THE CONJUGACY SEARCH PROBLEM IN PUBLIC KEY CRYPTOGRAPHY: UNNECESSARY AND INSUFFICIENT

VLADIMIR SHPILRAIN AND ALEXANDER USHAKOV

Abstract. The conjugacy search problem in a group G is the problem of recovering an $x \in G$ from given $g \in G$ and $h = x^{-1}gx$. This problem is in the core of several recently suggested public key exchange protocols, most notably the one due to Anshel, Anshel, and Goldfeld, and the one due to Ko, Lee at al.

In this note, we make two observations that seem to have eluded most people's attention. The first observation is that solving the conjugacy search problem is not necessary for an adversary to get the common secret key in the Ko-Lee protocol. It is sufficient to solve an apparently easier problem of finding $x, y \in G$ such that $h = ygx$ for given $g, h \in G$.

Another observation is that solving the conjugacy search problem is not sufficient for an adversary to get the common secret key in the Anshel-Anshel-Goldfeld protocol.

Keywords: public key exchange, conjugacy search problem, decomposition problem

1. Introduction

One of the possible generalizations of the *discrete logarithm problem* to arbitrary groups is the so-called *conjugacy search problem* (CSP): given two elements g, h of a group G and the information that $g^x = h$ for some $x \in G$, find at least one particular element x like that. Here g^x stands for $x^{-1}gx$. The (alleged) computational difficulty of this problem in some particular groups (namely, in braid groups) has been used in several group based public key protocols, most notably in [1] and [8].

In this note, we show that solving the conjugacy search problem is unnecessary for an adversary to get the common secret key in the Ko-Lee (or any similar) protocol. This was mentioned, in passing, in the paper [8], but the significance of this observation was downplayed there. On the other hand, we show that solving the conjugacy search problem (or even the simultaneous conjugacy search problem) is insufficient to get the common secret key in the more sophisticated Anshel-Anshel-Goldfeld protocol because an adversary would also have to solve either the membership search problem or the membership decision problem for a subgroup of the platform group. This raises the

Research of the first author was partially supported by the NSF grant DMS-0405105. Research of the second author was partially supported by Umbanet Inc. through an award from the U.S. Department of Commerce NIST, Advanced Technology Program, Cooperative Agreement No. 70NANB2H3012.

stock of the latter protocol and makes one think there might be more to it than meets the eye.

2. Why solving CSP is unnecessary

First we recall the (generalized) Ko-Lee protocol. A group G (with efficiently solvable word problem) and two commuting subsets $A, B \subseteq G$ (i.e., $ab = ba$ for any $a \in A,\ b \in B$) are public. An element $w \in G$ is public, too.

(1) Alice selects a private $a \in A$ and sends the element $a^{-1}wa$ to Bob.
(2) Bob selects a private $b \in B$ and sends the element $b^{-1}wb$ to Alice.
(3) Alice computes $K_A = a^{-1}b^{-1}wba$, and Bob computes $K_B = b^{-1}a^{-1}wab$. Since $ab = ba$ (and therefore, $a^{-1}b^{-1} = b^{-1}a^{-1}$) in G, one has $K_A = K_B = K$ (as an element of G), which is now Alice's and Bob's common secret key.

Note that since we want the key space to be as big as possible, we may assume, to simplify the language in what follows, that, say, the set A is maximal with the property $ab = ba$ for any $a \in A,\ b \in B$.

Now suppose an adversary finds a_1, a_2 such that $a_1wa_2 = a^{-1}wa$ and b_1, b_2 such that $b_1wb_2 = b^{-1}wb$. Suppose also that both a_1, a_2 commute with any $b \in B$. Then the adversary gets

$$a_1b_1wb_2a_2 = a_1b^{-1}wba_2 = b^{-1}a_1wa_2b = b^{-1}a^{-1}wab = K.$$

We emphasize that these a_1, a_2 and b_1, b_2 do not have to do anything with the private elements originally selected by Alice or Bob, which simplifies the search substantially. We also point out that, in fact, it is sufficient for the adversary to find just one pair, say, $a_1, a_2 \in A$, to get the common key:

$$a_1(b^{-1}wb)a_2 = b^{-1}a_1wa_2b = b^{-1}a^{-1}wab = K.$$

In summary, to get the secret key K, the adversary does not have to solve the conjugacy search problem, but instead, it is sufficient to solve an apparently easier problem which some authors (see e.g. [2, 8]) call the *decomposition problem*:

Given two elements w and w' of a group G, find any elements x and y that would belong to a given subset $A \subseteq G$ and satisfy $x \cdot w \cdot y = w'$.

