Alexander Lein-McDonough
Pascal Bakker

Final Project: Diffe-Hellman

December 8^{th}, 2017
MATH 4400-01
Mark Mixer

1 Introduction

Ever since we could send written messages between people, there has been a need to protect the information contained within from getting in the wrong hands. To cover for this fear, we have used methods of encryption to make the message sent look like a mess of characters rather than something readable. In the modern world with the emergence of the internet, this unease has not died out. Sending sensitive data across the web has become common place; however, with hackers and intrusive bodies continuously proving that this data can be intercepted and leaked, it has become quintessential for a more secure method of encryption to send information without the data being decyphered by unwanted parties. To do so, software has used relativley new, complex methods of encryption where the messages sent can be more well protected than when the seeds of cryptology were just being planted. One developed encryption method, is the Diffiie-Hellman Key Exchange; it uses the mathematical properties of the modulus function, as well as groups, to create a shared key that is computationally hard to break with the public information sent between clients. This paper will explain how the Diffe-Hellman key exchange works, and show how it operates within groups.

2 History

The protocol for what would be called the Diffe-Hellman key exchange would first be defined by Malcom Williamson, an agent for GCHQ, in 1974 (Rosen, 302). Since he was an agent for the GCHQ, the algorithm was created and used in secret (Rosen, 302). The algorithm would be first described in public two years later by Whitfield Diffe and Martin Hellman; This algorithm would be patented in the same year by the company, Public Key Partners (University of Cincinatti). In 1997, the patent expired and the algorithm became part of the public domain (University of Cincinatti).

3 Definitions and Concepts

In order to understand the Diffe-Hellman key exchange, we first must consdier what division means when the operation is closed within the set of integers, $\mathbb{Z}$. Let $a, b \in \mathbb{Z}$, then we can redefine division based upon the division algorithm, $a = qb + r$, where q is the quotient and r is the remainder upon dividing a by b (Rosen, 239). From this, we can define the modulus function as $r = a \mod b$. Given the definition of the modulus function, we can say that $m, n \in \mathbb{Z}$ are modularly congruent, denoted as $m \equiv n \pmod{b}$, if and only if $m \mod b = n \mod b$ and $b \in \mathbb{Z}^+$ (Rosen, 241). An integer i is the primitive root modulo n if $\forall x | x \not\equiv 0$

(mod b), $\exists k \in \mathbb{Z} | i^k \equiv x \pmod{b}$ where k is reffered to as the discrete logarithm of a to the base i mod b (Rosen, 306).

Before we start using the Diffe-Hellman key exchange, we would also need to understand some conceptual ideas in the world of encryption. The specific type of encryption that the Diffe-Hellman key exchange uses is Public key encryption, a method of encryption where the key used to encrypt your information is public knowledge, while the key (or keys) used to decrypt your message is kept secret (Rosen, 306). These keys used to encrypt and decrcypt your data are used in a one-way function, a function where it is easy to preform the operation with every input, but hard to invert the operation given any output (Kowalczyk).

To see how Diffe-Hellman works within groups, we will have to know what as group is, along with the properties of groups. $G = (S, +)$ where S is a set and $+$ is a binary operation, is a group if and only if

1. (Closure) $\forall a, b \in G, a + b \in G$
2. (Associativity) $\forall a, b, c \in G, (a + b) + c = a + (b + c)$
3. (Identity) $\exists e \in G | a + e = a, \forall a \in G$
4. (Inverses) $\forall a \in G, \exists b \in G | a + b = b + a = e$

(Gallian, 42-43). Based upon the definition of a group we can say that a group, G, is cosnidered to be an Abelian Group if and only if $\forall a, b \in G, a + b = b + a$ (Commutativity) (Gallian, 43). The function f, where $f : G \mapsto H$ and $G = (S, +), H = (T, *)$ are groups, is an isomorphism if and only if

1. f is a bijection from G to H
2. $\forall a, b \in G, f(a + b) = f(a) * f(b)$ (f preserves the group operations)

(Gallian, 121). We can say from this that an isomorphism f is an automorphism if $f : G \mapsto G$ (Gallian, 128).

4 Diffe-Hellman Example

The following is based upon an example from Discrete Mathematics and its Applications (Rosen, 302).

Alice and Bob want to send secret numbers over a direct line of communication. Knowing of the fact that their secret number could be seen by others if just sent on its own, they agree to use a Diffe Hellman Key Exchange to send their secret number across their line of communication. First, Alice and Bob agree on a public key, the prime number p and a primitive root of p, a. Then, both of them decide on their own, a positive integer private key of which to raise the secret number to; Alice chooses k_1 and Bob chooses k_2. With the private and public keys being determined, Alice and Bob encrypt a using the public key and their respective private keys

$$(a^{k_1}) \bmod p = b_1, (a^{k_2}) \bmod p = b_2$$

and they send their encrypted numbers to eachother. Once each of them have recived the number from the other, then they the number on their end using the same public key and respective private keys

$$(b_1^{k_2}) \bmod p = c, (b_2^{k_1}) \bmod p = c$$

At the end, they both have a shared key c; this works since

$$(b_1^{k_2}) \bmod p = ((a^{k_1})^{k_2}) \bmod p = ((a^{k_2})^{k_1}) \bmod p = (b_2^{k_1}) \bmod p$$

5 Diffe-Hellman and Groups

5.1 Generalized Problem

The following proof is based from The Diffe-Hellman Key Exchange Protocol and Non-Abelian Nilpotent Groups (Mahalanobis).

In this section, the Diffie-Hellman Key Exchange will be generalized and show that both Abelian and non-Abelian groups can be used to encrypt the key.

