New approaches to group-based post-quantum cryptography

Dorota Wedmann

25 June 2025

Warsaw University of Technology

Advances in Group Theory and Applications 2025

Few words about post-quantum cryptography

Post-quantum cryptography is implemented on classical machines and is supposed to be safe against quantum attacks (mainly attacks using Shor's and Grover's algorithms). It is a current and urgent topic.

Group-based post-quantum cryptography is extremely versatile – we can build a lot of cryptographic schemes with different functionalities using group problems.

Few words about post-quantum cryptography

Post-quantum cryptography is implemented on classical machines and is supposed to be safe against quantum attacks (mainly attacks using Shor's and Grover's algorithms). It is a current and urgent topic.

Group-based post-quantum cryptography is extremely versatile – we can build a lot of cryptographic schemes with different functionalities using group problems.

Group problems used in cryptography that can be restricted to subsets

- Conjugacy search problem
- Subgroup membership problem
- Factorization problem
- Subgroup intersection problem
- Decomposition search problem

Carvalho and Malheiro proposed in [CM25] using **orbits through automorphism** as subsets. The orbit of g through automorphism ϕ is the set $Orb_\phi(g) = \{\phi^k(g) \mid k \in \mathbb{N}\}$.

Conjugacy search problem restricted to orbits

For given $\phi \in Aut(G)$ and elements $x, y, s \in G$, decide whether x and y are conjugate by an element in the orbit of s through ϕ, i.e., whether there exists $k \in \mathbb{N}$ such that $\phi^k(s^{-1})x\phi^k(s) = y$?

Identification scheme

In an identification scheme between a prover Alice and a verifier Bob, the goal is for Alice to prove to Bob that she knows some secret without revealing information about the secret. We have a **public key** that is known to Alice, Bob, and all potential adversaries, and Alice's **private key** that is known only to Alice. One instance of the identification scheme goes as follows:

1. Alice chooses at random some element r as her **commitment** and sends it to Bob.
2. Bob chooses a random bit as his **challenge** and sends it to Alice.
3. Alice generates a **response** to the challenge that depends on her private key, commitment, and Bob's challenge. Alice sends the response to Bob.
4. Bob **verifies** the response using the public key, his challenge, and the response from Alice.

Identification scheme using conjugacy problem restricted to orbits

Alice's **public key** is an automorphism ϕ of the group G and three elements $x, y, s \in G$ such that $y = \phi^{k+n}(s^{-1})x\phi^{k+n}(s)$, where integers k, n are her **private key**.

1. Alice chooses $j \in \mathbb{N}$ such that $n > j$ and $r \in G$, then sends automorphism ϕ^{k+j} and element $\phi^{n+k}(r)$ to Bob.
2. Bob chooses a random bit c and sends it to Alice.
   - If $c = 0$, then Alice sends element $\phi^{n-j}(s)$ to Bob and Bob checks if the equality $y = \phi^{k+j}(\phi^{n-j}(s^{-1}))x\phi^{k+j}(\phi^{n-j}(s))$ is satisfied. If it is, then Bob accepts the authentication.
   - If $c = 1$, then Alice sends element $\phi^{n-j}(sr)$ to Bob and Bob checks if the equality $\phi^{k+n}(r^{-1})y\phi^{k+n}(r) = \phi^{k+j}(\phi^{n-j}(r^{-1}s^{-1}))x\phi^{k+j}(\phi^{n-j}(sr))$ is satisfied. If it is, then Bob accepts the authentication.

Identification scheme works because:

- If $c = 0$, then Bob gets element $\phi^{n-j}(s)$ and calculates $\phi^{k+j}(\phi^{n-j}(s^{-1}))x\phi^{k+j}(\phi^{n-j}(s)) = \phi^{k+n}(s^{-1})x\phi^{k+n}(s) = y$.
- If $c = 1$, then Bob gets element $\phi^{n-j}(sr)$ and calculates $\phi^{k+j}(\phi^{n-j}(r^{-1}s^{-1}))y\phi^{k+j}(\phi^{n-j}(sr)) = \phi^{n+k}(r^{-1}s^{-1})x\phi^{n+k}(sr) = \phi^{n+k}(r^{-1})\phi^{n+k}(s^{-1})x\phi^{n+k}(s)\phi^{n+k}(r) = \phi^{n+k}(r^{-1})y\phi^{n+k}(r)$.

Rational and algebraic subsets of a group

Let $G = \langle A \mid R \rangle$ be a finitely generated group, A a finite set of generators, $\hat{A} = A \cup A^{-1}$, and $\pi : \hat{A}^* \to G$ the canonical "onto" homomorphism.

Subset $K \subseteq G$ is **rational** if there exists a regular language $L \subseteq \hat{A}^*$ such that $\pi(L) = K$. Rational subsets of the group G will be denoted by $Rat(G)$.

Subset $K \subseteq G$ is **algebraic** if there exists a context-free language $L \subseteq \hat{A}^*$ such that $\pi(L) = K$. Algebraic subsets of the group G will be denoted by $Alg(G)$.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

Subsets instead of subgroups

Pros:

- Subsets have less algebraic structure to exploit in attacks than subgroups.
- Proving the computational hardness of group problems restricted to rational or algebraic subsets can be easier than in the case of subgroups because we can use known computationally hard problems from formal language theory:
  - Rational subset membership for Baumslag-Solitar groups $BS(1, q)$ with $q \geqslant 2$ is decidable and PSPACE-complete [CCZ20],
  - Construction of an automaton group with a PSPACE-complete word problem [WW23].

Cons:

- Not every cryptographic scheme that uses subgroups can be modified by substituting subgroups with subsets.
- Depending on the concrete scheme, using subsets instead of subgroups can be infeasible.

References

Michaël Cadilhac, Dmitry Chistikov, and Georg Zetzsche.
Rational subsets of baumslag-solitar groups.
arXiv preprint arXiv:2006.11898, 2020.

André Carvalho and António Malheiro.
Subsets of groups in public-key cryptography.
Advances in Mathematics of Communications, 19(3):980–995, 2025.

Jan Philipp Wächter and Armin Weiß.
An automaton group with pspace-complete word problem.
Theory of Computing Systems, 67(1):178–218, 2023.