Vladimir Shpilrain
The City College of New York

Public key exchange using semidirect product of (semi)groups

November 15, 2012

The Diffie-Hellman public key exchange (1976)

1. Alice and Bob agree on a public (finite) cyclic group G and a generating element g in G. We will write the group G multiplicatively.
2. Alice picks a random natural number a and sends g^a to Bob.
3. Bob picks a random natural number b and sends g^b to Alice.
4. Alice computes $K_A = (g^b)^a = g^{ba}$.
5. Bob computes $K_B = (g^a)^b = g^{ab}$.

Since $ab = ba$ (because $\mathbb{Z}$ is commutative), both Alice and Bob are now in possession of the same group element $K = K_A = K_B$ which can serve as the shared secret key.

Efficiency for legitimate parties

Exponentiation by "square-and-multiply":

$$g^{22} = (((g^2)^2)^2)^2 \cdot (g^2)^2 \cdot g^2$$

Complexity of computing g^n is therefore $O(\log n)$, times complexity of reducing *mod p* (more generally, reducing to a "normal form").

Efficiency for legitimate parties

Exponentiation by "square-and-multiply":

$$g^{22} = (((g^2)^2)^2)^2 \cdot (g^2)^2 \cdot g^2$$

Complexity of computing g^n is therefore $O(\log n)$, times complexity of reducing *mod p* (more generally, reducing to a "normal form").

Efficiency for legitimate parties

Exponentiation by “square-and-multiply”:

$$g^{22} = (((g^2)^2)^2)^2 \cdot (g^2)^2 \cdot g^2$$

Complexity of computing g^n is therefore $O(\log n)$, times complexity of reducing *mod p* (more generally, reducing to a “normal form”).

Security assumptions

To recover g^{ab} from (g, g^a, g^b) is hard.

To recover a from (g, g^a) (discrete log problem) is hard.

Security assumptions

To recover g^{ab} from (g, g^a, g^b) is hard.

To recover a from (g, g^a) (discrete log problem) is hard.

Variations on Diffie-Hellman: why not just multiply them?

1. Alice and Bob agree on a (finite) cyclic group G and a generating element g in G. We will write the group G multiplicatively.
2. Alice picks a random natural number a and sends g^a to Bob.
3. Bob picks a random natural number b and sends g^b to Alice.
4. Alice computes $K_A = (g^b) \cdot (g^a) = g^{b+a}$.
5. Bob computes $K_B = (g^a) \cdot (g^b) = g^{a+b}$.

Obviously, $K_A = K_B = K$, which can serve as the shared secret key.

Drawback: anybody can obtain K the same way!

Variations on Diffie-Hellman: why not just multiply them?

1. Alice and Bob agree on a (finite) cyclic group G and a generating element g in G. We will write the group G multiplicatively.
2. Alice picks a random natural number a and sends g^a to Bob.
3. Bob picks a random natural number b and sends g^b to Alice.
4. Alice computes $K_A = (g^b) \cdot (g^a) = g^{b+a}$.
5. Bob computes $K_B = (g^a) \cdot (g^b) = g^{a+b}$.

Obviously, $K_A = K_B = K$, which can serve as the shared secret key.

Drawback: anybody can obtain K the same way!

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.

Cayley-Hamilton

Note: The whole ring R should not be commutative because otherwise, the Cayley-Hamilton theorem kills large powers of a matrix.

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 $\rho(h)$.

Extensions by automorphisms

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.

Extensions by automorphisms

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.

Extensions by automorphisms

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.

Key exchange using extensions by automorphisms (Habeeb-Kahrobaei-Koupparis-Shpilrain)

Let G be a group (or a semigroup). An element $g \in G$ is chosen and made public as well as an arbitrary automorphism (or an endomorphism) ϕ of G. Bob chooses a private $n \in \mathbb{N}$, while Alice chooses a private $m \in \mathbb{N}$. Both Alice and Bob are going to work with elements of the form (g, ϕ^k), where $g \in G,\ k \in \mathbb{N}$.

1. Alice computes $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^m)$ and sends **only the first component** of this pair to Bob. Thus, she sends to Bob **only** the element $a = \phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$ of the group G.

2. Bob computes $(g, \phi)^n = (\phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^n)$ and sends **only the first component** of this pair to Alice: $b = \phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$.

3. Alice computes $(b, x) \cdot (a,\ \phi^m) = (\phi^m(b) \cdot a,\ x \cdot \phi^m)$. Her key is now $K_A = \phi^m(b) \cdot a$. Note that she does not actually "compute" $x \cdot \phi^m$ because she does not know the automorphism x; recall that it was not transmitted to her. But she does not need it to compute K_A.