Obviously, *some* x and y satisfying the latter equality always exist (e.g. $x = 1,\ y = w^{-1}w'$), so the point is to have them satisfy the condition $x, y \in A$. We note that this condition may not be easy to verify for some subsets A; the corresponding problem is known as the membership (decision) problem. We address a ramification of this problem, which we call the *membership search problem*, in the next Section 3; here we point out that even in the particular situation considered in [8], it is not immediately clear how to solve the membership decision problem. The authors of [8] do not address this problem; instead they mention, in justice, that if one uses a "brute force" attack by simply going over elements of A one at a time, the above condition will be satisfied automatically. This however may not be the case with other, more practical, attacks.

We also note that the conjugacy search problem is a special case of the decomposition problem where w' is conjugate to w and $x = y^{-1}$. The claim that the decomposition problem should be easier than the conjugacy search problem is intuitively clear since it is generally easier to solve an equation with two unknowns than a special case of the same equation with just one unknown.

3. Why solving CSP is insufficient

The protocol that we describe below, due to Anshel, Anshel, and Goldfeld [1], is more complex than the protocol in the previous section, but it is more general in the sense that there are no requirements on the group G other than to have efficiently solvable word problem. This really makes a difference and gives a big advantage to the protocol of [1] over that of [8].

A group G and elements $a_1, ..., a_k, b_1, ..., b_m \in G$ are public.

(1) Alice picks a private $x \in G$ as a word in $a_1, ..., a_k$ (i.e., $x = x(a_1, ..., a_k)$) and sends $b_1^x, ..., b_m^x$ to Bob.

(2) Bob picks a private $y \in G$ as a word in $b_1, ..., b_m$ and sends $a_1^y, ..., a_k^y$ to Alice.

(3) Alice computes $x(a_1^y, ..., a_k^y) = x^y = y^{-1}xy$, and Bob computes $y(b_1^x, ..., b_m^x) = y^x = x^{-1}yx$. Alice and Bob then come up with a common private key $K = x^{-1}y^{-1}xy$ (called the *commutator* of x and y) as follows: Alice multiplies $y^{-1}xy$ by x^{-1} on the left, while Bob multiplies $x^{-1}yx$ by y^{-1} on the left, and then takes the inverse of the whole thing: $(y^{-1}x^{-1}yx)^{-1} = x^{-1}y^{-1}xy$.

It appears to be a common belief (see e.g. [6, 7]) that solving the (simultaneous) conjugacy search problem for $b_1^x, ..., b_m^x; a_1^y, ..., a_k^y$ in the group G would allow an adversary to get the secret key K. However, if we look at Step (3) of the protocol, we see that the adversary would have to know either x or y not simply as a word in the generators of the group G, but as a word in $a_1, ..., a_k$ (respectively, as a word in $b_1, ..., b_m$); otherwise, he would not be able to compose, say, x^y out of $a_1^y, ..., a_k^y$. That means the adversary would also have to solve the *membership search problem*:

Given elements $x, a_1, ..., a_k$ of a group G, find an expression (if it exists) of x as a word in $a_1, ..., a_k$.

We note that the membership *decision* problem is to determine whether or not a given $x \in G$ belongs to the subgroup of G generated by given $a_1, ..., a_k$. This problem turns out to be quite hard in most groups. For instance, the membership decision problem in a braid group B_n is algorithmically unsolvable if $n \geq 6$ because such a braid group contains subgroups isomorphic to $F_2 \times F_2$ (that would be, for example, the subgroup generated by $\sigma_1^2, \sigma_2^2, \sigma_4^2$, and σ_5^2, see [3]), where F_2 is the free group of rank 2. In the group $F_2 \times F_2$, the membership decision problem is algorithmically unsolvable by an old result of Mihailova [9].

We also note that if the adversary finds, say, some $x' \in G$ such that $b_1^x = b_1^{x'}, ...,$ $b_m^x = b_m^{x'}$, there is no guarantee that $x' = x$ in G. Indeed, if $x' = c_b x$, where $c_b b_i = b_i c_b$ for all i (in which case we say that c_b *centralizes* b_i), then $b_i^x = b_i^{x'}$ for all i, and therefore $b^x = b^{x'}$ for any element b from the subgroup generated by $b_1, ..., b_m$; in particular, $y^x = y^{x'}$. Now the problem is that if x' (and, similarly, y') does not belong to the subgroup A generated by $a_1, ..., a_k$ (respectively, to the subgroup B generated by $b_1, ..., b_m$), then the adversary may not obtain the correct common secret key K. On the other hand, if x' (and, similarly, y') does belong to the subgroup A (respectively, to the subgroup B), then the adversary will be able to get the correct K even though his x' and y' may be different from x and y, respectively. Indeed, if $x' = c_b x$, $y' = c_a y$, where c_b centralizes B and c_a centralizes A (elementwise), then

$$(x')^{-1}(y')^{-1}x'y' = (c_b x)^{-1}(c_a y)^{-1}c_b x c_a y = x^{-1}c_b^{-1}y^{-1}c_a^{-1}c_b x c_a y = x^{-1}y^{-1}xy = K$$

because c_b commutes with y and with c_a (note that c_a belongs to the subgroup B, which follows from the assumption $y' = c_a y \in B$, and, similarly, c_b belongs to A), and c_a commutes with x.

