphi)(a,\varphi)$ where $g = (a, \varphi) \in M \rtimes N$. If ϕ acts as an induced automorphism ψ on φ sending φ to φ^α for some α, since (g, ϕ) is public evaluating $\phi(g)$ for φ^α and appealing to an abelian discrete logarithm oracle yields α. We can write $s_{g,\phi}(x)$ as

$$\phi^{x-1}((a,\varphi))\phi^{x-2}((a,\varphi))...\phi((a,\varphi))(a,\varphi) = (\cdot, \psi^{x-1}(\varphi)\psi^{x-2}(\varphi)...\psi(\varphi)\varphi)$$

for some unspecified first entry. The second entry above can be rewritten

$$(\varphi^{\alpha^{x-1}})(\varphi^{\alpha^{x-2}})...(\varphi^\alpha)\varphi = \varphi^{\alpha^{x-1}+\alpha^{x-2}+...+\alpha+1}$$

Another appeal to an abelian discrete logarithm oracle obtains the exponent $\alpha^{x-1} + \alpha^{x-2} + ... + \alpha + 1 \bmod |\mathrm{Aut}(N)|$. We now split into two cases: if $\alpha = 1$,

then $\alpha^{x-1} + \alpha^{x-2} + ... + \alpha + 1 = x \bmod |\operatorname{Aut}(N)|$ and we are done. So suppose we are in the case of $\alpha \neq 1$.

By the CRT, it suffices to recover $x \bmod p_i$ from

$$b := \alpha^{x-1} + \alpha^{x-2} + ... + \alpha + 1 \bmod p_i$$

for all prime factors p_i of $|\operatorname{Aut}(N)|$ (which can be found efficiently with a quantum algorithm). To do this, rewrite

$$b = \alpha^{x-1} + \alpha^{x-2} + ... + \alpha + 1 = \frac{\alpha^x - 1}{\alpha - 1} \bmod p_i$$

and rearrange for

$$\alpha^x = b(\alpha - 1) + 1 \bmod p_i,$$

which can be done since we assumed $\alpha \neq 1$. A third appeal to an abelian discrete logarithm oracle gives $x \bmod p_i$, and hence $x \bmod |\operatorname{Aut}(N)|$. □

Corollary 2. *Let $n = \operatorname{poly}(\log p)p$ and $(g, \phi) \in G_p \rtimes \operatorname{Aut}(G_p)$. Then there is a quantum polynomial time algorithm to solve $SDLP_{G_p,g,\phi,n}$.*

Proof. We apply the theorem with $M = \mathbb{Z}/p^2\mathbb{Z}$ and $N = \mathbb{Z}/p\mathbb{Z}$, and note that by Corollary 1 any automorphism leaves the r component of a group element fixed, and so in the notation of the theorem, we always have $\alpha = 1$. We then obtain $x \mod |\operatorname{Aut}(N)| = x \mod p - 1$ as in the proof of the theorem. From this we can obtain $x \bmod p$. If $n = \operatorname{poly}(\log p)p$ we can then find the true value of x by exhaustion in polynomial time. □

We note that such values for n are possible by Theorem 4.

The consequence of all this is that when instantiating SPDH-Sign with $G = G_p$, one should choose n to be at least $n \approx p^2$.

5 An Attack in the Style of [7]

In [7], the scheme 'MAKE' [19] was cryptanalysed, and [18] extended the attack to the scheme 'MOBS' [20]. The scheme uses square matrices whose entries are bitstrings of k bits equipped with the logical operations of OR and AND. The authors of [7] found (in the notation of [6]) that, given such a matrix M and an automorphism h of the space of such matrices, and writing $A := h^{x-1}(M)...h(M)M$, one could obtain $h(A)M = h^x(M)A$. From this it was argued that MAKE and MOBS were insecure, since by linear algebra $h^x(M)$, and then h^x and finally x, could be computed (though the efficacy of the attack was disputed in [6]).

We note that one can obtain $\phi^x(g)$ given g, ϕ and $s_{g,\phi}(x)$, by computing

$$\phi^x(g) = \phi(s_{g,\phi}(x))g \cdot s_{g,\phi}(x)^{-1},$$

somewhat in the style of the attacks on MAKE and MOBS. It was known prior to this work that this element could be computed. Here, however, we observe

that since we know g, one can then obtain further information on x.

