Vladimir Shpilrain
The City College of New York
shpil@groups.sci.ccny.cuny.edu

February 29, 2012

The Ko-Lee et al. protocol

1. Alice and Bob agree on a group G and an element w in G. Thus, G and w are public.
2. Alice picks a private $a \in G$ and sends $w^a = a^{-1}wa$ to Bob.
3. Bob picks a private $b \in G$ and sends $w^b = b^{-1}wb$ to Alice.
4. Alice computes $K_A = (w^b)^a = w^{ba}$, and Bob computes $K_B = (w^a)^b = w^{ab}$.

If $ab = ba$, then Alice and Bob get a common private key $K_B = w^{ab} = w^{ba} = K_A$. Typically, there are two public subgroups A and B of the group G, given by their (finite) generating sets, such that $ab = ba$ for any $a \in A$, $b \in B$.

Example (Ko-Lee). Braid group.

The Ko-Lee et al. protocol

1. Alice and Bob agree on a group G and an element w in G. Thus, G and w are public.
2. Alice picks a private $a \in G$ and sends $w^a = a^{-1}wa$ to Bob.
3. Bob picks a private $b \in G$ and sends $w^b = b^{-1}wb$ to Alice.
4. Alice computes $K_A = (w^b)^a = w^{ba}$, and Bob computes $K_B = (w^a)^b = w^{ab}$.

If $ab = ba$, then Alice and Bob get a common private key $K_B = w^{ab} = w^{ba} = K_A$. Typically, there are two public subgroups A and B of the group G, given by their (finite) generating sets, such that $ab = ba$ for any $a \in A$, $b \in B$.

Example (Ko-Lee). Braid group.

The platform group G

(P0) The group G has to be well known. More specifically, the *conjugacy search problem* (i.e., recovering a from $(w,\ a^{-1}wa)$) in the group G either has to be well studied or can be reduced to a well-known problem.

(P1) The word problem in G should have a fast (e.g. quadratic-time) solution by a deterministic algorithm. Better yet, there should be an efficiently computable "normal form" for elements of G.

(P2) The conjugacy search problem should *not* have an efficient solution by a deterministic algorithm.

(P3) There should be a way to disguise elements of G so that it would be impossible to recover x from $x^{-1}wx$ just by inspection. Example: "normal form".

(P4) G should be "large", i.e. have a "fast growth". This is necessary to have a sufficiently large key space.

The platform group G

(P0) The group G has to be well known. More specifically, the *conjugacy search problem* (i.e., recovering a from $(w,\ a^{-1}wa)$) in the group G either has to be well studied or can be reduced to a well-known problem.

(P1) The word problem in G should have a fast (e.g. quadratic-time) solution by a deterministic algorithm. Better yet, there should be an efficiently computable "normal form" for elements of G.

(P2) The conjugacy search problem should *not* have an efficient solution by a deterministic algorithm.

(P3) There should be a way to disguise elements of G so that it would be impossible to recover x from $x^{-1}wx$ just by inspection. Example: "normal form".

(P4) G should be "large", i.e. have a "fast growth". This is necessary to have a sufficiently large key space.

The platform group G

(P0) The group G has to be well known. More specifically, the *conjugacy search problem* (i.e., recovering a from $(w,\ a^{-1}wa)$) in the group G either has to be well studied or can be reduced to a well-known problem.

(P1) The word problem in G should have a fast (e.g. quadratic-time) solution by a deterministic algorithm. Better yet, there should be an efficiently computable "normal form" for elements of G.

(P2) The conjugacy search problem should *not* have an efficient solution by a deterministic algorithm.

(P3) There should be a way to disguise elements of G so that it would be impossible to recover x from $x^{-1}wx$ just by inspection. Example: "normal form".

(P4) G should be "large", i.e. have a "fast growth". This is necessary to have a sufficiently large key space.