Key exchange using extensions by automorphisms (Habeeb-Kahrobaei-Koupparis-Shpilrain)

Let G be a group (or a semigroup). An element $g \in G$ is chosen and made public as well as an arbitrary automorphism (or an endomorphism) ϕ of G. Bob chooses a private $n \in \mathbb{N}$, while Alice chooses a private $m \in \mathbb{N}$. Both Alice and Bob are going to work with elements of the form (g, ϕ^k), where $g \in G,\ k \in \mathbb{N}$.

1. Alice computes $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^m)$ and sends **only the first component** of this pair to Bob. Thus, she sends to Bob **only** the element $a = \phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$ of the group G.

2. Bob computes $(g, \phi)^n = (\phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^n)$ and sends **only the first component** of this pair to Alice: $b = \phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$.

3. Alice computes $(b, x) \cdot (a,\ \phi^m) = (\phi^m(b) \cdot a,\ x \cdot \phi^m)$. Her key is now $K_A = \phi^m(b) \cdot a$. Note that she does not actually "compute" $x \cdot \phi^m$ because she does not know the automorphism x; recall that it was not transmitted to her. But she does not need it to compute K_A.

Key exchange using extensions by automorphisms (Habeeb-Kahrobaei-Koupparis-Shpilrain)

Let G be a group (or a semigroup). An element $g \in G$ is chosen and made public as well as an arbitrary automorphism (or an endomorphism) ϕ of G. Bob chooses a private $n \in \mathbb{N}$, while Alice chooses a private $m \in \mathbb{N}$. Both Alice and Bob are going to work with elements of the form (g, ϕ^k), where $g \in G,\ k \in \mathbb{N}$.

1. Alice computes $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^m)$ and sends **only the first component** of this pair to Bob. Thus, she sends to Bob **only** the element $a = \phi^{m-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$ of the group G.

2. Bob computes $(g, \phi)^n = (\phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g,\ \phi^n)$ and sends **only the first component** of this pair to Alice: $b = \phi^{n-1}(g) \cdots \phi^2(g) \cdot \phi(g) \cdot g$.

3. Alice computes $(b, x) \cdot (a,\ \phi^m) = (\phi^m(b) \cdot a,\ x \cdot \phi^m)$. Her key is now $K_A = \phi^m(b) \cdot a$. Note that she does not actually "compute" $x \cdot \phi^m$ because she does not know the automorphism x; recall that it was not transmitted to her. But she does not need it to compute K_A.

Using semidirect product (cont.)

4. Bob computes $(a, y) \cdot (b,\ \phi^n) = (\phi^n(a) \cdot b,\ y \cdot \phi^n)$. His key is now $K_B = \phi^n(a) \cdot b$. Again, Bob does not actually "compute" $y \cdot \phi^n$ because he does not know the automorphism y.

5. Since $(b, x) \cdot (a,\ \phi^m) = (a,\ y) \cdot (b,\ \phi^n) = (g,\ \phi)^{m+n}$, we should have $K_A = K_B = K$, the shared secret key.

Special case: Diffie-Hellman

$G = \mathbb{Z}_p^*$
$\phi(g) = g^k$ for all $g \in G$ and a fixed k, $1 < k < p-1$.

Then $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi(g) \cdot \phi^2(g) \cdot g,\ \phi^m)$.
The first component is equal to $g^{k^{m-1}+\ldots+k+1} = g^{\frac{k^m-1}{k-1}}$.
The shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$.

"The Diffie-Hellman type problem" would be to recover the shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$ from the triple $(g,\ g^{\frac{k^m-1}{k-1}},\ g^{\frac{k^n-1}{k-1}})$. Since g and k are public, this is equivalent to recovering $g^{k^{m+n}}$ from the triple $(g,\ g^{k^m},\ g^{k^n})$, i.e., this is exactly the standard Diffie-Hellman problem.

Special case: Diffie-Hellman

$G = \mathbb{Z}_p^*$
$\phi(g) = g^k$ for all $g \in G$ and a fixed k, $1 < k < p-1$.

Then $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi(g) \cdot \phi^2(g) \cdot g,\ \phi^m)$.
The first component is equal to $g^{k^{m-1}+\ldots+k+1} = g^{\frac{k^m-1}{k-1}}$.
The shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$.

"The Diffie-Hellman type problem" would be to recover the shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$ from the triple $(g,\ g^{\frac{k^m-1}{k-1}},\ g^{\frac{k^n-1}{k-1}})$. Since g and k are public, this is equivalent to recovering $g^{k^{m+n}}$ from the triple $(g,\ g^{k^m},\ g^{k^n})$, i.e., this is exactly the standard Diffie-Hellman problem.