In more detail and for $G = G_p$, suppose we have $g \in G_p$. Write $g = \begin{pmatrix} 1+pa & b \\ 0 & 1 \end{pmatrix}$ for some $a \in \mathbb{Z}/p\mathbb{Z}$ and $b \in \mathbb{Z}/p^2\mathbb{Z}$. We then compute $\phi^x(g) = \begin{pmatrix} 1+pa' & b' \\ 0 & 1 \end{pmatrix}$ for some $a' \in \mathbb{Z}/p\mathbb{Z}$ and $b' \in \mathbb{Z}/p^2\mathbb{Z}$. Here we will consider the case of the inner automorphisms $\mathrm{Inn}(G_p)$, and of elements in $\mathrm{Out}(G_p) := \mathrm{Aut}(G_p)/\mathrm{Inn}(G_p)$.

First consider inner automorphisms. Recall that the inner automorphisms act on $s^b r^m$ by conjugation, and that by Lemma 1 if $\phi \in \mathrm{Inn}(G_p)$, then

$$\phi(s^b r^m) = s^{b+p(mc-nb)} r^m$$

We then compute

$$\phi^x(s^b r^m) = s^{b+xp(mc-nb)} r^m$$

We can multiply by r^{-m} to obtain $s^{b+xp(mc-nb)}$, use a discrete logarithm oracle to find $b + xp(mc - nb)$, and then use linear algebra to find $x \bmod p$.

In the case of outer automorphisms, we found in Proposition 1 that these are given by the maps

$$\phi(s^b r^m) = s^{bw+pmu} r^m$$

where ϕ corresponds to a pair $(u, w) \in \mathbb{Z}/p\mathbb{Z} \rtimes \mathbb{Z}/p\mathbb{Z}^\times$. We then compute

$$\phi^x(g) = s^{bw^x+pmu(w^{x-1}+\ldots+w+1)} r^m$$

We can cancel the r^m, since it is public, for

$$s^{bw^x+pmu(w^{x-1}+\ldots+w+1)}$$

and can hence recover $bw^x + pmu(w^{x-1} + \ldots + w + 1)$ by solving the discrete logarithm problem instance. If we compute

$$\begin{aligned} s^{bw^x+pmu(w^{x-1}+\ldots+w+1)} \cdot s^{-bw^{x-1}-pmu(w^{x-2}+\ldots+w+1)} \\ = s^{bw^x-bw^{-1}+pmu(w^{x-1}+\ldots+w+1)-pmu(w^{x-2}+\ldots+w+1)} \\ = s^{bw^{x-1}(w-1)+pmuw^{x-1}}, \end{aligned}$$

we can then obtain $bw^{x-1}(w-1) + pmuw^{x-1} = w^{x-1}(b(w-1) + pmu)$, and if $w \neq 1$ we can cancel the righthand factor for w^{x-1}, and recover $x - 1 \bmod p$ from a discrete logarithm oracle.

Finally, we note all automorphisms are obtained from composing inner and outer automorphisms.

6 SPDH and the Linear Hidden Shift Problem

In this section we show that Theorem 5 implies a solution to the linear hidden shift (LHS) problem as defined in [1]. We begin by defining this problem formally. Let $\langle \mathbf{g}_i, \mathbf{s} \rangle := \prod_j g_{i_j}^{s_j}$, where $g_{i_j} \in G$ and G is written multiplicatively.

Definition 9. The search LHS problem $\mathrm{LHS}_{G,X,\mathbf{s}}$ is hard over a regular group action $(G, X, \star)$ if for any $m = \mathrm{poly}(\lambda)$, $\mathbf{s} \leftarrow \{0,1\}^n$, and for any PPT attacker $\mathcal{A}$, we have

$$\Pr\left[\mathcal{A}\left(\{(x_i, \mathbf{g}_i, (\langle \mathbf{g}_i, \mathbf{s}\rangle) \star x_i)\}_{i\in[m]}\right) \text{ outputs } \mathbf{s}\right] \leq \mathrm{negl}(\lambda),$$

where $\mathbf{g}_i \leftarrow G^n$ and $x_i \leftarrow X$ are sampled independently, over all random coins in the experiment.