We emphasize that the adversary ends up with the corrrect key K (i.e., $(x')^{-1}(y')^{-1}x'y' = x^{-1}y^{-1}xy$) *if and only if* c_b commutes with c_a. The only visible way to ensure this is to have $x' \in A$ and $y' \in B$. Without verifying at least one of these inclusions, there seems to be no way for the adversary to make sure that he got the correct key.

Therefore, it appears that if the adversary chooses to solve the conjugacy search problem in the group G to recover x and y, he will then have to face either the membership search problem or the membership decision problem; the latter may very well be algorithmically unsolvable in a given group. The bottom line is that the adversary should actually be solving a more difficult version of the conjugacy search problem:

Given a group G, a subgroup $A \leq G$, and two elements $g, h \in G$, find $x \in A$ such that $h = x^{-1}gx$, given that at least one such x exists.

Finally, we note that what we have said in this section does not affect some heuristic attacks on the Anshel-Anshel-Goldfeld protocol suggested by several authors [4, 5, 7] because these attacks, which use "neighbourhood search" type (in a group-theoretic context also called "length based") heuristic algorithms, are targeted, by design, at finding a solution of a given equation (or a system of equations) as a word in given elements. The point that we make in this section is that even if a fast (polynomial-time) *deterministic* algorithm is found for solving the conjugacy search problem in braid groups, this will not be sufficient to break the Anshel-Anshel-Goldfeld protocol *by a deterministic attack.* As for heuristic attacks, their limitations are explained in [10].

References

[1] Anshel, I., Anshel, M., Goldfeld, D., *An algebraic method for public-key cryptography.* Math. Res. Lett. 6, 287–291 (1999).

[2] Cha, J. C., Ko, K. H., Lee, S. J., Han, J. W., Cheon, J. H., *An Efficient Implementation of Braid Groups.* In: ASIACRYPT 2001, Lecture Notes Comp. Sc., vol. 2248, 144–156 (2001).

[3] Collins, D., *Relations among the squares of the generators of the braid group*, Invent. Math. 117, 525–529 (1994).

[4] Garber, D., Kaplan, S., Teicher, M., Tsaban, B., Vishne, U., *Length-based conjugacy search in the braid group*, preprint.
http://arXiv.org/abs/math.GR/0209267

[5] Garber, D., Kaplan, S., Teicher, M., Tsaban, B., Vishne, U., *Probabilistic solutions of equations in the braid group*, preprint.
http://arxiv.org/abs/math.GR/0404076

[6] Hofheinz, D., Steinwandt, R., *A practical attack on some braid group based cryptographic primitives.* In: Public Key Cryptography, 6th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2003 Proceedings, 187–198 (Y. G. Desmedt, ed., Lecture Notes Comp. Sc., vol. 2567) Berlin Heidelberg New York Tokyo: Springer 2002.

[7] Hughes, J., Tannenbaum, A., *Length-based attacks for certain group based encryption rewriting systems.* In: Workshop SECI02 Securitè de la Communication sur Intenet, September 2002, Tunis, Tunisia.
http://www.network.com/˜hughes/

[8] Ko, K. H., Lee, S. J., Cheon, J. H., Han, J. W., Kang, J., Park, C., *New public-key cryptosystem using braid groups.* In: Advances in cryptology – CRYPTO 2000 (Santa Barbara, CA), 166–183 (Lecture Notes Comp. Sc., vol. 1880) Berlin Heidelberg New York Tokyo: Springer 2000.

[9] Mihailova, K. A., *The occurrence problem for direct products of groups.* Dokl. Akad. Nauk SSSR 119, 1103–1105 (1958). (Russian)

[10] Shpilrain, V., *Assessing security of some group based cryptosystems.* In: Group theory, statistics, and cryptography, 167–177 (Contemp. Math., vol. 360) Providence, RI: Amer. Math. Soc. 2004.

Department of Mathematics, The City College of New York, New York, NY 10031
E-mail address: shpil@groups.sci.ccny.cuny.edu

Department of Mathematics, CUNY Graduate Center, New York, NY 10016
E-mail address: aushakov@mail.ru