Veri Şifreleme Teknikleri Kriptoloji (Cryptology)

Kriptoloji (Cryptology)
Kriptolojinin temel olarak ayrıldığı iki dalı vardır: Kriptografi(Cryptography) ve Kripto analiz(Cryptanalysis).
Kriptoloji basit anlamda şifreli belgeler, gizli yazılar bilimidir. İki ayrı ana dalı vardır: Kriptografi ve Kripto analiz. Kriptografi, metin şifrelemekte kullanılan tekniklerin tümünü inceleyen bilim dalıdır. Kripto analiz, ise şifrelenmiş bir metnin şifrelenmemiş düz metin haline getirebilmek için yapılabilecek her türlü saldırıları ve türlerini inceleyen bilim dalıdır. Kaba tabiriyle, kriptografi bilimi ile şifrelenen metinler, kripto analiz bilimi ile kırılmaya çalışılır.

Kriptolojinin tarihçesine baktığımızda uzun yıllar geriye gitmemiz gerekmektedir. M.Ö. 1900'lü yıllarda firavunların mezarlarındaki yazıtlarda kullanılan semboller bilinen ilk kriptografik dönüşümlerdir. Daha çok sıkı askeri disiplinleri ve talimleriyle tanınan Kuzey Yunanistandaki tarihi bir şehir olan Sparta' da M.Ö. 475 yılında bilinen ilk kriptografik iletişim aracı 'skytale' geliştirilmiştir. M.Ö. 60'lı yıllarda ise Julius Ceaser askeri anlamda şifreli metinleri kullanan ilk kişidir. Daha sonraki makalelerimizde daha detaylı olarak anlatacağımız Ceaser şifrelemesi o zamanlara dayanmaktadır. Bilinen en eski bilimsel kripto analiz eseri, ünlü Yunan bilim adamı alKalkashandi tarafından 1412 yazılmıştır. Ardından birinci dünya savaşı yıllarına kadar bu alanda pek bir gelişme kaydedilmemiştir. 1917'de Edward Hugh Hebern tarafından Rotor machine adı verilen bir şifreleme aracı geliştirilmiştir. 1971'de de IBM tarafından Lucifer adı verilen şifreleme şeması oluşturulmuştur. 1975'te DES, 1976'da da Diffie and Hellman şifreleme teknikleri geliştirilmiştir. Burada ismi geçen şifreleme yöntemleri daha sonraki makalelerimizde çok daha detaylı olarak işlenecektir.

Kriptografiyi temel olarak üç ayrı sınıfa ayırmak mümkün. Birincisi düz metni belirli şifreleme algoritmaları çerçevesinde geri dönüşümü olacak şekilde şifrelemektir. Harflerin yerlerinin değiştirilmesi ya da her harfin başka bir harfle değiştirilmesidir. Bir örnek ile açıklamak istersek, DOTNET kelimesini TTEDON ve ya EPUOFU şeklinde şifreleyebiliriz. İkinci yöntem olarak da, gönderen ve/veya alıcı tarafından bilinen anahtarlar kullanılarak metinleri şifreleyebiliriz. DES, IDEA vs.. Üçüncü yöntem olarak da metinleri bloklar halinde ve bir bütün olarak şifreleyebiliriz. Block Cipher ve Stream Cipher.

Kripto analizinde kullanılacak stratejiler kullanılan şifreleme planına ve elimizdeki bilgilere göre değişmektedir. Aşağıdaki tabloda şifrelenmiş metin üzerine yapılabilecek saldırı çeşitleri listelenmiştir:

Saldırı türü	**Kripto analiz için bilinenler**
Sadece Şifrelenmiş Metin	Şifreleme algoritması Deşifre edilecek şifreli metin
Şifrelenmemiş Metin	Şifreleme algoritması Deşifre edilecek şifreli metin Bir ya da daha fazla şifrelenmemiş metin örneği çifti
Belirli Şifrelenmemiş Metin	Şifreleme algoritması Deşifre edilecek şifreli metin Şifreli metin ve bu metne karşılık gelen şifrelenmemiş metin örneği
Belirli Şifrelenmiş Metin	Şifreleme algoritması Deşifre edilecek şifreli metin Şifrelenmiş ve deşifre edilmiş anlamlı metin
Belirli Metin	Şifreleme algoritması Deşifre edilecek şifreli metin Şifrelenmiş ve deşifre edilmiş anlamlı metin Şifreli metin ve bu metne karşılık gelen şifrelenmemiş metin örneği