For SDPGA: the search LHS problem is hard over $(\mathbb{Z}/n\mathbb{Z}, G_p \rtimes \mathrm{Aut}(G_p), \star)$ if for any $m = \mathrm{poly}(\lambda)$ and for any PPT attacker $\mathcal{A}$, we have

$$\Pr\left[\mathcal{A}\left(\{((g_i, \phi_i), \mathbf{x}_i, (\langle \mathbf{x}_i, \mathbf{s}\rangle) \star (g_i, \phi_i))\}_{i\in[m]}\right) \text{ outputs } \mathbf{s}\right] \leq \mathrm{negl}(\lambda),$$

where $\mathbf{x}_i \leftarrow (\mathbb{Z}/n\mathbb{Z})^n, \mathbf{s} \leftarrow \{0,1\}^n, (g_i, \phi_i) \leftarrow G_p \rtimes \mathrm{Aut}(G_p)$ sampled independently, over all random coins in the experiment. Note that additively

$$(\langle \mathbf{x}_i, \mathbf{s}\rangle) \star (g_i, \phi_i) = (\sum_j s_j x_{i_j}) \star (g_i, \phi_i) = s_{g_i,\phi_i}\left(\sum_j s_j x_{i_j}\right)$$

We now prove our result:

Theorem 6. *Let $(g, \phi) \in G_p \rtimes \mathrm{Aut}(G_p)$, where $g = (a, \varphi) \in G_p$. Let $m \geq n$. Then there is a quantum polynomial time algorithm to solve $LHS_{G,X,\mathbf{s}}$.*

Proof. Write $x'_i = \sum_j s_j x_{i_j}$. We are given the (g_i, ϕ_i), $\mathbf{x}_i$, and $s_{g_i,\phi_i}(x'_i)$. We therefore use the method of Theorem 5 to find $b_i := x'_i \bmod p-1$, for $i = 1, ..., m$. This gives us the m equations

$$\begin{aligned} b_1 &= \sum_j s_j x_{1_j} \bmod p-1 \\ &\vdots \\ b_m &= \sum_j s_j x_{m_j} \bmod p-1 \end{aligned}$$

This is m equations in the n unknown values of $s_1, ..., s_n$ with known coefficients $x_{i_j} \bmod p-1$. Since $s_i \in \{0,1\}$ the modulo operation leaves s_i unchanged. Thus when $m \geq n$ we can solve this system of equations for the s_i, and so solve the search LHS instance. □

7 On the Equivalence of SCDH and SDLP

Here we reduce SDLP to the semidirect computational Diffie-Hellman (SCDH) problem via an efficient quantum algorithm. Since SCDH reduces to SDLP trivially, this establishes the quantum polynomial equivalence of the two problems, stated as an open problem in [3]. We do this by transforming SDLP instances into hidden subgroup problem (HSP) instances, assuming the presence of a SCDH oracle. Recall:

Definition 10. (Hidden subgroup problem) Let $f : G \to S$ be a function from finite group G to a set S that is constant on the cosets of some $H \leq G$; i.e. $f(g) = f(g')$ if and only if $gH = g'H$. Given f, G, S, find a generating set of H.

We refer below to $\mathrm{SCDH}^2_{g,\phi,n}$, which is the general SCDH problem restricted to the task of doubling in the argument of $s_{g,\phi}(x)$; that is, one solves $\mathrm{SCDH}^2_{g,\phi,n}$ if given g, ϕ, and $s_{g,\phi}(x)$, one computes $s_{g,\phi}(x+x) = s_{g,\phi}(2x)$. Note that this is weaker than a general SCDH oracle which returns $s_{g,\phi}(a+b)$ given $s_{g,\phi}(a)$ and $s_{g,\phi}(b)$ for any $a, b \in \mathbb{Z}/n\mathbb{Z}$.

Theorem 7. *There is a quantum polynomial-time reduction from $SDLP_{g,\phi,n,x}$ to $SCDH^2_{g,\phi,n}$.*

Proof. Let $x \in \mathbb{Z}/n\mathbb{Z}$, $(g, \phi) \in G \rtimes \mathrm{Aut}(G)$, and suppose we are given $s_{g,\phi}(x)$. We assume that given (g, ϕ), $s_{g,\phi}(x)$, and $s_{g,\phi}(y)$, we are able to compute $s_{g,\phi}(x+y)$ in the case $x = y$. In particular, we can then compute $s_{g,\phi}(ax)$ for any a in (classical) polynomial time by computing $s_{g,\phi}(2x) = s_{g,\phi}(x+x)$, writing a in base 2, and then repeatedly doubling and adding in the argument of $s_{g,\phi}(\cdot)$ appropriately.