Let G be a cyclic group of order n and let ϕ,ψ be the automorphism of G to G. In the key exchange, the parties selects an element 'a' from the coset $G/Z(G)$. Then party 1 selects an automorphism $\phi_A \in$Aut(G) and sends ϕ_A(a) to party 2. Then party 2 selects an automorphism $\psi_B \in$Aut(G) and sends it to party 1.

Afterwards, party 1 calculates

$$\psi_B(\phi_A(a))$$

and party 2 calculates

$$\phi_A(\psi_B(a))$$

with

$$\phi_A(\psi_B(a)) = \psi_B(\phi_A(a))$$

Thus, the Diffie-Hellman key exchange can be described as a group of automorphisms of group G. In it, the private keys can be taken from ϕ_A and ψ_B where $\phi_A = xax^{-1}$ and $\psi_B = yay^{-1}$ where the private keys in the exchange are 'x' and 'y'.(Mahalanobis, 6-9) The difficulty of breaking the key exchange relies on the Conjugacy Problem. That is, to find out if there is a z such that y = zxz^-1.(Mahalanobis, 6-9) In this problem, the Conjugacy Problem is easily solvable, meaning z can be retrieved depending on the group. To solve this problem, the Diffie-Hellman Key Exchange should use a group that is undecidable for the conjugacy problem. Abelian finite groups solve this problem, such as the group of integers. However, non-abelian groups can be used, such as Braid Groups.(Computational Complexity And The Conjugacy Problem)

5.2 Conjugacy Problem and Braid Groups

Braid groups are composed of elements made of braids and an group operation of composition of braids.Importantly, however, Braid Groups are undecidable for the conjugacy problem. (Paris, 4-7) Conjugacy problem is defined as determining whether two elements in a group are conjugate elements in G. In the previous group, the messages can be decoded since ϕ_A and ψ_B are known.
Let G be a Braid Group and let g_1 and g_2 be elements of the group. To solve the conjugacy problem, $g_2 = gg_1g^{-1}$ must be found for a conjugator $g \in G$. However, since the conjugacy class is infinite, the intrusive third party cannot go through every conjugate in the set to determine g. Additionally, the problem is solvable for B_3; so for the Diffie-Hellman Key Exchange to be secure, $n > 3$ for B_n. Thus conjugacy problem is computational hard to break, and thus avoids the problem an integer group might have. (Harlander, et al)

5.3 Decrypting The Message Proof

The Following Proof is based from Diffie-Hellman Key Exchange (University of North Texas, 1-3).

Let G be a finite cyclic group with element $< p >$ generating the group. The secret key in the group is $k = p^{a^b}$ which is equal to $p^{a^b} = p^{b^a}$. With a message m, the encrypted message is $m_{encrypted} = mk$. With the current information, it appears that one would need to know both a and b to decrypt the message. However, using the properties of Groups, it can be shown that only one key is required to decrypt the message.

First rewrite $(p^a b)^-1$

$$(p^{ab})^{-1} = p^{-ab} = 1^a p^{-ab} = (p^{|G|})^a p^{-ab} = p^{a|G|-b}$$

Since p is the generator the group, $p^{|G|}$ is the identity and thus can be multiplied by the inverse of key. Thus the inverse can be rewritten as $p^{a|G|-b}$

Then multiply $mp^{ab} * (p^{-ab})$

$$mkk^{-1} = mp^{ab}(p^{-ab}) = mp^{ab}(p^{a^{|G|-b}}) = m(p^{a^{b+|G|-b}}) = m * (p^{a^{|G|}}) = m * 1 = m \qquad (1)$$

Thus, a user does not need to know both private keys to decrypt the message. Instead, they only need to know the order of the group, their own private key, and the public key sent from the other user. And, because solving p^a is computationally hard, encrypted messages are secure.

6 Conclusion

The Diffie-Hellman Key Exchange can be generalized to automorphisms within a group. To ensure the key exchange is secure, it is important that private keys cannot be extracted from either automorphism, thus ensuring that the conjugacy problem for the group is computationally hard. With a certain group, depending on the chosen public key, it can be

computationally easy to break. Using a group that has been proven to be hard to break the the conjugate ensures security. While finitely generated abelian groups uphold this property, non-abelian groups such as Braid Groups also work.

7 Bibliography

Gallian, Joseph A. Contemporary Abstract Algebra. 9th ed., Cengage Learning, 2017.

Harlander, Jens, et al. Conjugacy Problem - Theory And Application. Boise State University, math.boisestate.edu/reu/publications/ConjugacyPoster.pdf.

Kowalczyk, Chris. One-Way function. Crypto-IT, 22 Nov. 2017, www.crypto-it.net/eng/theory/one-way-function.html.

Mahalanobis, Ayan. The Diffie-Hellman Key Exchange Protocol And Non-Abelian Nilpotent Groups. Cornell University Library, arxiv.org/pdf/math/0602282.pdf.

Miasnikov, Alexei, and Paul Schupp. Computational Complexity And The Conjugacy Problem. Cornell University Library, arxiv.org/pdf/1605.00598.pdf.

Paris, Luis. Braid Groups and Artin Groups. Cornell University Library, arxiv.org/pdf/0711.2372.pdf.

Rosen, Kenneth H. Discrete mathematics and its applications. 7th ed., McGraw-Hill, 2013.

Diffe-Hellman Key Exchange. Diffie-Hellman key exchange, University of Cincinati, 9 Jan. 2015, gauss.ececs.uc.edu/Users/Franco/Project/dh.htm.

Diffie-Hellman Key Exchange. University of North Texas, Computer Science and Engineering at University of North Texas, www.cse.unt.edu/ tarau/teaching/PP/NumberTheoretical/Diffie%E2%80%93Hellman%20key%20exchange.pdf.