Kriptolojiden bahesedip de İkinci Dünya Savaşının kaderini değiştiren Enigma' dan bashetmemek olmazdı. Rotor Machine prensibine dayanan Enigma, Alman Arthur Scherbuis tarafından geliştirilmiş bir şifreleme aracıdır. Kauçuk, lastik gibi elektriği iletmeyen bir maddeden yapılan Rotor bir buz hokeyi diski boyutlarındadır. Bu disk veri girişi ve çıkışı için iki ayrı levhadan oluşmuştur. Hem veri girişi levhası hem de veri çıkışı levhasında diski çevreleyen eşit olarak bağlanmış 26 bağlantısı vardır. Veri girişi levhasındaki 26 bağlantı rotorun veri çıkışı levhasındaki 26 bağlantı noktasına rastgele bağlanmaktadır. Makinenin tüm gizemli kısmını da zaten rastgele oluşturulmuş bu bağlantılar sağlamaktadır. İkinci Dünya Savaşında İngilizler Almanların enigma ile şifrelenmiş belgelerini çözümleyemediler. Bir İngiliz baskıyla bir enigma aracının İngilizlerce bulununcaya kadar enigmanın sırrı çözülemedi. Enigma ile şifrelenmiş bilgileri kırmayı başaran İngilizler Almanların şüphelenmemesi için bir sonraki Alman saldırısına izin verdiler. Almanlar ne olduğunu dahi anlayamadan İngilizler savaşı lehlerine çevirmeyi başardılar. Bu da kriptolojinin ve şifreleme mekanizmalarının önemi üzerine sizlere bir fikir verebilir.

Burada bahsedilmesi gereken önemli bir diğer terim de "Güvenli Sistemler" dir. İki çeşit güvenli sistem vardır:

-Koşulsuz Güvenli Sistemler(Unconditionally Secure Systems)
-Matematiksel olarak Güvenli Sistemler(Computationally Secure Systems)

Eğer şifrelenmiş metin, boyutu ne olursa olsun, şifrelenmemiş metni oluşturmak için yeterli bilgiyi içermiyorsa bu tür sistemler Koşulsuz Güvenli Sistemler olarak adlandırılmaktadır. Şifrelenmiş metni kırmanın maliyeti şifrelenmemiş metnin değerinden fazlaysa ve/ve ya şifrelenmiş metni kırmak için gereken zaman bilginin geçerlilik süresinden fazlaysa da, bu tür sistemler Matematiksel olarak Güvenli Sistemler olarak adlandırılır.

Ayrıca şunu da belirtmekte yarar var ki, bu ve bundan sonraki makalelerimizde geçecek olan düz metinden (plain text) kastımız, şifrelenmemiş metindir.

Klasik Şifreleme Teknikleri

Döküman mahiyetinde olan bu yazıda klasik şifreleme tekniklerini anlatmaya calıştım.

Steganography(Metni Gizleme): Sözlüklerde tam karşılığını bulamayacağınız kadar eski bir kelime olan David Kahn “The Story of Secret Writing” adlı eseriyle bugünkü anlamını kazanmıştır: Şifrelenmemiş düz bir metni çeşitli dönüşümler kullanarak diğer kişilerce anlaşılamaz bir metin haline getirilmesi işlemidir. Verilebileecek en basit örnek, bir metnin tüm harflerinin başka bir metnin içindeki kelimelerin ilk harflerine gizlenmesidir. Örneğin; “Sezen Aksu ve aşk şarkıları benim için tüm tesellilerden iyidir.”

