Internet üzerinde mahremiyet ve kullacının mahremiyetini kendisine karşı güvence altına alan bir yazılım pratiği: Hohha anlık mesajlaşma servisi

İsmail Kizir[1]

[1] Hohha Internet Services Ltd. Teknolojiden Sorumlu Başkanı, Siyaset Bilimci

ikizir@gmail.com

Özet: Haberleşme özgürlüğü ve mahremiyeti modern dünyada tüm demokratik devletlerin anayasalarının olmazsa olmaz maddelerinden biridir[i].
Dünyada, kamuoyunun gündeminde gittikçe daha fazla yer eden “internette mahremiyet” konusunu kısaca özetledikten sonra, “Hohha anlık mesajlaşma” uygulaması ve bu uygulamanın üstüne bina edildiği özgün algoritmalar ele alınarak, kişisel mesajların mahremiyetinin hangi yöntemlerle güvence altına alındığı işlenecektir.
Söz konusu yazılım için tasarlanan ve MIT lisansı ile kamuya açılan, anahtarın da şifrelenen metin ile değiştiği, özgün “Hohha Dinamik XOR” algoritmasının çalışma mantığı, avantajları ve zaafları detaylı bir şekilde masaya yatırılacaktır.
Anahtar Sözcükler: Mahremiyet, şifreleme, Hohha Dinamik XOR, simetrik şifreleme, RSA, anlık mesajlaşma yazılımları, Hohha.
Abstract: Freedom and privacy of communication is one of the most crucial parts of all modern democratic constitutions. In this paper, we will briefly explain various aspects of the popular “privacy on Internet” subject, present our Hohha Instant Messaging Service and the methods it uses to ensure user's privacy.
We will also explain in detail our MIT Licensed “Hohha Dynamic XOR encryption algorithm” which is dynamically updating the key according to plaintext during encryption.

1. Internet üzerinde mahremiyet

Bazı inançlarda, Tanrı dahil, adı bilinen her varlığın üzerinde hakimiyet kurulabileceğine, bundan dolayı, Tanrı'nın kimsenin bilmediği gizli bir ismi olduğuna inanılır.
Binlerce yıl geçmesi, bu inancı gülünç kılmak bir yana, seküler anlamda güçlendirdi: Yukarda bahsettiğimiz inancın teaffuzunu bile “şeytan işi” addedenler başta olmak üzre, güç peşinde koşan herkes, yeni çağın seküler dininin bir numaralı virdini her fırsatta tekrar eder ve uygular oldu: “Bilgi güçtür”!
Günümüzde ister sanal ekonomi olsun, ister reel ekonomi, “bilgi”nin en büyük değer olduğu tartışmasız bir gerçek.
Bu tartışmasız gerçekten dolayı, özellikle “yeni ekonomi” şirketlerinin hemen hemen tümü, bilginin her türlüsüne erişebilmek için korkunç bir rekabet içinde.
Bu rekabet, öyle boyutlara ulaşmış durumda ki, milyon hatta milyar dolar maliyetli hizmetlerin, “ücretsiz” verilmesi sıradanlaştı, hatta giderek “olmazsa olmaz” bir hal aldı.
Bir zamanlar, "ücretsiz", "karşılıksız", "borç" kavramları tüm dünyada şüphe ile karşılanırken, yeni ekonomideki dönüşümle, kitleler, milyar dolarlık maliyetlerin nasıl karşılandığını pek de umursamadan ücretsiz verilen tüm servisleri büyük bir iştahla kullanmaya başladı.
Zamanla, özellikle “ücretsiz” servislerin, hem de, bırakın özel şirketleri; vakıf adı altında çalışarak en idealist görünenlerinin verdiklerinin bile, hiç de öyle “karşılıksız” olmadığını görür olduk[ii].
Daha kötüsü, Hegel'in, biraz da kaçamak yollu, “Tanrı'nın Yeryüzündeki Yürüyüşü”[iii] olarak adlandırdığı devlet, hiç sahip olmadığı kadar fazla ilahi vasfa sahip oldu. “Teoride” kendi rızamız ile, her noktada kameralar ile izleniyor, cep telefonlarımız sayesinde, attığımız her adımı takip “ettiriyoruz”[iv]! Üstelik bir tane de değil. Herbiri her an diğeri ile savaş halinde olabilen devletlerimiz, dolayısı ile “Tanrılarımız”, her devlet içinde de, “burada Tanrı ben olacağım sen olacaksın

kavgasındaki Tanrıcıklarımız" var. Ve her yaptıklarını bizlerin, yani "kullarının" iyiliği için yapıyorlar! Herşeyi biliyor, görüyor ve yargılıyorlar. Tüm bu "koruma önlemlerine" gerekçe olarak, biz kulların da başından tehdit ve tehlike hiç eksik olmuyor!

