Diffie-Hellman Anahtar Değişimi

SÜMEYYE SOYLU

Hacettepe Üniversitesi Matematik Bölümü, 4. Sınıf Öğrencisi

sumeyyesoylu1101@gmail.com

Öncelikle biraz bilgisayar çağı öncesine dönelim; hatta yazının icat edildiği zamana kadar gidebiliriz. Çünkü gizli bilgilere sahip olma kavramının uygarlığın başlangıcına kadar dayandığı düşünülmektedir. Savaşlarda, politik entrikalarda veya kişisel mesajlarda bilgilerin yalnızca istenilen kişi tarafından anlaşılması her zaman büyük önem taşımıştır. Bu amaçla geliştirilen kriptografi, yani şifreleme bilimi, binlerce yıllık bir geçmişe sahiptir.

Örneğin Roma İmparatorluğu'nda Julius Caesar, mesajlarını gizli tutabilmek için her harfi alfabede belirli bir sayıda kaydırarak başka bir harf ile değiştirdiği basit bir şifreleme yöntemi kullanmıştır. Mesela her harf 3 birim sağa kaydırıldığında, şifreli metni çözmek isteyen kişinin de her harfi 3 birim sola kaydırması gerekir. Burada 3 sayısı anahtar olarak kullanılır ve mesajın doğru şekilde çözülebilmesi için bu anahtarın alıcı tarafından bilinmesi şarttır. Eğer bu bilgi (anahtar) gizli tutulmazsa, şifreleme amacına ulaşmaz ve ayrıca anahtar olmadan şifre çözülemez, çünkü alıcı her harfi kaç birim geri kaydıracağını yalnızca anahtar sayesinde bilebilir.

O zamandan itibaren uygarlıklar, harflerin başka harf veya sembollerle değiştirilmesine dayanan yerine koyma şifreleri (örneğin Caesar şifresi) ve harflerin yerlerinin değiştirilmesine dayanan yer değiştirme şifreleri yöntemlerini geliştirerek şifreleme tekniklerini çeşitlendirmiştir. Kullanılan tekniklerin karmaşıklık seviyesi ise dönemden döneme ve ülkeden ülkeye farklılık göstermiştir. Örneğin, Amerika Birleşik Devletleri 1917'de I. Dünya Savaşı'na girerken, İtalya'da 1600'lü yıllarda kullanılan şifrelerden bile daha basit yöntemler kullanıyordu.

Tarih boyunca, 20. yüzyıla kadar geliştirilen şifreleme yöntemlerinin tamamı, bugün simetrik şifreleme olarak adlandırdığımız yapıya dayanıyordu. Zamanla bu yöntemler, polialfabetik şifreleme gibi daha karmaşık tekniklerle gelişmiş, 20. yüzyılda ise II. Dünya Savaşı sırasında kullanılan Enigma gibi mekanik rotor makinelerinden, günümüzde kullanılan modern simetrik şifreleme yöntemleri olan blok ve akış şifreleme algoritmalarına evrilmiştir.

Simetrik şifrelemede hem şifreleme hem de şifre çözme işlemi için aynı gizli anahtar kullanılır. Bu anahtar, algoritmanın en önemli bileşenidir. Peki neden bu kadar önemli olduğunu bir örnekle açıklayalım, çünkü temel mantığı her zaman aynıdır. Örneğin blok şifrelemede, mesaj önce belirli uzunluktaki parçalara bölünür ve her parça, anahtarın belirlediği karmaşık matematiksel işlemlerden geçirilerek şifrelenir. Çözme algoritması ise aynı anahtarı kullanarak bu işlemleri tersine çevirir ve orijinal mesajı ortaya çıkarır. Burada asıl önemli olan, tüm matematiksel dönüşümlerin anahtara bağlı olmasıdır. Anahtar değiştiğinde, algoritmanın tüm adımlarındaki matematiksel işlemler de değişir. Bu nedenle yanlış anahtar kullanan biri, ortaya çıkan karışıklıktan hiçbir anlamlı bilgi elde edemez.

Tabii simetrik şifrelemenin büyük bir sıkıntısı vardır; en büyük sorun anahtarın güvenli bir şekilde paylaşılmasıdır.

İşte bu soruna çözüm arayışıyla birlikte, 1976 yılında Whitfield Diffie ve Martin Hellman, **"New Directions in Cryptography"** başlıklı makalelerini yayımladıklarında, kriptografi dünyasında yeni bir dönem başlamış oldu [1]. Çünkü Diffie-Hellman protokolü, tarihte ilk kez yayımlanan asimetrik algoritma olarak, daha önce mümkün olmadığı düşünülen bir şeyi başardı: İki tarafın, önceden hiç karşılaşmadan ortak bir gizli anahtar oluşturabilmesini sağladı. Bu gelişme, asimetrik kriptografinin diğer adıyla açık anahtarlı kriptografinin doğuşuna zemin hazırladı.