We then define a map $f : \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z} \to \mathcal{X}_{g,\phi}$, $(a, b) \mapsto \phi^b(s_{g,\phi}(ax))s_{g,\phi}(b)$. This can be rewritten $f(a, b) = s_{g,\phi}(ax + b)$. Observe that if $f(a, b) = f(a', b')$, then we must have $ax + b = a'x + b' \bmod n$, since the group action of $\mathbb{Z}/n\mathbb{Z}$ on $G \rtimes \mathrm{Aut}(G)$ is regular. We then find that $f(a, b) = f(a', b')$ if and only if $(a, b) = (a', b') + \lambda(1, -x)$. This is an HSP instance, which can be solved in quantum polynomial time via Shor. □

8 Relation of SDLP to the Hidden Subgroup Problem

In this final section we explain why we could not solve the SDLP problem via a reduction to a hidden subgroup problem instance in an analogous manner to the abelian discrete logarithm problem (DLP).

DLP is reduced to HSP via the map $f(a, b) = s^a g^b$ where $g^x = s$, with $a, b \in \mathbb{Z}/n\mathbb{Z}$. Then $f(a, b) = g^{ax+b}$, and $f(a, b) = f(a', b')$ iff $(a, b) = (a', b') + \lambda(1, -x)$.

In that spirit, one might try setting $f(a, b, c) = (s_{g,\phi}(x), \phi^a)^c(g, \phi)^b$. Then if $a = x$, we have $f(a, b, c) = (g, \phi)^{cx+b}$ and we would have defined a map from an abelian group into the cyclic group $\langle(g, \phi)\rangle$, as is done for DLP. The condition $a = x$ seems problematic, however. Note $f(a, b, c) = f(a', b', c')$ if $(a, b, c) = (x, b', c') + \lambda(0, -x, 1)$, as (some) solutions have the form $(x, 0, 0) + \langle(0, -x, 1)\rangle$, which is an affine line in $(\mathbb{Z}/n\mathbb{Z})^3$. This however is not a 'period' in the sense of Shor that Shor's algorithm for the HSP requires. Thus an obstacle for defining the required map is the 'hiding' of ϕ^x, which prevents an adversary for defining a map into $\langle(g, \phi)\rangle$.

One might observe that we are not given a group element, but merely an element of the orbit $\mathcal{X}_{g,\phi}$ of (g, ϕ) under the action of $\mathbb{Z}/n\mathbb{Z}$. This might prompt one to attempt to define a map $f : \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z} \to \mathcal{X}_{g,\phi}$ in the spirit of the above map. This would seek to define a map $f(a, b) = s_{g,\phi}(ax+b)$. Then since the

group action is regular, $f(a,b) = f(a',b')$ if and only if $(a,b) = (a',b') + \lambda(1,-x)$ and we could use Shor's period finding algorithm. Since we can add b in the argument, to define such a map one would first have to define a map $f'(a) = s_{g,\phi}(ax)$. Referring to the previous section, one can see that this is in fact how Theorem 7 was proved, since the possibility of defining such a map follows from assuming SCDH. However, it seems that without the SCDH assumption, one cannot compute $s_{g,\phi}(ax)$ given the available information. This thus can be seen as an obstacle to a complete quantum solution to SDLP.

References

[1] N. Alamati, L. De Feo, H. Montgomery, and S. Patranabis. "Cryptographic Group Actions and Applications". In: *ASIACRYPT 2020*. Ed. by S. Moriai and H. Wang. Springer International Publishing, 2020, pp. 411–439.

[2] C. Battarbee, D. Kahrobaei, L. Perret, and S. F. Shahandashti. *A Subexponential Quantum Algorithm for the Semidirect Discrete Logarithm Problem*. Cryptology ePrint Archive, Paper 2022/1165. Presented at NIST's Fourth PQC Standardization Conference. 2022. URL: https://eprint.iacr.org/2022/1165.

[3] C. Battarbee, D. Kahrobaei, L. Perret, and S. F. Shahandashti. "SPDH-Sign: Towards Efficient, Post-quantum Group-Based Signatures". In: *Post-Quantum Cryptography*. Ed. by T. Johansson and D. Smith-Tone. Springer Nature Switzerland, 2023, pp. 113–138. ISBN: 978-3-031-40003-2.