Bu durumun, özellikle 2011 yılında Wikileaks kurucusu Julian Assange'ın Time Dergisi'ne kapak olmasından sonra, en azından, konunun kamuoyu önünde tartışılması açısından değişmeye başladığını düşünüyoruz. 2013-2015 yılları arasında, pek çok ülke yönetiminin yabancı servisler tarafından dinlendiğinin ortaya çıkması[v] ve bunun devletler düzeyinde "sükut" ile doğrulanması; 2013 yılı öncesine de uzanan Wikileaks belgelerinde geçen kişisel yazışmalar ve bunların nasıl elde edildiği ve son olarak 2015 yılı sonlarında, CIA Direktörü'nün bile, görece önemsiz bazı iş dokümanlarının kişisel mesajlaşması için kullandığı bir e-posta servisi üzerinden ele geçirilmesi tartışmaları tekrar alevlendirdi: CIA direktörü bile olsanız, "kişisel" mahremiyete; dolayısı ile bu mahremiyeti güvence altına alacak, pratik olarak kullanabileceğiniz araçlara muhtaçsınız! Pratik diyoruz, zira, CIA Direktörü de, şüphesiz, PGP[vi]'nin varlığından ve bu araç sayesinde mahrem yazışmalarını RSA şifreleri ile ücretsiz servisler üzerinde bile koruyabileceğinden haberdardı. Fakat, gündelik hayatta kullanılabilecek kadar pratik değil. Ücretsiz e-posta hizmet veren şirketler de "nedense", isteseler, mevcut sistemlerine entegre edebilecekleri bu teknolojilerin bahsini bile açmıyorlar.

Kamuoyunda bir kıpırdanma olsa da, geniş toplum kesimlerinin yeterli bir bilinç düzeyine ulaştığını söylemek zor. Mahremiyet hakkında, Medya ile yaratılan algı, sadece yasadışı işler çevirenlerin mahremiyet peşinde koşacaklarıdır. Türk medyasında da, "beni dinleseler ne olur? Saklayacak birşeyim yok!" temalı köşe yazıları görülmemiş şey değildir. Sıradan insanlar gözünde, "beni dinleseler ne olur?" mottosu(virdi), gizli bir övünme ile(vecd) söylenen, adeta, devletine(Tanrısına) sadakat sunma ritüeli(ibadeti) haline gelmiştir.

Gelinen noktada, geniş kitleler, mahremiyet konusunu, tek boyut ile, sadece kriminal boyut ile ele almakta, diğer üç boyutu ıskaladıklarından dolayı da, haberleşmelerinin üzerindeki mahremiyet örtüsünün sadece "kendi devletleri" tarafından kaldırılacağını, onların da bu bilgileri sadece "iyilik" için kullandığını düşünüp, bu örtünün kaldırılmasına razı olmalarını beyan etmenin bir erdem olduğunu düşünmektedir.

Biz, mahremiyet konusunu 4 boyutta değerlendiriyoruz.

Kriminal boyut: Her türlü suç faaliyetinin, yetkili otoriteler tarafından, iletişimin dinlenmesi yoluyla, yasalar çerçevesinde takibi, karşı çıkılabilecek bir durum değil. Zaten bu husus, tüm demokratik devletler tarafından yasalara ve bilebildiğimiz kadarıyla, tümünde, mahkeme kararına bağlanmıştır. Fakat, kriminal aktivitelerin varlığı, kriminal aktivitelere karıştığı ispatlanmamış bireylerin en temel anayasal hakkı olan "haberleşmenin gizliliği"nin rafa kaldırılması için bahane teşkil etmez.

Meraklı gözler: Kişinin mahrem bilgisinin devletler dışındaki yerli ya da yabancı oluşumlar tarafından ele geçirilmesi olasılığı. Çevredeki meraklı bir komşudan, kıskanç bir kocaya, ya da hacker şebekelerine kadar geniş bir yelpazenin, yasal olmayan yollardan, haberleşmeyi dinlemesi ve şantaj dahil çeşitli amaçlar için kullanmasıdır. Bu, sadece haberleşmenin şifrelenmesi ile aşılabilecek bir durumdur.

Ticari şirketlerin elinde "biriken" mahrem bilgi: Özellikle, ücretsiz servislerin elinde biriken bilgi ile ne yapıldığı ve bu bilgilerin, devletlerin de dahil olduğu diğer kişi&kurumlara karşı nasıl korunduğu en önemli meselelerden biridir. CIA direktörünün bile, ücretsiz bir serviste "biriktirilen" kişisel mesajları çalınarak medyaya sızdırılabilmiştir[vii]. Bu bilgilerin, kişilik profili çıkartılmak için kullanılması ve bu kişilik profilinin en masum satış artırma arzusundan, ciddi kriminal aktivitelere kadar geniş bir yelpazede kullanılması da olası. Üstelik, söz konusu şirket sunucuları ile yürütülen haberleşmenin şifrelenmesi, sadece meraklı gözlere karşı bir önlem olmaktan öteye gidemiyor ve bu sorunu çözmüyor. Devletler bile, kendilerine emanet edilen en

mahrem bilgileri korumakta “korkutucu” derecede yetersiz kalmaktalar. Türkiye'de MERNİS'de kayıtlı tüm bilgilerin üç beş kuruş veren tüm kişilere, hem de kapıda pazarlama yöntemi ile satılması, kara mizah olsa gerek: Kanun temcilcisi pek çok avukatın el altında bu programdan bulundurduğu herkesin bildiği bir “sır”dır[viii].

Yabancı devletlerin elinde biriken mahrem bilgi: *Mahrem bilginin yasal ya da gayrimeşru yabancı oluşumların eline geçmesi birinci dereceden bir ulusal güvenlik zaafıdır.* Burada, onlarca yıl boyunca biriktirilen verilerden bahsediyoruz. Özellikle, “bulut mimarisi” ve “büyük veri” kavramları ile ön plana çıkartıldığı üzre, bugün, en azından teoride, rahatlıkla, “Internet üzerinde özel ya da kamuya açık her harfin depolandığını” söylemek mümkündür.