Asimetrik şifreleme sistemlerinde her kullanıcının bir açık anahtarı ve bir de gizli anahtarı bulunur. Açık anahtar herkesle paylaşılabilirken, gizli anahtar yalnızca sahibinde kalır. Gönderici, alıcının açık anahtarını kullanarak mesajı şifreler ve bu mesajı yalnızca alıcı, kendi gizli anahtarıyla çözebilir. Böylece güvenli iletişim için önceden gizli anahtar paylaşımı yapmaya gerek kalmaz.

Burada not olarak belirtmek gerekir ki, ele alacağımız Diffie-Hellman bir asimetrik şifreleme algoritması değil, bir anahtar paylaşım protokolüdür.

Şimdi, anahtar değişim problemini daha iyi kavrayabilmek için bir senaryo hayal edelim. Daha önce hiç iletişim kurmamış iki tarafın, aralarındaki iletişimi gizli tutarak haberleşmek istediklerini varsayalım. İletişim kanalı üzerinde bir saldırganın bulunabileceği ihtimalini bilerek, taraflar mesajlarını saldırgan tarafından çözülemeyecek ve tamamen okunamaz hâle getiren bir şifreleme sistemi kullanmaktadır.

Ancak bu şifreleme sisteminde, gönderici mesajı şifrelemek için bir anahtar (gizli bilgi) kullanmakta ve alıcının da bu şifreli mesajı çözebilmesi için aynı anahtara sahip olması gerekmektedir. Göndericinin bu anahtarı güvenli olmayan iletişim kanalı üzerinden doğrudan alıcıya iletmesi durumunda, saldırganın anahtara erişme ve böylece şifreli mesajı çözme riski ortaya çıkar.

İşte bu noktada, taraflar arasında güvenli bir şekilde ortak bir anahtarın paylaşılması sorunu, yani anahtar değişim problemi, karşımıza çıkmaktadır.

1976 yılında Whitfield Diffie ve Martin Hellman tarafından önerilen **Diffie-Hellman Anahtar Değişimi**, anahtar dağıtım problemlerine bir çözüm sunar. Yani, iki tarafın daha önce birbirleriyle tanışmadan, güvensiz bir kanal üzerinden ortak bir gizli anahtar üzerinde anlaşmasını sağlar. Bu sayede, ortak üretilen anahtar ile gönderici mesajını şifreleyip güvensiz ağ üzerinden alıcıya iletebilirken, alıcı da aynı anahtarı kullanarak bu şifreli mesajı çözebilir.

Diffie-Hellman anahtar değişim protokolü, p bir asal sayı olmak üzere, mod p altında tanımlı sonlu cisim $\mathbb{Z}_p$'nin sıfır dışı elemanlarının oluşturduğu çarpımsal bir grup olan $\mathbb{Z}_p^*$ üzerinde tanımlanmıştır. Burada

$$\mathbb{Z}_p = \{\overline{0},\ \overline{1},\ \overline{2},\ \ldots,\ \overline{p-1}\}, \quad \mathbb{Z}_p^* = \{\overline{1},\ \overline{2},\ \ldots,\ \overline{p-1}\}$$

şeklindedir.

$(\mathbb{Z}_p^*, \cdot)$ sonlu grubu, abelyen (değişmeli) ve devirli bir gruptur. Devirli oluşuna açıklık getirelim:

Bir $g \in \mathbb{Z}_p^*$ için, eğer

$$\{g^1 \pmod p,\ g^2 \pmod p,\ \ldots,\ g^{p-1} \pmod p\} = \mathbb{Z}_p^*$$

eşitliği sağlanıyorsa, yani $\mathbb{Z}_p^*$ kümesinden aldığımız bir eleman g'nin mod p altındaki $g^1, g^2, \ldots, g^{p-1}$ kuvvetleri, $\mathbb{Z}_p^*$'deki tüm elemanları hiçbir tekrar olmadan veriyorsa, bu durumda g sayısına mod p için bir **ilkel kök**, $\mathbb{Z}_p^*$ grubuna da devirli grup denir.

Şimdi bu protokolün işleyiş sürecini ele alalım. Tarafları gönderici olarak Ayşe ve alıcı olarak Burak şeklinde adlandıralım. Ardından, taraflar arasında anahtar değişiminin nasıl gerçekleştiğini adım adım inceleyelim.