The platform group G

(P0) The group G has to be well known. More specifically, the *conjugacy search problem* (i.e., recovering a from $(w,\ a^{-1}wa)$) in the group G either has to be well studied or can be reduced to a well-known problem.

(P1) The word problem in G should have a fast (e.g. quadratic-time) solution by a deterministic algorithm. Better yet, there should be an efficiently computable "normal form" for elements of G.

(P2) The conjugacy search problem should *not* have an efficient solution by a deterministic algorithm.

(P3) There should be a way to disguise elements of G so that it would be impossible to recover x from $x^{-1}wx$ just by inspection. Example: "normal form".

(P4) G should be "large", i.e. have a "fast growth". This is necessary to have a sufficiently large key space.

The platform group G

(P0) The group G has to be well known. More specifically, the *conjugacy search problem* (i.e., recovering a from $(w,\ a^{-1}wa)$) in the group G either has to be well studied or can be reduced to a well-known problem.

(P1) The word problem in G should have a fast (e.g. quadratic-time) solution by a deterministic algorithm. Better yet, there should be an efficiently computable "normal form" for elements of G.

(P2) The conjugacy search problem should *not* have an efficient solution by a deterministic algorithm.

(P3) There should be a way to disguise elements of G so that it would be impossible to recover x from $x^{-1}wx$ just by inspection. Example: "normal form".

(P4) G should be "large", i.e. have a "fast growth". This is necessary to have a sufficiently large key space.

Ramifications of the Ko-Lee protocol

1. Alice and Bob agree on a group G, two subsets $A, B \subseteq G$ commuting elementwise, and an element w in G.
2. Alice randomly selects private elements $a_1, a_2 \in A$. Then she sends the element $a_1 w a_2$ to Bob.
3. Bob randomly selects private elements $b_1, b_2 \in B$. Then he sends the element $b_1 w b_2$ to Alice.
4. Alice computes $K_A = a_1 b_1 w b_2 a_2$, and Bob computes $K_B = b_1 a_1 w a_2 b_2$. Since $a_i b_i = b_i a_i$ in G, one has $K_A = K_B = K$.

Using matrices

Stickel 2005, Maze-Monico-Rosenthal 2007

There is a public ring (or a semiring) R and public $n \times n$ matrices S, M_1, and M_2 over R. The ring R should have a non-trivial commutative subring C. One way to guarantee that would be for R to be an algebra over a field K; then, of course, $C = K$ will be a commutative subring of R.

1. Alice chooses polynomials $p_A(x), q_A(x) \in C[x]$ and sends the matrix $U = p_A(M_1) \cdot S \cdot q_A(M_2)$ to Bob.
2. Bob chooses polynomials $p_B(x), q_B(x) \in C[x]$ and sends the matrix $V = p_B(M_1) \cdot S \cdot q_B(M_2)$ to Alice.
3. Alice computes $K_A = p_A(M_1) \cdot V \cdot q_A(M_2) = p_A(M_1) \cdot p_B(M_1) \cdot S \cdot q_B(M_2) \cdot q_A(M_2)$.
4. Bob computes $K_B = p_B(M_1) \cdot U \cdot q_B(M_2) = p_B(M_1) \cdot p_A(M_1) \cdot S \cdot q_A(M_2) \cdot q_B(M_2)$.

Since any two polynomials in the same matrix commute, one has $K = K_A = K_B$, the shared secret key.

Using matrices

Stickel 2005, Maze-Monico-Rosenthal 2007

There is a public ring (or a semiring) R and public $n \times n$ matrices S, M_1, and M_2 over R. The ring R should have a non-trivial commutative subring C. One way to guarantee that would be for R to be an algebra over a field K; then, of course, $C = K$ will be a commutative subring of R.