Dahası, bu, bireylerin rızası ile gerçekleşiyor. Kolaylık ve pratiklik adına, tüm e-postalar. hatta, anlık mesajlar depolansın isteniyor. Şirketler de, “hangi telefondan hangi bilgisayardan girerseniz girin farketmez, verilerinizi senkronize ediyoruz” diyerek bunun reklamını yapıyorlar. Bu verilerin, yabancı devletlerin eline geçmediğini ya da geçmeyeceğini de kimse garanti edemiyor. *Bir çocuğun daha 8-10 yaşından itibaren yazdığı her mesaj, her e-posta, dolayısı ile, hayatının tüm detayları, bir daha hiç silinmeyecek şekilde depolanıyor. Başka ülkeden MİT müsteşarı, Cumhurbaşkanı ve Genelkurmay başkanı ithal edemeyeceğimize göre, bir gün, "tüm bu hayatı depolanan" çocuklarımızdan biri doğal olarak Cumhurbaşkanı ya da MİT müsteşarı, diğeri Genelkurmay başkanı olacak ve “ufacık çocuğu dinleseler ne olur” diyerek geçiştirdiğimiz tüm o veriler korkunç bir ulusal güvenlik zaafiyeti olarak karşımıza çıkacak. Kişilik analizinden, şantaj malzemesine kadar her türlü bilgi her an kullanıma hazır şekilde depolanmaya devam ediyor*!

2. Kullacının mahremiyetini kendisine karşı güvence altına alan bir yazılım pratiği: Hohha anlık mesajlaşma servisi

Bu bölümde, yukarda anlatıların ışığında, mahremiyeti güvence altına almak üzre tasarladığımız bir anlık mesajlaşma yazılımından bahsetmek istiyoruz: Hohha anlık mesajlaşma yazılımı.

Kullanılan yöntemlerin daha iyi anlaşılabilmesi için, genel olarak şifreleme teorisine değinmemiz gerekiyor.

Simetrik ve asimetrik şifreleme

Simetrik şifreleme, aynı anahtarın hem şifreleme, hem de şifrelenen metni açmak için kullanıldığı algoritmalara verilen isimdir. Fakat, simetrik bir şifreleme kullanabilmek için, kimsenin eline geçmeyeceğini garanti altına alarak, şifreleme anahtarının alıcıya iletilebilmesi gerekir ki, bu, ya eski zamanlardaki gibi, alıcı ile yüzyüze görüşerek yapabileceğiniz, ya da Internet gibi ortamlarda, ancak, "asimetrik" şifreleme kullanarak gerçekleştirebileceğiniz bir işlemdir.

Asimetrik şifreleme, şifreleme için ayrı, şifrelenen metni açmak için ayrı anahtarların kullanıldığı, açık anahtar(public key) denilen, herkesin eline geçmesinde hiçbir sakınca olmayan anahtarın sadece şifreleme için kullanılabildiği, fakat, bu anahtar kullanılarak yapılan şifrelemenin de, ancak ve ancak bu anahtar ile birlikte oluşturulmuş ve şifreyi çözecek olandan başka kimsenin elinde bulunmaması gereken "mahrem anahtar"(private key) ile çözülebildiği şifreleme yöntemidir.

Şimdiye kadar oluşturulan, en azından bizim bildiğimiz tüm asimetrik şifreleme yöntemleri, simetrik şifrelemeye göre onlarca/yüzlerce hatta binlerce kat yavaştır. Bundan dolayı, standart web güvenlik protokollerinin tümü asimetrik şifrelemeyi sadece simetrik şifreleme anahtar değişimi için kullanır ve bundan sonrasına simetrik şifreleme algoritmaları ile devam ederler. Bu anahtar değişim işlemi, söz konusu algoritmayı geliştirenlerin isimlerinden hareketle Diffie-Hellman[ix] anahtar değişimi olarak adlandırılır

XOR şifrelemesi

Bilinen en hızlı ve kolay simetrik şifreleme algoritmalarından biri XOR şifrelemesidir[x].

Aynı fonksyon ve aynı anahtarın hem şifreleme hem de şifrelenen veriyi açmak için kullanılması ve fonksyonun basitliği genelde, şifrelemeye giriş dersinde ilk anlatılan algoritmalardan biri olmasını sağlar.

Çok sağlam bir temele, kolay anlaşılabilir ve kolay uygulanabilir olmasına rağmen, klasik XOR şifreleme bazı temel zayıflıklar barındırdığından dolayı yalın haliyle gerçek hayatta pek tercih edilmez. **Fakat, teorik olarak bu algoritmanın, şifrelenen metin uzunluğuna eşit ve gerçekten rastgele seçilmiş bir şifreleme anahtarı ile kullanıldığında, kırılmasının imkansız olduğu kabul edilmiştir.**

Şifreleme anahtarında 0 bulunması durumunda, şifrelenen karakterin elde ediliyor olması, şifreleme anahtarındaki karakter ile şifrelenen karakterin aynı olması durumunda ise, 0 değeri elde ediliyor olması, şifrelemenin kırılmasını büyük ölçüde kolaylaştırır. En kötüsü de, şifrelenmiş metni bir kısmının ya da tamamının bilinebildiği durumlarda, anahtarın ilgili kısımlarının zahmetsizce çözülebilmesidir.

Şifreleme anahtarının kısa olması durumunda ve şifrelenen metindeki belirli örüntülerin tespit edilebilmesi durumunda da zayıf bir algoritmadır. Çok uzun şifreleme anahtarları seçilmesi de, depolama ve veri iletişiminde artış gibi başka pratik zorlukları beraberinde getirmektedir.

