Cryptanalysis of Stickel's key exchange scheme

Vladimir Shpilrain

Department of Mathematics, The City College of New York, New York, NY 10031
http://www.sci.ccny.cuny.edu/~shpil `shpil@groups.sci.ccny.cuny.edu` [*]

Abstract. We offer cryptanalysis of a key exchange scheme due to Stickel [11], which was inspired by the well-known Diffie-Hellman protocol. We show that Stickel's choice of platform (the group of invertible matrices over a finite field) makes the scheme vulnerable to linear algebra attacks with very high success rate in recovering the shared secret key (100% in our experiments). We also show that obtaining the shared secret key in Stickel's scheme is not harder for the adversary than solving the *decomposition search problem* in the platform (semi)group.

1 Introduction

In this paper, we offer cryptanalysis of a key exchange scheme due to Stickel [11]. His protocol is reminiscent of the well-known Diffie-Hellman protocol (see e.g. [3]), although formally it is not a generalization of the latter. We show in our Section 4 that Stickel's choice of platform (the group of invertible matrices over a finite field) makes the protocol vulnerable to linear algebra attacks. It appears that even such a seemingly minor improvement as using non-invertible matrices instead of invertible ones would already make Stickel's protocol significantly less vulnerable, at least to linear algebra attacks.

Perhaps more importantly, we show in Section 3 that to obtain the shared secret key in Stickel's scheme, the adversary does not have to solve any discrete logarithm-type problem; instead, he/she can solve the apparently easier *decomposition search problem* in the platform (semi)group G which is:

> Given a recursively presented (semi)group G, two recursively generated sub(semi)groups $A, B \leq G$, and two elements $u, w \in G$, find two elements $x \in A$ and $y \in B$ that would satisfy $x \cdot w \cdot y = u$, provided at least one such pair of elements exists.

We give some background on this problem in Section 3, and in Section 4 we describe a platform-specific attack on Stickel's scheme which boils down to solving a system of 1922 linear equations with 961 unknowns over a finite field $\mathbf{F}_{2^m}$. Finally, in Section 5 we give a couple of simple suggestions on improving Stickel's scheme.

2 Stickel's protocol

Let G be a public non-abelian finite group, $a, b \in G$ public elements such that $ab \neq ba$. The key exchange protocol goes as follows. Let N and M be the orders of a and b, respectively.

1. Alice picks two random natural numbers $n < N, m < M$ and sends $u = a^n b^m$ to Bob.
2. Bob picks two random natural numbers $r < N, s < M$ and sends $v = a^r b^s$ to Alice.
3. Alice computes $K_A = a^n v b^m = a^{n+r} b^{m+s}$.
4. Bob computes $K_B = a^r u b^s = a^{n+r} b^{m+s}$.

Thus, Alice and Bob end up with the same group element $K = K_A = K_B$ which can serve as the shared secret key.

When it comes to implementation details, the exposition in [11] becomes somewhat foggy. In particular, it seems that the author actually prefers the following more general version of the above protocol.

Let $w \in G$ be public.

1. Alice picks two random natural numbers $n < N, m < M$, an element c_1 from the center of the group G, and sends $u = c_1 a^n w b^m$ to Bob.
2. Bob picks two random natural numbers $r < N, s < M$, an element c_2 from the center of the group G, and sends $v = c_2 a^r w b^s$ to Alice.
3. Alice computes $K_A = c_1 a^n v b^m = c_1 c_2 a^{n+r} w b^{m+s}$.
4. Bob computes $K_B = c_2 a^r u b^s = c_1 c_2 a^{n+r} w b^{m+s}$.

Thus, Alice and Bob end up with the same group element $K = K_A = K_B$.

We note that for this protocol to work, G does not have to be a group; a semigroup would do just as well (in fact, even better, as we argue in Section 5).

In [11], it was suggested that the group of invertible $k \times k$ matrices over a finite field F_{2^l} is used as the platform group G. We show in Section 4 that this choice of platform makes the protocol vulnerable to linear algebra attacks, but first, in Section 3, we discuss a general (i.e., not platform-specific) approach to attacking Stickel's protocol. We emphasize that this general approach works if G is any semigroup, whereas the attack in Section 4 is platform-specific; in particular, it only works if G is a group, but may not work for arbitrary semigroups.

3 Preliminary cryptanalysis of Stickel's protocol

Recall that Alice transmits $u = c_1 a^n w b^m$ to Bob.