Adım 1: Taraflar, herkes tarafından bilinebilecek şekilde yeterince büyük bir asal sayı p ve mod p altında ilkel bir kök g üzerinde anlaşırlar.

Adım 2: Gönderen (Ayşe), rastgele bir a tam sayısı seçer ve bu değeri gizli tutar. Alıcı (Burak), rastgele bir b tam sayısı seçer ve o da bu değeri gizli tutar.

Adım 3: Ayşe $A \equiv g^a \pmod p$ ifadesini hesaplar ve bu değeri Burak'a gönderir. Burak ise $B \equiv g^b \pmod p$ değerini hesaplayıp Ayşe'ye gönderir.

Adım 4: Ayşe, Burak'tan kendisine gelen B değerini kullanarak $B^a \pmod p$ hesaplamasını yapar. Burak ise Ayşe'den aldığı A değerini kullanarak $A^b \pmod p$ işlemini gerçekleştirir.

4. adımın sonunda Ayşe, $B^a \equiv g^{ba} \pmod p$ hesaplamış, Burak ise $A^b \equiv g^{ab} \pmod p$ hesaplamış oldu. $ab = ba$ olduğundan her iki taraf da aynı gizli anahtara ulaşmış olur. Artık bu anahtar, daha sonra simetrik şifreleme sistemlerinde kullanılabilir.

Somutlaştırarak anlatmak gerekirse:

Burak, yeni bir yemek bloguna abone olur. Bu blogda, dünya çapındaki şefler gizli tariflerini takas etmek amacıyla iletişim kurmaktadır. Bir gün Burak, blog üzerinden "AyseChef06" adlı kullanıcıdan bir mesaj alır.

Ayşe, Akdeniz mutfağına ait çok özel bir sos tarifini paylaşmak istediğini, ancak blogun mesajlaşma sistemine güvenmediğini belirtir.

Bunun üzerine Burak, bu sorunu birlikte çalışarak çözebileceklerini söyler. Önerisi şudur: Blogun herkese açık kanalını kullanarak, sadece kendilerinin bileceği ortak bir gizli anahtar oluştursunlar. Daha sonra bu anahtar, bir simetrik şifreleme sisteminde (örneğin AES, 3DES) kullanılarak, sos tarifi güvenli bir şekilde şifrelenebilir. Böylece tarifin içeriği blog üzerinden iletilse bile, üçüncü şahıslar tarafından okunamaz hâle gelecektir.

Şimdi bu süreci nümerik bir örnekle açıklayalım:

Adım 1: Burak, herkese açık blog kanalında iki bilgi paylaşır: Asal sayı $p = 71$ ve $g = 7$ ilkel kök üzerinde anlaşalım.

Adım 2: Ayşe gizli tam sayı olarak 6 değerini seçiyor ve Burak 60'ı seçiyor.

Adım 3:Ayşe $A \equiv 7^6 \pmod{71} \equiv 2$ değerini hesaplıyor ve Burak'a gönderiyor.
Burak $B \equiv 7^{60} \pmod{71} \equiv 30$ değerini hesaplıyor ve Ayşe'ye gönderiyor.

Adım 4: Ayşe, Burak'tan gelen $B = 30$ değerini kullanarak: $K \equiv 30^6 \pmod{71} \equiv 45$ hesaplıyor. Burak ise Ayşe'nin gönderdiği $A = 2$ değerini kullanarak: $K \equiv 2^{60} \pmod{71} \equiv 45$ hesaplıyor.

Yani, $K = 45$ ikisinin de kullanacağı ortak gizli anahtar olur.

Peki, bu iletişimi izleyen bir kulak misafiri neden ortak anahtara ulaşamaz? Bu sorunun yanıtını açıklayalım.

Diffie-Hellman anahtar değişiminin güvenliği, *ayrık logaritma problemi*ne dayanır ve bu problemin çözülmesi zor olduğu düşünülmektedir.

$\mathbb{Z}_p^*$ kümesinde tanımlı **Ayrık Logaritma Problemi** şu şekilde ifade edilir:
p asal sayı olmak üzere, $\mathbb{Z}_p^*$ grubunda bir ilkel kök g ve bir sayı h verildiğinde,

$$g^x \equiv h \pmod{p}$$

eşitliğini sağlayan x tam sayı değerine, h'nin g tabanına göre **ayrık logaritması** denir.

Burada asıl mesele şudur: Eğer g, x ve p biliniyorsa, h'yi bulmak kolaydır. Ancak g, h ve p biliniyorken x'i bulmak yeterince büyük sayılar için oldukça zordur. Bu probleme **ayrık logaritma problemi** adı verilir.