Fakat yukarda bahsettiğimiz zayıflıklar giderilebilirse, XOR, kırılması en zor şifreleme algoritmalarından biri haline gelebilir. Algoritmik karmaşıklığı O(N) olmasından dolayı yoğun işlem yapan uygulamalarda ideal bir kullanım sağlayabilir. Üçüncü bölümde, Hohha için özel geliştirdiğimiz ve MIT Lisansı ile kamuya açtığımız "Hohha Dinamik XOR" algoritması ele alınacaktır.

Hohha RSA[xi] ve Hohha Dinamik XOR ile mahremiyetinizi güvence altına alır.

Birinci bölümde anlatılanlardan yola çıkarak, Internet üzerinde mahremiyet konusunda, çözülmesi gereken iki temel problem olduğunu söylemek mümkündür: Birincisi, bilgimizin, servisi veren şirket dışındakilere karşı güvence altına alınması, ki, bu sorun, artık, standartlaşacak derecede kolay bir şekilde çözülmüştür. Standart hale gelen SSL protokolü sayesinde, sunucu ve istemci arasındaki haberleşme kolayca şifrelenebiliyor ve haberleşmeyi "dinleyen"lere karşı ciddi bir engel oluşturuyor. Fakat, bilgimizin "servis veren şirkete karşı" korunması, hemen hiç bahsedilmeyen bir durum. Hohha, bu iki sorunu birden, yani mahremiyetin hem üçüncü kişi ve kurumlara, hem de kendisine karşı korunmasını temin etmek amacıyla geliştirildi.

Hohha ilk çalıştırıldığı her cihazda, sadece o cihazda kullanmak üzre, 4096 bit'lik bir RSA anahtar çifti oluşturur[xii].

Hohha giriş sunucusu da, günümüzde finans kurumları ve devletlerin kullandığı, 4096 bit RSA anahtar çifti kullanmaktadır. Sunucu açık anahtarı istemci yazılımın içinde gömülü gelmekte ve böylece aradaki adam saldırısı(Man-in-the-middle attack)[xiii] olasılığı devredışı bırakılmaktadır. Üstelik, her yazılım güncellemesinde bu anahtar değiştirilebilmekte ve ekstra bir güvenlik sağlanmaktadır. **https://letsencrypt.org/** adresindeki girişime destek verilmesi de düşünülmekte.

İstemci ile sunucu arasındaki ilk haberleşme bahsi geçen gömülü 4096 bit anahtarı ile yapılır. Kullanıcı, sunucuya, hem kendi açık RSA anahtarını hem de sunucunun açık RSA anahtarı ile şifrelenmiş bir biçimde giriş doğrulama bilgilerini, oluşturduğu simetrik şifre anahtarını ve bu anahtarın gerçek sahibi olduğuna dair bir imza gönderir. Bunun karşılığında da, sunucu, eğer giriş bilgileri doğrulanmış ise, kullanıcının açık RSA anahtarı ile şifrelenmiş biçimde, geriye, dağıtık modelin gereği olarak, simetrik şifrelenmiş biçimde oturum belirtecini(Session identifier) ve kullanıcının kalıcı TCP ya da WebSocket bağlantısı kuracağı sunucunun ip adreslerini gönderir. İstemci ilgili sunucuya bağlanır, belirlenmiş simetrik anahtar ile şifreli biçimde giriş bilgilerini gönderir ve bundan sonrası Hohha DinamikXOR şifrelemesi ile devam eder. Şimdiye kadar anlatılan, sadece, Hohha dışındaki gözlere koruma sağlar.

Hohha, hiçbir mesajı depolamaz. Her mesaj alıcısına ulaştırıldığı anda sistemden silinir.

Tüm kullanıcılar, bire bir sohbet ettikleri tüm diğer kullanıcıların açık RSA anahtarlarına(public keys) sahip olurlar. Fakat, sistemdeki kimse, sunucu dahil, kimsenin mahrem anahtarına(private key)