Ancak her durum için böyle bir mesajı oluşturmak zor ve zaman alıcı bir işlemdir. Oluşturulacak mesajdaki kelimelerin anlamlı bir bütünlük oluşturacak şekilde bulunmasının ancak çok üst düzey bir yapay zeka uygulamasıyla bilgisayar ortamına aktarılması mümkündür. Hala daha “natural language processing” (bilgisayarların konuşma dillerini anlayabilmesi olarak basitçe tanımlanabilir) konusunda mükemmel bir sisteme ulaşılamamış olması kadar önemli bir nokta da kırılmasının çok kolay olmasıdır. Tarihçesine göre çeşitli teknikler kullanılmıştır. Bazıları:

Character marking(Harf işaretleme): İsminden de anlaşılabileceği üzere, önceden belirlenmiş bazı karakterlerin daha derin (daha koyu değil) olarak yazılması mantığına dayanır. İşaretli harfler ancak belirli bir açıyla parlak bir ışığa tutulduğunda görülebilir.
Invisible ink (Görünmeyen mürekkep): Belirli bir sıcaklığa ulaşmadan ya da kimyasal bir işlemden geçirilmeden görülemeyen özel bir mürekkep ile yazılır.
Pin Punctures (İğne delikleri): Belirli harflerin üzerinde gözle görülemeyecek kadar küçük delikler açılarak gerçekleştirilir. Deliklerle işaretlenmiş araçlar parlak bir ışığa tutularak okunur.
Her ne kadar bu teknikler çok eski gibi gözükse de steganography tekniğinin çağdaş örnekleri de bu tekniklerin eşleniği durumundadır. Çağdaş örneği olarak bir CD deki bilgilerin her byte’inin son bitini kullanarak şifreleyebiliriz. Örneğin oluşturulabilinecek en büyük çözünürlükteki 32 bitlik resmlerin kaydedildiğ bir CD’yi ele alalım. Burada biraz teknik bilgi verelim. Bir resmin 32 bit olması; her pixelinin 32 bitlik renk kodlarıyla gösterilmesi demektir. En sonuncu bitin değişmesi resmin genel içeriğini ve renklerin ihmal edilebilir ölçülerde değiştirir. Bu sayede 700 MB’lık bir CD’de yaklaşık 12 MB’lık bilgi saklayabilirsiniz. Buna benzer bir şifreleme çok yakın bir zamanda kullanılmıştır.

Substitution techniques(Yerine koyma teknikleri): Düz bir metindeki harflerin yerine başka harflerle, sayılarla ya da semboller koyarak yapılan şifreleme türüdür. Eğer düz metin ardışık bitler olarak görülebiliyorsa, çeşitli bit patternlerinin(örneklemlerinin) değiştirilmesi de bu tekniğe dahil edilebilir.

Caesar Cipher(Sezar Şifrelemesi), bilinen en eski yerine koyma tekniğidir. Ünlü Roma İmparatoru Julius Caesar tarafından geliştirilmiştir. Sezar şifrelemesinde mantık her harfi kendisinden sonra gelen üçüncü harfle çembersel olarak değiştirmeye dayanmaktadır. Örneğin;

düz metin: “Bilgisayarların şifreleri kırıldı”
şifrelenmiş metin: “DLOİUÖÇBÇTOÇTKÖ ÜLHTĞOĞTL NKTKOGK”

Sezar şifrelemesi 3 önemli zayıflığı vardır. Şifrelenmiş metinden hangi dilin kullanıldığı rahatlıkla anlaşılabilir. Türkçe için düşündüğümüzde sadece 28 ayrı şifreleme geliştirilmiş olabilir. Şifreleme ve deşifreleme algoritmalarının biliniyor ve kolaylıkla uygulanabiliyor olması da diğer zayıf olduğu yönüdür. Sezar şifrelemesi ile şifrelenmiş bir metin “Brute Force” bir saldırı ile kırılabilir. Brute Force, kelime anlamı olarak kaba kuvvet demektir. En zayıf ama en kesin saldırı yöntemidir. Sezar şifrelemesi gibi algoritmaların bilindiği yöntemlerde olası bütün kombinasyonların denenmesi demektir. Daha sonraları çokça bahsedeceğimiz “Brute Force” saldıraların nasıl gerçekleştirileceği ve bu saldırılardan nasıl korunabileceğimizi kodlarıyla birlikte açıklayacağız. Şimdilik 28 ayrı şifrelemenin de denerek düz metnin ele geçirilmesi olarak bilmeniz yeterli olacaktır.