1. Alice chooses polynomials $p_A(x), q_A(x) \in C[x]$ and sends the matrix $U = p_A(M_1) \cdot S \cdot q_A(M_2)$ to Bob.
2. Bob chooses polynomials $p_B(x), q_B(x) \in C[x]$ and sends the matrix $V = p_B(M_1) \cdot S \cdot q_B(M_2)$ to Alice.
3. Alice computes $K_A = p_A(M_1) \cdot V \cdot q_A(M_2) = p_A(M_1) \cdot p_B(M_1) \cdot S \cdot q_B(M_2) \cdot q_A(M_2)$.
4. Bob computes $K_B = p_B(M_1) \cdot U \cdot q_B(M_2) = p_B(M_1) \cdot p_A(M_1) \cdot S \cdot q_A(M_2) \cdot q_B(M_2)$.

Since any two polynomials in the same matrix commute, one has $K = K_A = K_B$, the shared secret key.

The Anshel-Anshel-Goldfeld protocol

Can use ANY non-abelian group with efficiently solvable word problem as the platform.

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 then computes $K_A = x^{-1} \cdot (y^{-1}xy) = x^{-1}y^{-1}xy$.
4. Bob computes $y(b_1^x, ..., b_m^x) = y^x = x^{-1}yx$, and then computes $K_B = (y^{-1} \cdot x^{-1}yx)^{-1} = x^{-1}y^{-1}xy$.

   Thus, $K = K_A = K_B$ is the shared secret key.

Platform groups

- Braid groups
- Thompson's group
- Small cancellation groups
- Groups of matrices over various rings

Semidirect product

Let G, H be two groups, let $Aut(G)$ be the group of automorphisms of G, and let $\rho : H \to Aut(G)$ be a homomorphism. Then the semidirect product of G and H is the set

$$\Gamma = G \rtimes_{\rho} H = \{(g, h) : g \in G,\ h \in H\}$$

with the group operation given by

$$(g, h)(g', h') = (g^{\rho(h)} \cdot g',\ h \cdot h').$$

Here $g^{\rho(h)}$ denotes the image of g under the automorphism $\rho(h)$.

Holomorph

If $H = Aut(G)$, then the corresponding semidirect product is called the *holomorph* of the group G. Thus, the holomorph of G, usually denoted by $Hol(G)$, is the set of all pairs $(g,\ \phi)$, where $g \in G,\ \phi \in Aut(G)$, with the group operation given by

$$(g,\ \phi) \cdot (g',\ \phi') = (\phi'(g) \cdot g',\ \phi \cdot \phi').$$

It is often more practical to use a subgroup of $Aut(G)$ in this construction.

Also, if we want the result to be just a semigroup, not necessarily a group, we can consider the semigroup $End(G)$ instead of the group $Aut(G)$ in this construction.

Holomorph

If $H = Aut(G)$, then the corresponding semidirect product is called the *holomorph* of the group G. Thus, the holomorph of G, usually denoted by $Hol(G)$, is the set of all pairs $(g,\ \phi)$, where $g \in G,\ \phi \in Aut(G)$, with the group operation given by

$$(g,\ \phi) \cdot (g',\ \phi') = (\phi'(g) \cdot g',\ \phi \cdot \phi').$$

It is often more practical to use a subgroup of $Aut(G)$ in this construction.

Also, if we want the result to be just a semigroup, not necessarily a group, we can consider the semigroup $End(G)$ instead of the group $Aut(G)$ in this construction.

Holomorph

If $H = Aut(G)$, then the corresponding semidirect product is called the *holomorph* of the group G. Thus, the holomorph of G, usually denoted by $Hol(G)$, is the set of all pairs $(g, \ \phi)$, where $g \in G, \ \phi \in Aut(G)$, with the group operation given by

$$(g, \ \phi) \cdot (g', \ \phi') = (\phi'(g) \cdot g', \ \phi \cdot \phi').$$

It is often more practical to use a subgroup of $Aut(G)$ in this construction.

Also, if we want the result to be just a semigroup, not necessarily a group, we can consider the semigroup $End(G)$ instead of the group $Aut(G)$ in this construction.

Thank you