sahip değildir.
Bir kullanıcı, diğer bir kullanıcının açık anahtarına sahip değilse, bunu, sunucudan ister ve ayrıca, yanında gelen imza kontrolü ile de gerçekten kapalı anahtarın sahibinden gelip gelmediğini de kontrol eder.
Daha sonra, sıra, sadece bu iki kullanıcı arasında kullanılacak Hohha Dinamik XOR ortak simetrik anahtar belirlemeye gelir.
Mesajlaşmayı ilk başlatan kullanıcı, anahtarı oluştururarak, ve bunu diğer tarafın açık RSA anahtarı ile şifreleyerek sunucu vasıtası ile, karşı tarafa gönderir. Alıcı, sadece kendi mahrem RSA anahtarı ile açılabilecek bu bilgiyi deşifre eder ve bundan sonrasında bu iki kullanıcı arasındaki tüm ikili yazışmalar bu anahtar ile şifrelenir. Hohha, kişilerin yüzyüze ortak Hohha Dinamik XOR şifre anahtarı belirlemelerine de olanak tanımaktadır, ki, tavsiye edilen kullanım şekli budur. Bu anahtarların uzun süre kullanımının, anahtarı tamamen değiştirmeden güvenliğini sağlamak için ise, belirli aralıklarla otomatik olarak anahtarın "orjinal tuzlama değeri " değiştirilmektedir.
Dosya transferlerinde, her dosya için daha geniş gövdeli, sadece o dosyaya özel bir anahtar oluşturulur.
Dahası, güvenliği artırmak amacıyla, kullanıcıların diledikleri anda, RSA anahtarlarını değiştirmelerine, ya da farklı cihazlarda farklı RSA anahtarları kullanmalarına olanak tanınır. Alıcının açık anahtarı ile şifrelenen her sistem kontrol kodu ya da mesajın yanında, mesajın şifrelendiği açık anahtarın bir parmak izi(checkum) ile birlikte gönderilmesiyle gerçekleşir. Alıcı, kendisine gelen mesajın o an kullandığı anahtar ile şifrelenip şifrelenmediğini anlamak için, o an kullandığı anahtarın parmak izi ile, kendisine gönderilen parmak izini karşılaştırır. Eğer, bu parmak izleri farklı ise, ya anahtarı değiştirmiş, ya da farklı bir cihazdan farklı bir anahtar ile giriş yapmış demektir. Sunucuya, mesajın gönderen tarafından tekrar şifrelenip gönderilmesine dair bir komut gönderir. Söz konusu komut aynı zamanda, alıcının yeni anahtarını da içerir. Bu komutu alan sunucu program, komutu mesajı gönderene yönlendirir. Mesajı gönderen kullanıcının cihazında çalışan program, yeni anahtarla şifrelemeyi yaparak mesajı tekrar alıcıya yönlendirir.
Yıllık 5 Amerikan Doları karşılığı ücret ile piyasaya sürülmesi planlanan ve dileyen üyelerle MIT lisansı ile kaynak kodu paylaşılacak olan Hohha'nın, başka pek çok özelliği vardır ve sıradan bir mesajlaşma yazılımından çok daha fazlası olmak için tasarlanmıştır. Fakat, bu bildirinin amacı Hohha Platformu'nu tanıtmaktan çok, mahremiyet sorununu nasıl ve hangi yöntemlerle çözdüğünü anlatmaktır. Bundan dolayı, Hohha'nın özelliklerinden bahsetmeyi kesiyor ve kendi geliştirdiğimiz "Hohha XOR" simetrik şifreleme algoritmasını anlatacağımız üçüncü bölüme geçiyoruz.

3. Hohha Dinamik XOR şifrelemesi

Klasik XOR yönteminde, n byte(karakter) uzunluğunda bir şifreleme anahtarı olur, ve şifrelenecek verinin her m'inci karakteri, şifreleme anahtarının (m modulus n)'inci karakteri ile XOR işlemine tabi tutularak şifreli metin elde edilir. Şifreli metin tekrar tıpatıp aynı işleme tabii tutulduğunda ise, metin deşifre edilmiş olur. Temeli sağlam, kolay anlaşılır, kolay uygulanır ve hızlıdır. Günümüzde Amerilan Hükümeti'nin de gizli belgelerde kullandığı simetrik şifreleme metodu AES[xiv] de XOR temellidir.
Biz, klasik yöntemin zaaflarını ortadan kaldırmak için aşağıdaki yöntemleri uyguluyoruz:
1. Algoritma bir adet anahtar kullanıyor.
2. Anahtar oluşturan fonksyona girdi olarak zıplama sayısı(Z) ve 2'nin herhangi bir kuvvetine eşit olması gereken gövde uzunluğu(G) veriliyor. Bu fonksyon, önce bu G adet rastgele sayı ve tuzlama(iv,salting,nonce) amacıyla, 8 bayt rastgele sayı oluşturuyor. Bu sayılar, şifreleme teorisinde, özellikle anahtarın birden fazla kullanılacağı durumlarda aynı verinin her şifrelenişinde farklı bir çıktı vermesi için kullanılan ve şifrelemeye rastgelesellik katan bir tekniktir. Şifrelemede genelde Nonce ya da IV olarak adlandırılır. Biz, çeşitli sebeplerden dolayı tuzlama terimini tercih ettik.
3. Sonuçta, bu veriler birleştirilerek, (1+2+8+G) bayt uzunluğunda, en baştaki ilk bayt Z'ye eşit, sonraki 2 bayt gövde

uzunluğu , sonraki 8 bayt tuzlama verisi ve kalan bayt'ların da gövdeyi oluşturduğu tek bir anahtar elde ediliyor.

4. Şifre ya da deşifre işlemi başladığında, anahtarı oluşturan tüm baytların 32 bitlik denetim toplamı hesaplanır (CRC).

5. Anahtarın kullanıldıkça kendini açığa vurmaması ve tek bir seferde şifrelenen metin içinde de yapılan analizlerde kendini açığa vurmaması(leaking) için, mümkün olan en fazla rastgeleselliği yakalamamız gerekiyor. Şifreleme boyunca, oluşacak şifrelenmiş çıktının en rastgele şekilde oluşmasını amaçlıyoruz. Bu amaç için dinamik olarak değişen dört farklı değişken kullanıyoruz: M gövdede bulunduğumuz pozisyonu; Salt1 ve Salt2 her şifrelemede sadece o şifrelemeye özel olarak üretilen rastgele tuzlama değerlerinin 4'er baytından oluşturulan iki rastgele 32 bitlik sayıyı; V ise, başta anahtar CRC toplamına eşit bir sayıy temsil ediyor. Ve elbette anahtar gövdemiz var ve onu da dinamik olarak güncelliyoruz. Her zıplamada, tüm bu değişkenleri mümkün olduğunca yerinde kullanarak, mümkün olan en fazla rastgeleselliğe sahip bir değerler zinciri oluşturmalıyız. Z sayıda zıplama yapacağız.