Sezar şifrelemesindeki 28 ayrı şifrelemenin yetersizliği açıktır. monoalphabetic Ciphers(Tekli alfabeye dayanan şifreleme), Sezar Şifrelemesindeki mantığı biraz daha geliştirilerek, her harfe rastgele bir harfin eşleştirilmesi mantığına dayanmaktadır. Yani A ile eşlenen harf arasındaki ilişki diğer hiç bir harf arasında yer almamaktadır. Bunun basit bir örneğini her haftasonu gazetelerin ilavelerinde verdikleri “Şifreli bulmacalar” da görmekteyiz. Orada harfler yerine rakamlar verilmektedir. Çözülebiliriliğini arttırmak için şifrelenmemiş metin üzerine bir kaç örnek ve şifrelenmemiş metnin içeriği hakkında bilgi verilmektedir.

monoalphabetic Cipher'lar ile Türkçe için "29!<st1:metricconverter ProductID="-1"" w:st="on">-1"</st1:metricconverter> ayrı şifreleme söz konusudur. İlk bakışta çok güvenli izlenimi yaratmasına karşın monoalphabetic cipherlar da kırılabilmektedir. Hala daha şifrelemenin yapıldığı dil rahatlıkla tespit edilebiliyor. Şifrelemenin yapıldığı dil üzerine istatiksel bir araştırma yaparak kısa sürede kırmanız mümkündür. Her dilde belirli karakterlerin tekrarlanma frekansları belirlidir. Elimizdeki şifrelenmiş metinde en çok tekrarlanan karakterler ile şifrelemenin yapıldığı dildeki en yüksek frekanslı karakterleri eşleştirdiğinizde tahmin ettiğinizden çok daha kısa bir sürede kırabilirsiniz. Zaten eşleştirmelerden sonra elimizdeki şifreli metnin haftasonu gazetelerinde verilen bulmacalardan pek bir farkı kalmıyor.

monoalphabetic tekniğinin biraz daha geliştirmenin bir diğer yolu da şifreleme boyunca farklı "yerine koyma teknikleri" kullanmaktır. Bu tür yaklaşımların genel adı Polyalphabetic Ciphers(çoklu alfabeye dayanan şifreleme)' dır. Bu tür şifreleme mekanizmalarının çoğunda iki ortak özellik vardır:

1. Önceden tanımlanmış bir monoalphabetic yerine koyma kuralları kümesi vardır,

2. Bir anahtar verilen dönüştürme işlemi için hangi kuralın kullanılacağını belirler.

Multiple-Letter Encryption(Çoklu Harfle Şifreleme): Verilen düz metindeki ikili harflere tek bir birim olarak ele alan ve işleyen şifreleme tekniğine dayanan Playfair multiple-letter encryption için en çok bilinen yöntemdir. Digraph(tek sesi temsil eden iki harf) olarak alınan harfler yine digraph olarak şifrelenir. Playfair algoritması ingilizce için özelleşmiş bir şifreleme algoritmasıdır. 5x5' lik bir matris kullanılarak yapılır. Playfair, İngiliz bilim adamı 1854'te Sir Charles Wheatstone tarafından ilk defa ortaya atılmıştır. Ancak Playfair'e destek veren ve savunan Baron Playfair of St. Andrews ismiyle anılmıştır. Playfair ile ilgili yapacağınız hemen hemen her araştırmada karşınıza çıkacak klasik ve açıklayıcı bir örnek vardır. Bu örnek de Lord Peter Wimsey tarafından Dorothy Sayer's Have His Carcase adlı eserinden alınmıştır.

Yukarıda verilen örnekte anahtar kelime "monarchy" dir. Matris anahtar kelimede geçen harflerin dışındaki harflerin soldan soğa ve yukarıdan aşağıya doğru sırayla yazılmasıyla oluşturulmuştur. I ve J harfleri bir harf olarak sayılmıştır. Metin her seferinde iki karakter alınarak ve aşağıdaki dört kural uygulanarak şifrelenirler.