Şimdi protokole geri dönelim. Burada kulak misafiri, herkese açık olan asal sayı p ve ilkel kök g gibi bilgilere sahiptir. Adım 3'te Ayşe'nin Burak'a ve Burak'ın da Ayşe'ye açık ağ üzerinden direkt gönderdikleri $A \equiv g^a \pmod{p}$ ve $B \equiv g^b \pmod{p}$ değerlerini okusa da, burada Ayşe ve Burak tarafından gizli tutulan a ve b değerlerini bulmak için yukarıda tanımlanan ayrık logaritma problemini çözmesi gerekir.

Eğer saldırgan bu problemi çözebiliyorsa, yani tarafların gönderdikleri $A \equiv g^a \pmod{p}$ ve $B \equiv g^b \pmod{p}$ ifadelerinden a ve b'yi çıkarabilirse, bu durumda Ayşe ve Burak'ın ortak gizli anahtarı olan $g^{ab} \pmod{p}$ değerini hesaplaması kolay olur ve böylece Ayşe ile Burak'ın paylaştığı ortak anahtara erişebilir.

Verilen nümerik örneğe bakıldığında, $7^6 \pmod{71} \equiv 2$, $7^{30} \pmod{71} \equiv 30$ eşitliklerinde ayrık logların hesaplanması, küçük asal sayılar ile çalışıldığından deneme-yanılma yöntemiyle kolaylıkla gerçekleştirilebilir.

Bununla birlikte, ayrık logların hesaplanması için: Yaklaşık 20 veya daha az ondalık basamağa sahip asal sayılar için çalışan Bebek-Adım Dev-Adım yöntemi, 20 veya daha fazla ondalık basamağa sahip asal sayılar için ise İndeks Hesabı yöntemi tercih edilebilir.

Ancak gerçek hayattaki Diffie-Hellman uygulamalarında kullanılan asal sayı p, genellikle 2048 bit (yaklaşık 617 basamak) uzunluğunda, ilkel kök g, 2, 3, 5 veya 7 gibi küçük sayılar seçilirken, gizli tutulan a ve b de en az 256-512 bit uzunluğunda seçilmektedir. Bu durumda, ayrık logaritmanın hesaplanması için klasik bilgisayarlar üzerinde her ne kadar algoritmalar geliştirilmiş olsa da, kullanılan asal sayıların oldukça büyük olması nedeniyle bu algoritmalar üstel zaman gerektirmektedir ve dolayısıyla klasik bilgisayarlar üzerinde makul sürede hesaplama yapılamamaktadır.

Diğer yandan, kuantum bilgisayarlar için geliştirilen Shor algoritmasının ayrık logaritma problemini logaritmik zamanda çözebileceği teorik olarak gösterilmiştir [2]. Ancak mevcut teknoloji düzeyinde kuantum bilgisayarlar, büyük asal sayılar üzerinde işlem yapabilecek yeterli kapasiteye sahip değildir. Yine de teorik olarak mümkün görülmesi, gelecekte Diffie-Hellman protokollerinin kuantum tehditlere karşı savunmasız kalabileceğini göstermektedir. Bu doğrultuda, kuantum tehditlere karşı dirençli kriptografik sistemlerin geliştirilmesine yönelik çalışmalar kuantum sonrası (post-quantum) kriptografi alanında devam etmektedir.

■ Kaynaklar

[1] Diffie, W., & Hellman, M. (1976). *New Directions in Cryptography*. *IEEE Transactions on Information Theory*, 22(6), 644–654. https://doi.org/10.1109/TIT.1976.1055638

[2] Shor, P. W. (1994). *Algorithms for Quantum Computation: Discrete Logarithms and Factoring*. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science* (pp. 124–134). IEEE. https://doi.org/10.1109/SFCS.1994.365700

[3] Stallings, W. (2017). *Cryptography and Network Security: Principles and Practice* (7th ed.). Pearson.

[4] Paar, C., & Pelzl, J. (2010). *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer.

[5] Hoffstein, J., Pipher, J., & Silverman, J. H. (2008). *An Introduction to Mathematical Cryptography*. Springer.

[6] Smart, N. (2016). *Cryptography: An Introduction* (3rd ed.).

[7] Wood, D. (2021). *Real-World Cryptography*. Manning Publications, Shelter Island.

[8] Kraft, J., & Washington, L. C. (2015). *Kriptografi ile Sayılar Teorisine Giriş* (Çev. A. Yavuz). Palme Yayıncılık.

[9] Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996). *Handbook of Applied Cryptography*. CRC Press.