6. İlk zıplamada, Salt1'in alçak 8 bit değerini gövde M'in işaret ettiği pozisyondaki bayt ile XOR ediyoruz. Bu gövde değerini ise, Salt2'nin alçak 8 biti ile XOR ediyoruz. M değerini de Salt2 ile XOR ederek zıplamayı gerçekleştiriyoruz. Sonra, Salt2'yi sola dairesel bit kaydırma (circular rotate) işlemine tabi tutuyoruz.

7. İkinci zıplamada, Salt2'in alçak 8 bit değerini gövde M'in işaret ettiği pozisyondaki bayt ile XOR ediyoruz. Bu gövde değerini ise, Salt1'nin alçak 8 biti ile XOR ediyoruz. M değerini de V ile XOR ederek zıplamayı gerçekleştiriyoruz. Sonra, Salt1'yi sağa dairesel bit kaydırma (circular rotate) işlemine tabi tutuyoruz.

8. Eğer, ikiden fazla zıplama yapacaksak, diğer zıplamalar da bir ve ikinci zıplama ile aynı yöntemi kullanıyor.

9. Tüm zıplamalar bittikten sonra, şifrelenecek ya da deşifre edilecek karakteri Salt1, Salt2 ve V değerlerinin XOR edilmesi sonucu ile oluşan değer ile XOR ediyoruz.

Açık metnin o anki CRC kontrol toplam değerini güncelliyoruz. Ve V değişkenini bu CRC toplam değeri ile XOR ederek güncelliyoruz.

10. Bu işlemleri tüm şifrelenecek metin için tekrar ediyoruz.

Görüldüğü üzre, elimizde, kullanabileceğimiz tüm değişkenleri fonksyon içinde kullanarak hem çok büyük bir kaba kuvvet saldırı karmaşıklığı yakalıyor, hem de mümkün olan en fazla rastgeleselliği elde etmeye çalıştık. Elimizdeki “görsel kanıtlar”, algoritmanın bu konuda gözle görülür derecede başarılı olduğunu ispatlıyor[xv]. Dahası, saldırganın tüm saldırı araçlarına “karartma” uyguluyoruz. Başlangıç değerleri, her şifreleme için üretilen 64 bitlik bir rastgele sayıya bağımlı. Zıplama adımları gizli. Dolayısı ile, gövde içinde hangi değerlerin kullanıldığını saldırgan bilemiyor.

Şifre anahtarının dinamik bir biçimde değişmesi; bu değişmenin etmenlerinin bir önce şifrelenen bayt'a da bağlı olması şifrelemenin klasik XOR'un tüm zaaflarını ortadan kaldırıyor. Başka bir deyişle, şifrelenen metin anahtar ile şifrelenirken, anahtar da şifrelenen metin ile şifreleniyor.

Artırıldığında şifrelemenin gücünü artıran şifreleme anahtarı uzunluğunun algoritmik karmaşayı etkilememesi, algoritmanın başka bir kuvvetli yönüdür. Algoritmik karmaşayı, yani hızı etkileyen faktör zıplama sayısıdır.

XOR algoritmasının klasik zaafları null bytes(0 XOR V=V ve V XOR V=0) ve frekans analizleri, diferansiyel kriptoanaliz gibi yöntemler işe yaramaz hale getiriliyor. Zira, anahtarın kendisi de şifrelenen metnin akışına göre dinamik olarak değişmektedir.

Algoritma, simetrik şifreleme gerektiren pek çok alanlarda kullanılabilir. Hohha'nın tüm simetrik şifreleme işlemleri bu algoritma ile gerçekleştirilir.

Ayrıca, zıplamalar arasında, çeşitli kaynaklardan elde edilen entropi verisinin, M'in o an gövdede işaret ettiği baytın güncellenmesi için kullanılarak, Algoritma'nın, rastgele sayı üreteci olarak(Random number generator) da kullanılabileceği kanaatindeyiz.

Şifreleme yönteminin, bir blok şifreleme

olmaması ve deşifre işleminin direkt olarak şifrelenmiş metin ile aynı bellek üzerine yazılabiliyor olması, işlemci önbellek kullanımından daha iyi faydalanabileceğinden dolayı[xvi], algoritmaya başka bir avantaj sağlamaktadır.

Algoritma, şifre ya da deşifre işlemi için paralel çalıştırılamaz ya da rastgele erişimde kullanılamaz.

Veride herhangi bir bozukluk olması, örneğin anlık veri aktarımında aktarımında bir bayt hatalı veri gelmesi, sonraki tüm baytların da bozulmasına sebebiyet verir. Bundan dolayı, CRC kontrol değeri toplamı algoritmanın içine gömülmüştür. Bu değer kullanılarak herhangi bir bozulma kolayca tespit edilebilir.

Algoritma ve C uygulaması MIT&GPL lisansı ile kamuya açılmıştır[xvii]. Ortak akıl ile geliştirilmeye devam edilecektir.

Hohha XOR Algoritması: Hız

Algoritma, yapılan hız testlerinde, kırıldığı ispatlanan ve güvensiz addedilen RC4 dışında, AES de dahil, diğer tüm simetrik şifreleme yöntemlerinden daha iyi bir performans sergilemiştir[xviii].

Güvenirlik

Algoritma, rastlantı indeksi[xix](index of coincidence), Kasiski[xx], Vigenere Cypher[xxi] vb. analiz ve saldırı yöntemleriyle ortaya çıkartılan örüntüleri anlamlandırmayı neredeyse imkansız kılar! Örneğin, Türkçe açık bir metinde(plaintext), en sık tekrarlanan A harfi[xxii], şifrelenmiş metinde hemen her seferinde farklı bir değer alacaktır.