1. Tekrarlanan karakterler önceden kabul edilmiş bir doldurma karakteriyle birbirinden ayrılırlar. Kural olarak olmasada genel olarak X karakteri kullanılır. Örneğin, "balloon" kelimesi "ba lx lo on" şeklinde ikililere ayrılarak şifrelenecektir.

2. Matriste aynı satıra düşen düz metin karakterleri için kural, her harfi bir sağındaki harf ile değiştirmektir. Örneğin, AR digraphı, RM olarak şifrelenecektir. (Matristeki çemberselliğe dikkat edin).

3. Matriste aynı sutüna düşen düz metin karakterleri için kural, her harfi bir altındaki harf ile değiştirmektir. Örneğin, MU digraphı, CM olarak şifrelenecektir. (Matristeki çemberselliğe dikkat edin).

4. Eğer yukarıdaki üç koşuldan hiç biri de karşılanmıyorsa, her düz metin karakteri, kendisiyle aynı satırdaki, eşleniği karakterle aynı sutündaki harf ile değiştirilir. Örneğin, HS digraphı BP olarak, EA digraphı da IM/JM olarak şifrelenecektir.

monoalphabetic şifrelemelere göre çok daha güvenilirdir. Her şeyden önce, monoalphabetic şifrelemede, (İngilizce için) sadece 26 harf varken, burada (İngilizce için) 26x26=676 ayrı digraph vardır, bu sayede tek bir digraphın çözülmesi biraz daha zorlaşmıştır. Bununla beraber karakter tekrarlanma frekansının oluşturulacak şifreli metni etkilemez. Bu sebeplerden ötürü, uzun süre playfair'ın kırılmasının imkansız olduğu düşünüldü. I. Dünya Savaşında İngiliz Ordusu ve II. Dünya Savaşında da U.S. Ordusu ve müttefikleri tarafından kullanılmıştır. Bu kadar güvenli olduğu sanılan Playfair şifrelemesinin kırılması aslında tahmin edilen çok daha kolaydır. Şifrelenmemiş düz metnin dili ve yapısı hakkında bir çok bilgiyi şifrelenmiş metinden elde edebiliyoruz. Genellikle bu şifreleme tekniği ile şifrelenmiş metinlerin kırılması için bir kaç yüz karakterlik şifreli bir metin yeterli olmaktadır. Oysa bir şifreleme tekniğinin tam anlamıyla güvenli bir şifreleme tekniği olabilmesi için ne kadar şifrelenmiş metin olduğunun önemi olmamalıdır. Elimizde on binlerce karakter şifrelenmiş metin olsa bile, bu bizim şifreyi kırmamıza yetmemelidir.

Not: Her ne kadar playfair sadece İngilizce için geliştirilmiştir dediysek de, Türkçe için de aynı yöntemi uyarlamak mümkündür. Ancak bununla ilgili hiç bir resmi kaynakta belgelenmiş bir yazı bulunmamaktadır. O yüzden Türkçe için yoktur diyebiliriz, ancak yapılamaz diyemeyiz.

Encryption Şifreleme Tekniği

System.Security.Cryptography namespace'i güvenli şifreleme, deşifreleme ve diğer güvenlik hizmetlerini gerçekleştiren hashing, rasgele sayı üretimi ve mesaj doğrulama gibi şifreleme servislerini içerir. Daha önceki makalelerimizde bahsettiğimiz gibi şifreleme genel olarak;

-Gizlilik(Confidentiality)
-Veri Bütünlüğü(Data Integrity)
-Kimlik Doğrulama(Authentication)
güvenlik servislerini sağlamak için kullanılır. .NET ile sağlanan şifreleme tekniklerini 4 ayrı gruba ayırmamız mümkündür:

Simetrik Şifreleme (symmetric cryptography): Private-key encryption olarak da bilinir. Her iki tarafın da bildiği tek bir ortak anahtar kullanarak şifrelemeyi ve deşifrelemeyi gerçekleyen şifreleme teknikleridir. DES, RC2, Rijndael ve TripleDES(3DES) Cryptography namespace'i altında yer alan şifreleme teknikleridir. Bu şifreleme tekniğinin en önemli dezavantajı her iki tarafın tek bir anahtar üzerinde anlaşması ve sadece bu anahtarı kullanarak şifreleme/deşifreleme işlemini gerçeklemesidir. Bu teknikte hiç bir taraf, karşı tarafın gerçekten karşı taraf olup olmadığını bilememektedir. Hatta öyle ki iki tarafda olması gereken kişiler olmayabilir. Genellikle simetrik şifreleme diğer şifreleme tekniklerinde transfer edilecek anahtarlar gibi her iki taraf içinde ortak ifadelerin şifrelenmesi gerektiğinde kullanılır.
Asimetrik Şifreleme (asymmetric cryptography): Public-key encryption olarak da bilinir. İkili anahtar kullanarak şifreleme ve deşifrelemenin gerçeklendiği şifreleme tekniğidir. Bu şifreleme tekniğinde herkesin 2 anahtarı vardır. Bunlardan biri public'tir, yani herkesce bilinir. Diğeri ise private'dır ve sadece şifrelemeyi gerçekleyen tek bir taraf tarafından bilinir. Bu iki anahtarın rasgele seçilmiş iki anahtar şeklinde değildir. İkisi de birbirini bütünleyen iki anahtar şeklindedir. .NET Framework DSA ve RSA algoritmalarını kullanmaktadır.
Cryptographic signing: Verinin gerçektende belirli bir kişi ya da gruptan geldiğinin anlaşılması için tarafların verileri kendi imzaları ile şifrelemesi prensibine dayanır. Bu işlem hash fonksiyonlarını da kullanmaktadır. Veriyi gönderenin ve alanın gerçekten de kendilerinin olduğunu kanıtlamalarını sağlar. Gönderen kendi private anahtarı ile şifrelerken gönderenin public anahtarını bilen herhangi biri veriyi gönderenin gerçekten de gönderen olup olmadığını anlayabilir ancak alıcıdan başka kimse gönderilen bilgiyi deşifre edemez. Yine .NET Framework DSA ve RSA bu tür şifreleme teknikleri için kullanmaktadır. Bir önceki şifreleme tekniği ile kullanım ayrılıklarını daha sonraki makalelerimizde detaylı olarak irdeleyeceğiz.
Cryptographic hashes: Herhangi bir boyuttaki bir bilgiyi sabit uzunluktaki bir byte dizisiyle eşleştirir. Hashler istatiksel olarak tektir(unique), yani iki farklı byte diziliminin aynı hash değerine sahip olamaz. Hash işlemi genellikle tek yönlü fonksiyonlarla gerçeklenir. Tek yönlü fonksiyonlar, matematikteki tersi alınamayan fonksiyonlardır. Örnek olarak bir sayının belirli bir sayıya göre modunun alınmasını verebiliriz. 25 ve sayısının mod 10 daki karşılığı 5 iken mod 10 daki karşılığı 5 olan sayı sadece 25 değildir. Yani şifrelenmiş metni kaybettik! Geri dönüşümüz yoktur. Verdiğimiz örnekten de dikkat edeceğiniz üzere iki farklı sayı aynı hash değerine karşılık geldi. Oysa demin farklı iki byte diziliminin aynı hash değerine sahip olamayacağını söylemiştik. Bu noktada kendimizle çelişiyor gibi olabiliriz. Ama günümüzde kullanılan hiç bir hash fonksiyonu verdiğimiz örnekteki kadar basit ve dar alanlı değildir. HMACSHA1, MACTripleDES, MD5, SHA1, SHA2, SHA3, SHA5 .NET Framework'te yer alan bu tür şifreleme teknikleri örneklerindendir.
Yukarıda saydığımız şifreleme tekniklerinin .NET ile gerçeklenmesini bu yazı dizimizin ilerki makalelerinde bulabilirsiniz. Yine şifreleme teknikleri kadar önemli bir noktada "Random Number Generation" rasgele sayı üretimidir. .NET Framework RNGCryptoServiceProvider sınıfı ile bu sorunun üstesinden gelmeyi bilmiştir.