[4] C. Battarbee, D. Kahrobaei, and S. F. Shahandashti. "Cryptanalysis of Semidirect Product Key Exchange Using Matrices Over Non-Commutative Rings". In: *Mathematical Cryptology* 1.2 (Mar. 2022), 2–9. URL: https://journals.flvc.org/mathcryptology/article/view/130528.

[5] C. Battarbee, D. Kahrobaei, and S. F. Shahandashti. *Semidirect Product Key Exchange: the State of Play*. Cryptology ePrint Archive, Paper 2023/594. 2023. URL: https://eprint.iacr.org/2023/594.

[6] C. Battarbee, D. Kahrobaei, D. T., and S. F. Shahandashti. "On the efficiency of a general attack against the MOBS cryptosystem". In: *Journal of Mathematical Cryptology* 16.1 (2022), pp. 289–297. DOI: doi:10.1515/jmc-2021-0050.

[7] D. R. L. Brown, N. Koblitz, and J. T. LeGrow. "Cryptanalysis of "MAKE"". In: *Journal of Mathematical Cryptology* 16.1 (2022), pp. 98–102. DOI: doi:10.1515/jmc-2021-0016.

[8] W. Castryck and N. V. Meeren. *Two remarks on the vectorization problem*. Cryptology ePrint Archive, Paper 2022/1366. 2022. URL: https://eprint.iacr.org/2022/1366.

[9] K. Conrad. *GROUPS OF ORDER p^3*. URL: https://kconrad.math.uconn.edu/blurbs/grouptheory/groupsp3.pdf.

[10] J.-M. Couveignes. *Hard Homogeneous Spaces*. Cryptology ePrint Archive, Paper 2006/291. 2006. URL: https://eprint.iacr.org/2006/291.

[11] G. D'Alconzo and A. J. Di Scala. *Representations of Group Actions and their Applications in Cryptography*. Cryptology ePrint Archive, Paper 2023/1247. 2023. URL: https://eprint.iacr.org/2023/1247.

[12] W. Diffie and M. Hellman. "New directions in cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[13] O. W. Gnilke and J. Zumbrägel. *Cryptographic Group and Semigroup Actions*. Cryptology ePrint Archive, Paper 2023/017. 2023. URL: https://eprint.iacr.org/2023/017.

[14] M. Habeeb, D. Kahrobaei, C. Koupparis, and V. Shpilrain. "Public Key Exchange Using Semidirect Product of (Semi)Groups". In: *Applied Cryptography and Network Security*. Ed. by M. Jacobson, M. Locasto, P. Mohassel, and R. Safavi-Naini. Springer Berlin Heidelberg, 2013, pp. 475–486. ISBN: 978-3-642-38980-1.

[15] M. Imran and G. Ivanyos. *Efficient quantum algorithms for some instances of the semidirect discrete logarithm problem*. Cryptology ePrint Archive, Paper 2023/1953. 2023. URL: https://eprint.iacr.org/2023/1953.

[16] D. Kahrobaei and V. Shpilrain. "Using Semidirect Product of (Semi)groups in Public Key Cryptography". In: *Pursuit of the Universal*. Ed. by A. Beckmann, L. Bienvenu, and N. Jonoska. Springer International Publishing, 2016, pp. 132–141. ISBN: 978-3-319-40189-8.

[17] J. Katz and Y. Lindell. *Introduction to Modern Cryptography, Second Edition*. Chapman & Hall/CRC Cryptography and Network Security Series. Taylor & Francis, 2014. ISBN: 9781466570269.

[18] C. Monico. *Remarks on MOBS and cryptosystems using semidirect products*. Cryptology ePrint Archive, Paper 2021/1114. 2021. URL: https://eprint.iacr.org/2021/1114.

[19] N. Rahman and V. Shpilrain. "MAKE: A matrix action key exchange". In: *Journal of Mathematical Cryptology* 16.1 (2022), pp. 64–72. DOI: doi:10.1515/jmc-2020-0053.

[20] N. Rahman and V. Shpilrain. *MOBS (Matrices Over Bit Strings) public key exchange*. Cryptology ePrint Archive, Paper 2021/560. 2021. URL: https://eprint.iacr.org/2021/560.

[21] V. Roman'kov. "Linear decomposition attack on public key exchange protocols using semidirect products of (semi)groups". In: *CoRR* abs/1501.01152 (2015). URL: http://arxiv.org/abs/1501.01152.

[22] P. W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134.