İlişkili anahtar(related key attack) saldırısına karşı güvenlidir. Anahtardaki en ufak bir değişim tamamen farklı bir şifreli metin elde edilmesine yol açar.

Şu ana değin, algoritmanın görebildiğimiz tek "teorik" zaafı, saldırganın hem tuzlama değerini, hem açık metin değerini ele geçirmesidir. Bu durumda şifre çok zorlanmadan kırılabilir.

Algoritma modelini şu şekilde tasarladık:

Bir anahtar oluşturulduğunda, anahtarın içinde Orijinal bir tuzlama değeri barındırıyor. Bu tuzlama değeri, sadece ve sadece diğer tuzlama değerlerini şifrelemek amacıyla kullanılıyor.

Her farklı şifreleme için yeni bir rastgele tuzlama değeri oluşturulmalı. Bu değer, Orijinal anahtar ve Orijinal tuzlama değeri ile şifrelenmeli ve alıcıya ulaştırılmalı. Asıl şifrelenecek metin ise, Orijinal anahtar ve bu tuzlama değeri ile şifrelenerek alıcıya gönderilmelidir.

Alıcı, önce anahtar ve orijinal tuzlama değeri ile o an kullanacağı tuzlama değerini deşifre edecek ve daha sonra elde ettiği bu tuzlama değeri ve anahtar ile gerçek veriyi deşifre edecektir. Ayrıca, bu zaaf gibi görünen nokta, özellikle yüzyüze belirlenen ortak anahtarların çok uzun süre kullanılabilmesi için bir avantaj haline geliyor: Belirli aralıklarla, anahtarı değil, fakat, orijinal tuzlama değeri değiştiriliyor.

Bu, teorik olarak bir açıktır. Lakin, tuzlama değerlerinin doğaları itibarı ile rastgele sayılar olması, bunun şifre kırma yöntemleri elde edilmesine pek imkan bırakmıyor. Sadırganın bu atağı gerçekleştirebilmesinin bizim görebildiğimiz tek yolu, rastgele sayı üretecinin zaaflarından yararlanmasıdır. Bundan dolayı, rastgele sayıların güvenilir kaynaklardan elde edilmesi hayati bir önem taşımaktadır.

Tuzlama değerlerini elde edemediği sürece, saldırganın hem şifreli metni ve şifrelenen metni bilmesinde, elinde, çok fazla sayıda hem şifreli hem şifrelenmemiş metin biriktirmediği sürece bir sakınca yok. Bu sakınca da, düzenli aralıklarla anahtarın "orjinal tuzlama değerinin değiştirilmesi ile" aşılıyor.

Tuzlama ve anahtarın dinamik değişimi diferansiyel kriptoanaliz[xxiii] yöntemlerini devredışı bırakıyor. İlk zıplama pozisyonu tuzlama değerine bağlıdır. Anahtar her kullanılışında farklı çıktılar veriyor. Ayrıca, şifreli metnin kendi içinde de şifreli karakterler arasında korelasyon kurmayı imkansızlaştırıyor.

Sonuç

ABD'nin gizli belgeleri için kullandığı AES, özellikle, işlemcilerde bu şifreleme için özel geliştirilmiş komut seti desteği ve çok tatmin edici hızı da göz önünde alındığında ilk akla gelen seçenek. AES'in CBC dışındaki kullanım şekilleri oldukça yavaş. Ve söylediğimiz gibi, "standartlaşmış" algoritmaların tarihi, bize güven vermiyor. AES'in, en çok kullanılan CBC modunun

kırılabildiğini ve güvensiz olduğunu kanıtlayan bir bilimsel makale mevcut[xxiv]! TLS 1.3 standarrdı taslağı bu şifrelemeye yer vermiyor ve mevcut SSL 3.0 standardında güvenilir tek bir algoritma mevcut değil[xxv].

“Daha iyiyi” arayış hiç bitmiyor!

Kriptoloji uzmanı değiliz ve böyle bir iddiamız yoktur. Kamuya açılmış kütüphaneler ile, standart şifreleme yöntemleri kullanmak hiç zor değil.

Özellikle altını çizmek isteriz ki, yeni bir şifreleme algoritması geliştirmeye girişmek altına girmeyi arzu etttiğimiz bir yük değildir. Fakat, detaylı bir şekilde açıkladığımız gibi, Standart, güvenilir ve hızlı şifreleme algoritmaları eksikliği, Google'ın da dahil olduğu pek çok ticari şirketi, kendi çözümlerini geliştirmeye zorlamış, bu yıl içinde, Google'ın standartlarda yer almayan Cha-Cha20 algoritmasını kullanmaya başlayacağını duyurması üzerine, söz konusu algoritma TLS 1.3 “Müsvette”(Draft)sinde yer almaya başlamıştır. Bu durumda, şifre uzmanı olmayan şirketlerin, global düzeyde, kriptoanalistlerin ve bu alanda çalışan akademisyenlerin, ya mesleki yeterliliklerini, ya da mesleki etiklerini sorgulamaya başlamaları doğaldır.

Topluma, aslında güvenilmez olduğunu bile bile, “Amerikan Hükümeti'nin çok gizli dokümanları şifrelemekte kullandığı standart algoritmaları kullanıyoruz” reklamı yapmak ve zahmetsizce hazır kütüphaneler kullanmak aslında ticari olarak doğru olan seçenektir. Biz zor fakat dürüst olan yolu tercih ediyoruz.