Şimdiye kadar bahsettiğimiz şifreleme tekniklerine ek olarak çok basit anlamda çalışan kendi şifreleme tekniğimizi geliştirebiliriz. Sizlere daha önceki makalelerimizde bahsettiğimiz şifreleme tekniklerinden Ceaser şifreleme tekniğini gerçekleyen ve kullanan bir web servisi uygulamasını geliştirelim.

Ceaser Cipher

Örnek Ceaser Cipher uygulamamız için öncelikle bir web servisi uygulaması başlatalım. Bu ve bundan sonraki örneklerimiz bizim kullanacağımız isim konvansiyonu doğrultusunda web servisimize SYCryptologyServices adını verelim. Ve bu web servisi projesine "CeaserCipher.asmx" isimli bir web servisi oluşturalım. Aşağıdaki kod parçaçıklarını da sırasıyla kodumuza ekleyelim;

```
private byte CeaserCipherIncrement=2;

[WebMethod]
public string CeaserCipherEncrypt(string plainText) {
StringBuilder sb=new StringBuilder();
for (int i=0;i<plainText.Length;i++)
{
byte b=(byte)plainText[i];
b+=CeaserCipherIncrement;
sb.Append((char)b);
}
return sb.ToString();
}
```

Eklediğimiz bu fonksiyon kendisine gönderilen plainText'in her karakterinin ASCII karşılığının değerine 2 ekleyerek plaintext'i şifrelemektedir. Hatırlayacağınız üzere Ceaser Cipher sadece 26 harf üzerinden çalışıyor ve toplamı 26'yı aşan karakterlerin 26'ya göre modu alınarak şifreleme işlemi gerçeklenmekteydi. Boşluk sayı vb özel karakterlerinde şifrelenmesine olanak tanımak için biz ASCII karşılığına 2 ekledik. Mod'unu almak yerine değişkenimizi byte tanımladık. Bildiğiniz üzere byte [0..255] aralığındaki sayıları ifade etmektedir. Toplamı 255 sınırını aşan sayılarda mod alma işlemi programlama dili tarafından otomatik yapılmaktadır. şimdi de şifrelenmiş bu metni deşifreleyecek web fonksiyonunu geliştirelim:

```
[WebMethod]
public string CeaserCipherDecrypt(string encryptedText) {
StringBuilder sb=new StringBuilder();
for(int i=0;i<encryptedText.Length;i++)
{
byte b=(byte)encryptedText[i];
b-=CeaserCipherIncrement;
sb.Append((char)b);
}
return sb.ToString();
}
```

Eklediğimiz bu fonksiyon kendisine CeaserCipherEncrypt fonksiyonu ile şifrelenmiş olarak gönderilen metnin her karakterinin ASCII karşılığının değerinden 2 çıkartılarak plaintext'i elde etmektedir. Yukarıda da bahsettiğimiz gibi, Ceaser Cipher'dan farklı olarak 256 karakter üzerinden çalışan bir web metodu geliştirmiş olduk. Herhangi bir metni bu web servislerini kullanarak şifreleyebilir ve deşifreleyebiliriz. Bunları web servis olarak gerliştirerek herkesin her uygulamasından bu fonksiyonları kullanmasına olanak tanımış olduk. Örneğin bu web servisini bir "Managed C++ console" uygulamasından bile çağırabilirsiniz. Bu tür basit ve sistemi çok fazla yormayan bir şifreleme tekniğini, kullanıcıların log on olmadan girdiklerinde sitenizden yararlanamamaları için kullanabilirsiniz. Bunun için yapmanız gereken ASP.NET kullanıcıya gönderilen tüm response'i bu metotları kullanarak şifrelemek ve deşifrelemektir.

Bu yazımızda System.Security.Cryptography namespace'i altında yer alan şifreleme tekniklerinden bahsettik. Bundan sonraki yazılarımızda bu şifreleme tekniklerinin her birini detaylı olarak inceleyeceğiz ve her birine ilişkin bir örnek uygulama geliştireceğiz. Ayrıca bu yazımızda Ceaser şifreleme tekniğine benzer bir şifreleme ve deşifreleme gerçekleştiren bir web servisi geliştirdik. Siz de daha önceleri sizlerle paylaştığımız şifreleme teknikleri için benzer uygulamalar geliştirebilirsiniz.

alıntıdır (security)