Special case: Diffie-Hellman

$G = \mathbb{Z}_p^*$
$\phi(g) = g^k$ for all $g \in G$ and a fixed k, $1 < k < p - 1$.

Then $(g, \phi)^m = (\phi^{m-1}(g) \cdots \phi(g) \cdot \phi^2(g) \cdot g,\ \phi^m)$.
The first component is equal to $g^{k^{m-1}+\ldots+k+1} = g^{\frac{k^m-1}{k-1}}$.
The shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$.

"The Diffie-Hellman type problem" would be to recover the shared key $K = g^{\frac{k^{m+n}-1}{k-1}}$ from the triple $(g,\ g^{\frac{k^m-1}{k-1}},\ g^{\frac{k^n-1}{k-1}})$. Since g and k are public, this is equivalent to recovering $g^{k^{m+n}}$ from the triple $(g,\ g^{k^m},\ g^{k^n})$, i.e., this is exactly the standard Diffie-Hellman problem.

Platform: matrices over group rings

Our general protocol can be used with *any* non-commutative group G if ϕ is selected to be an inner automorphism. Furthermore, it can be used with any non-commutative *semigroup* G as well, as long as G has some invertible elements; these can be used to produce inner automorphisms. A typical example of such a semigroup would be a semigroup of matrices over some ring.

We use the semigroup of 3×3 matrices over the group ring $\mathbb{Z}_7[A_5]$, where A_5 is the alternating group on 5 elements.

Then the public key consists of two matrices: the (invertible) conjugating matrix H and a (non-invertible) matrix M. The shared secret key then is:

$K = H^{-(m+n)}(HM)^{m+n}$.

Platform: matrices over group rings

Our general protocol can be used with *any* non-commutative group G if ϕ is selected to be an inner automorphism. Furthermore, it can be used with any non-commutative *semigroup* G as well, as long as G has some invertible elements; these can be used to produce inner automorphisms. A typical example of such a semigroup would be a semigroup of matrices over some ring.

We use the semigroup of 3×3 matrices over the group ring $\mathbb{Z}_7[A_5]$, where A_5 is the alternating group on 5 elements.
Then the public key consists of two matrices: the (invertible) conjugating matrix H and a (non-invertible) matrix M. The shared secret key then is:
$K = H^{-(m+n)}(HM)^{m+n}$.

Security assumptions

To recover $H^{-(m+n)}(HM)^{m+n}$ from $(M,\ H,\ H^{-m}(HM)^m,\ H^{-n}(HM)^n)$ is hard.

To recover m from $H^{-m}(HM)^m$ is hard.

Security assumptions

To recover $H^{-(m+n)}(HM)^{m+n}$ from $(M,\ H,\ H^{-m}(HM)^m,\ H^{-n}(HM)^n)$ is hard.

To recover m from $H^{-m}(HM)^m$ is hard.

Conclusions

• Even though the parties do compute a large power of a public element (as in the classical Diffie-Hellman protocol), they do not transmit the whole result, but rather just part of it.

• Since the classical Diffie-Hellman protocol is a special case of our protocol, breaking our protocol even for any cyclic group would imply breaking the Diffie-Hellman protocol.

Conclusions

• Even though the parties do compute a large power of a public element (as in the classical Diffie-Hellman protocol), they do not transmit the whole result, but rather just part of it.

• Since the classical Diffie-Hellman protocol is a special case of our protocol, breaking our protocol even for any cyclic group would imply breaking the Diffie-Hellman protocol.

Conclusions

• If the platform (semi)group is not commutative, then we get a new security assumption. In the simplest case, where the automorphism used for extension is inner, attacking a private exponent amounts to recovering an integer n from a product $g^{-n}h^n$, where g, h are public elements of the platform (semi)group. In the special case where $g = 1$ this boils down to recovering n from h^n, with public h ("discrete log" problem).

On the other hand, in the particular instantiation of our protocol, which is based on a non-commutative semigroup extended by an inner automorphism, recovering the shared secret key from public information is based on a different security assumption than the classical Diffie-Hellman protocol is.

Conclusions

• If the platform (semi)group is not commutative, then we get a new security assumption. In the simplest case, where the automorphism used for extension is inner, attacking a private exponent amounts to recovering an integer n from a product $g^{-n}h^{n}$, where g, h are public elements of the platform (semi)group. In the special case where $g = 1$ this boils down to recovering n from h^{n}, with public h ("discrete log" problem).

On the other hand, in the particular instantiation of our protocol, which is based on a non-commutative semigroup extended by an inner automorphism, recovering the shared secret key from public information is based on a different security assumption than the classical Diffie-Hellman protocol is.

Thank you