Our first observation is: to get a hold of the shared secret key K in the end, it is sufficient for the adversary (Eve) to find any elements $x, y \in G$ such that $xa = ax$, $yb = by$, $u = xwy$. Indeed, having found such x, y, Eve can use Bob's transmission $v = c_2 a^r w b^s$ to compute:

$$xvy = xc_2a^rwb^sy = c_2a^rxwyb^s = c_2a^rub^s = K.$$

This implies, in particular, that multiplying by c_i does not enhance the security of the protocol. More importantly, this also implies that it is not necessary for Eve to recover any of the exponents n, m, r, s; instead, she can just solve a system of equations $xa = ax$, $yb = by$, $u = xwy$, where a, b, u, w are known and x, y unknown elements of the platform (semi)group G. This shows that, in fact, Stickel's protocol departs from the Diffie-Hellman protocol farther than it seems. Moreover, solving the above system of equations in G is actually nothing else but solving the (subsemigroup-restricted) *decomposition search problem* which is:

> Given a recursively presented (semi)group G, two recursively generated sub(semi)groups $A, B \leq G$, and two elements $u, w \in G$, find two elements $x \in A$ and $y \in B$ that would satisfy $x \cdot w \cdot y = u$, provided at least one such pair of elements exists.

In reference to Stickel's scheme, the sub(semi)groups A and B are the *centralizers* of the elements a and b, respectively. The centralizer of an element $g \in G$ is the set of all elements $c \in G$ such that $gc = cg$. This set is a subsemigroup of G; if G is a group, then this set is a subgroup.

There are several key exchange protocols that directly use the alleged hardness of the decomposition search problem in various (semi)groups, see e.g. [2], [7], [8], [9]. So far, no particular (semi)group has been recognized as providing a secure platform for any of those protocols. Several attacks on the decomposition search problem in various "abstract" groups (i.e., in groups given by generators and defining relators) were reported, see e.g. [1], [4], [5]. It appears likely that semigroups of matrices over specific rings can generally make good platforms, as we argue in [6]. Stickel, too, used matrices in his paper [11], but he has made several poor choices, as we are about to see in the next Section 4. Also, Stickel's scheme is *at most* as secure as those schemes that are directly based on the alleged hardness of the decomposition search problem, because there are ways to attack Stickel's scheme without attacking the relevant decomposition search problem; for instance, Sramka [10] has offered an attack aimed at recovering one of the exponents n, m, r, s in Stickel's protocol. Our attack that we describe in Section 4 is more efficient, but on the other hand, it is aimed at recovering the shared secret key only, whereas Sramka's attack is aimed at recovering a private key.

4 Linear algebra attack

Now we are going to focus on the particular platform group G suggested by Stickel in [11]. In his paper, G is the group of invertible $k \times k$ matrices over a finite field $\mathbf{F}_{2^l}$, where $k = 31$. The parameter l was not specified in [11], but from what is written there, one can reasonably guess that $2 \leq l \leq k$. The choice

of matrices a, b, w is not so important for our attack; what is important is that a and b are invertible. We note however that the choice of matrices a and b in [11] (more specifically, the fact that the entries of these matrices are either 0 or 1) provides an extra weakness to the scheme as we will see at the end of this section.

Recall from our Section 3 that it is sufficient for Eve to find at least one solution of the system of equations $xa = ax$, $yb = by$, $u = xwy$, where a, b, u, w are known and x, y unknown $k \times k$ matrices over $\mathbf{F}_{2^l}$. Each of the first two equations in this system translates into a system of k^2 linear equations for the (unknown) entries of the matrices x and y. However, the equation $u = xwy$ does not translate into a system of linear equations for the entries because it has a product of two unknown matrices. We therefore have to use the following trick: multiply both sides of the equation $u = xwy$ by x^{-1} on the left (here is where we use the fact that x is invertible!) to get

$$x^{-1}u = wy.$$

Now, since $xa = ax$ if and only if $x^{-1}a = ax^{-1}$, we denote $x_1 = x^{-1}$ and replace the system of equations mentioned in the previous paragraph by the following one:

$$x_1 a = ax_1, \ yb = by, \ x_1 u = wy.$$

Now each equation in this system translates into a system of k^2 linear equations for the (unknown) entries of the matrices x_1 and y. Thus, we have a total of $3k^2$ linear equations with $2k^2$ unknowns. Note however that a solution of the displayed system will yield the shared key K if and only if x_1 is invertible because $K = xvy$, where $x = x_1^{-1}$.