Hakemlerin,Yürütme Kurulu üyelerinin ve akademisyen dostlarımızın katkı ve eleştirileri ile, prıogramımızın tasarımında pek çok kez değişikli yaptık.

Algoritma ve bu makalenin yazımı, ürünün piyasaya çıkışını geciktirecek. Fakat, bizim açımızından çok faydalı bir süreç oldu. Özellikle ODTÜ'de Olimpik Takım'da iken de bizi eğiten Bilgisayar Müh. Bölümü Ekibi'ne, başta dostlarımız Dr. Onur Tolga Şehitoğlu ve Dr. Meltem Turhan Yöndem olmak üzre teşekkür ederiz.

Şifreleme teorisi tarihi, “Kırılması imkansız” ve “popüler” algoritmalar mezarlığıdır. Fakat asıl korkunç gerçek şu: Hemen tüm simetrik şifreleme yöntemleri, RSA asimetrik şifrelemesine bağımlı. RSA kırılırsa, en güçlü simetrik şifreler bile anlamsız kalacak! Tüm finans kurumları, ordular, istihbarat örgütleri... Tüm Internet bu standart üzerinden şifreleniyor. Şu anda 2048 bit'den aşağısı güvenli addedilmiyor. 4096 bit'in üstü ise, 4096 bit'e göre, aşırı bir hız maliyeti getirirken, aynı oranda güvenlik getirmiyor. Kısacası, RSA'in sonu geliyor veya daha kötüsü, çoktan geldi, biz bilmiyoruz!

RSA'in ansızın kırılması, global bir ekonomik ve siyasi krize sebep olacaktır ve bu olasılık çok yakın, çok güçlü! Müşterilerimize, “yüzyüze simetrik anahtar belirleme” dışında, hiçbir mevcut yönteme çok bel bağlamamalarını salık veriyoruz.

“Mahremiyet” ihtiyacı, su ihtiyacına çok benziyor. İkisi de son derece hayati. İkisi de yokluğu hissedilmediği sürece umursanmıyor ve para ile satın alınmak istenmiyor, ikisi de ancak yokluğu fark edildiği zaman çok ağır bedeller ödetiyor ve ikisinin de yokluğu hissedilmediği sürece kitlelere para ile satılabilmesi son derece zor.

Bu ürünü milyar dolarlık rakipler karşısında tutundurabilmek büyük bir çaba gerektiriyor. Bunun için, bankaların kredi kartları için yaptıkları gibi, yeni üye getiren üyelere bonus olarak belirli bir yüzde verilmesi ve piramidal referans zinciri oluşturulması gibi pek çok ticari pazarlama yöntemi deniyoruz.

Fakat, söylediğimiz gibi, sonuçta, belirleyici olan, kitlelerin, aynen su örneğinde olduğu gibi, bu ihtiyacın da para karşılığı giderilebilecek kadar önemli olduğunun farkına varmaları. Er ya da geç bu uyanışın gerçekleşeceğine inanıyoruz. Bu makalede bahsettiğimiz pek çok olay da bunun ispatıdır. Ümidimiz, bu uyanışın, hem birey hem ulus bazında çok ağır bedeller ödenmeden gerçekleşmesidir.

i Örneğin, T.C. Anayasası Madde 22: https://www.tbmm.gov.tr/develop/owa/anayasa.maddeler?p3=22

ii http://www.theregister.co.uk/2015/09/03/wikipedia_industrial_scale_smears_and_blackmail/

iii https://mises.org/library/hegel-state-gods-will

iv http://www.wired.com/2012/03/ff_nsadatacenter/

v http://www.hurriyet.com.tr/nsa-122-lideri-dinlemis-26111203

vi https://en.wikipedia.org/wiki/Pretty_Good_Privacy

vii http://www.turkiyegazetesi.com.tr/dunya/316112.aspx

viii http://www.turkhukuksitesi.com/showthread.php?t=36968

ix https://tr.wikipedia.org/wiki/Diffie-Hellman_anahtar_değişimi

x https://en.wikipedia.org/wiki/XOR_cipher

xi https://tr.wikipedia.org/wiki/RSA

xii Rastgele sayı üretmek için, sunucuda, güvenirliği ispatlanmış /dev/urandom, istemcide window.crypto standart kaynakları kullanılır. Bu iki kaynağın da kriptoloji için yeterli entropi kullandığı genel kabul görür.

xiii https://en.wikipedia.org/wiki/Man-in-the-middle_attack

xiv https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

xv http://ismail-kizir.blogspot.com.tr/2015/11/visual-proofs-of-hohha-dynamic-xor.html

xvi https://en.wikipedia.org/wiki/Locality_of_reference
http://gameprogrammingpatterns.com/data-locality.html

xvii http://ismail-kizir.blogspot.com.tr/

xviii Dipnot kısmına tablo yerleştirilemediğinden ayrı bir doküman olarak veriliyor.

xix http://www.thonky.com/kryptos/index-of-coincidence

xx https://en.wikipedia.org/wiki/Polyalphabetic_cipher

xxi https://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher

xxii https://en.wikipedia.org/wiki/Letter_frequency

[xxiii] https://en.wikipedia.org/wiki/Differential_cryptanalysis

[xxiv] http://link.springer.com/chapter/10.1007%2F3-540-45708-9_2

[xxv] https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-30