Since u is a known invertible matrix, we can multiply both sides of the equation $x_1 u = wy$ by u^{-1} on the right to get $x_1 = wyu^{-1}$, and then eliminate x_1 from the system:

$$wyu^{-1}a = awyu^{-1}, \ yb = by.$$

Now we have just one unknown matrix y, so we have $2k^2$ linear equations for k^2 entries of y. Thus, we have a heavily overdetermined system of linear equations (recall that in Stickel's paper, $k = 31$, so $k^2 = 961$). We know that this system must have at least one non-trivial (i.e., non-zero) solution; therefore, if we reduce the matrix of this system to an echelon form, there should be at least one free variable. On the other hand, since the system is heavily overdetermined, we can expect the number of free variables to be not too big, so that it is feasible to go over possible values of free variables one at a time, until we find some values that yield an invertible matrix y. (Recall that all entries of y are either 0 or 1; this is an extra weakness of Stickel's scheme that we mentioned before.) Note that checking the invertibility of a given matrix is easy because it is equivalent to reducing the matrix to an echelon form. In fact, in all our experiments there was just one free variable, so the last step (checking the invertibility) was not

needed because if there is a unique non-zero solution of the above system, then the corresponding matrix y should be invertible.

5 Suggestions on improving Stickel's scheme

The most obvious suggestion on improving Stickel's scheme is, as we mentioned before, to use non-invertible elements a, b, w; this implies, in particular, that the platform should be a semigroup with (a lot of) non-invertible elements. If one is to use matrices, then it makes sense to use the semigroup of *all $k \times k$* matrices over a finite ring (not necessarily a field!). Such a semigroup typically has a lot of non-invertible elements, so it should be easy to choose a, b, w non-invertible, in which case the linear algebra attack from the previous section would not work. One more advantage of not restricting the pool to invertible matrices is that one can use not just powers a^j of a given public matrix in Stickel's protocol, but arbitrary expressions of the form $\sum_{i=1}^{p} c_i \cdot a^i$, where c_i are constants, i.e. elements of the ground ring.

Of course, there is no compelling reason why matrices should be employed in Stickel's scheme, but as we have explained in Section 3, with an abstract platform (semi)group G, Stickel's scheme is broken if the relevant decomposition search problem is solved, and so far, no particular abstract (semi)group has been recognized as resistant to known attacks on the decomposition search problem.

6 Conclusions

1. We have shown that obtaining the shared secret key K in Stickel's scheme is not harder for the adversary than solving the decomposition search problem in the platform (semi)group G.
2. We have described an efficient linear algebra attack, with 100% success rate (according to our experiments), on Stickel's scheme with parameters suggested in [11].
3. We have suggested possible improvements of Stickel's scheme related to the choice of platform; our main suggestion is to use a semigroup with a lot of non-invertible elements instead of a group.

References

1. D. Garber, S. Kaplan, M. Teicher, B. Tsaban, U. Vishne, *Probabilistic solutions of equations in the braid group*, Advances in Applied Mathematics **35** (2005), 323–334.
2. K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J. Kang, C. Park, *New public-key cryptosystem using braid groups*, Advances in cryptology—CRYPTO 2000 (Santa Barbara, CA), 166–183, Lecture Notes in Comput. Sci. **1880**, Springer, Berlin, 2000.
3. A. J. Menezes, *Handbook of Applied Cryptography*. CRC Press, 1996.

4. A. G. Myasnikov, V. Shpilrain, and A. Ushakov, *A practical attack on some braid group based cryptographic protocols*, in CRYPTO 2005, Lecture Notes Comp. Sci. **3621** (2005), 86–96.
5. D. Ruinskiy, A. Shamir, B. Tsaban, *Cryptanalysis of group-based key agreement protocols using subgroup distance functions*, in PKC 2007, Lecture Notes Comp. Sc. **4450** (2007), 61-75.
6. V. Shpilrain, *Hashing with polynomials*, in: ICISC 2006, Lecture Notes Comp. Sc. **4296** (2006), 22–28.
7. V. Shpilrain and A. Ushakov, *Thompson's group and public key cryptography*, Lecture Notes Comp. Sc. **3531** (2005), 151–164.
8. V. Shpilrain and A. Ushakov, *A new key exchange protocol based on the decomposition problem*, Contemp. Math., Amer. Math. Soc. **418** (2006), 161–167.
9. V. M. Sidelnikov, M. A. Cherepnev, and V. Y. Yashcenko, *Systems of open distribution of keys on the basis of noncommutative semigroups*, Ross. Acad. Nauk Dokl. **332** (1993). English translation: Russian Acad. Sci. Dokl. Math. **48** (1994), 384-386.
10. M. Sramka, *On the Security of Stickel's Key Exchange Scheme*, preprint.
11. E. Stickel, *A New Method for Exchanging Secret Keys.* In: Proc. of the Third International Conference on Information Technology and Applications (ICITA05) 2 (2005), 426–430.