Kriptolojik Rasgele Sayı Üreteçleri

Cryptographic Random Number Generators

Fatih ÖZKAYNAK

Fırat Üniversitesi Yazılım Mühendisliği Bölümü 23119 Elazığ/Türkiye

E-Posta: ozkaynak@firat.edu.tr, ozkaynak_fatih@hotmail.com

Öz

Rasgele sayı üreteçleri kriptolojik uygulamalar için önemli bir araçtır. Çünkü rasgelelik kaynağının yetersizliği tüm sistemin güvenliğini etkileyebilmektedir. Gerekli rasgele verinin frekansı ve miktarı uygulama ile büyük farklılık gösterebilmektedir. Bu yüzden kullanıcının ya yüksek kalitede rasgele veri ya da çok büyük miktarda sözde rasgele veri üretmek isteyebileceği hesaba katılmalıdır. Rasgele sayı üreteçlerinin çeşitli tipleri bulunmaktadır. Bu çalışmada kriptolojik uygulamalar için gürbüz rasgele sayı üreteçlerinin gereksinimlerini ve bu gereksinimleri gerçekleştirecek mimari tanımlanmıştır.

Anahtar Sözcükler : Kriptoloji, Rasgele sayı üreteci

Abstract

Random number generators are an important tool for cryptographic applications. Since inadequate source of randomness can be effect security of whole system. The frequency and the amount of required random data can differ greatly with the application. Therefore, random data generation should take into account the fact that the user can request either high quality random data or a great amount of pseudorandom data. There are several types of random number generators. This paper describes requirements of robust random number generators for cryptographic applications and an architecture to realize these requirements.

Keywords : Cryptography, Random Number Generators

1. Giriş

Rasgelelik istatistik, oyun teorisi, benzetim, nümerik analiz, eğlence gibi birçok uygulama alanında gereksinim duyulan temel bir karakteristiktir. Kriptolojik uygulamaların büyük bir çoğunluğu da rasgele sayılara ihtiyaç duymaktadır. Oturum anahtarları, imza anahtarları ve parametreleri, kimlik doğrulama protokolleri, geçici anahtarlar, sıfır bilgi ispatı, blok şifreler için başlangıç vektörleri ve yan kanal saldırılarına karşı koruma önlemleri olan körleştirme ve maskeleme işlemleri rasgele sayılara gereksinim duyan kriptolojik uygulamalardan sadece birkaçıdır [1-3].

Literatürde genellikle kriptolojik uygulamaların gereksinim duyduğu rasgelelik ile diğer uygulamalar için sağlanması yeterli olan rasgelelik birbiri ile karıştırılmakta ve sonuç olarak çeşitli güvenlik kriterleri sağlanmamaktadır. Bu durum ise tüm sistemin güvenliğini etkilemektedir [3-5]. Örneğin herhangi bir benzetim uygulamasında kullanılacak rasgele sayı üreteçlerinin iyi istatistiksel özellikler göstermesi yeterli iken bir şifreleme algoritmasında anahtar olarak kullanılacak rasgele sayıların tahmin edilemez olması yani saldırgan maksimum uzmanlık bilgisine ve sınırsız hesaplama gücüne sahip olsa bile kör bir tahminden (kaba kuvvet saldırısından) daha fazlasını yapmasına olanak verilmemelidir. Diğer bir ifade ile n rasgele bit değerinden oluşan anahtarın tahmin edilmesi ortalama 2^{n-1} deneme gerektirmelidir.

Gönderim ve kabul tarihi : 26.12.2014 – 22.12.2015

Bu çalışmada kriptolojik uygulamaların gereksinim duyduğu rasgele sayı üreteçlerinin (RSÜ) tasarım ve değerlendirme kriterleri detaylı olarak tartışılmıştır. Bölüm 2'de rasgele sayı üreteçleri ile alakalı kısa bir literatür özeti verilmiştir. Bölüm 3'de rasgele sayı üreteçleri gerekirci (deterministtik) RSÜ ve gerçek RSÜ olmak üzere iki temel tasarım sınıfı olarak sınıflandırılmıştır. Gerekirci RSÜ tasarımları Bölüm 4'de incelenirken gerçek RSÜ tasarımları Bölüm 5'de tartışılmıştır. Bölüm 6'da kriptolojik uygulamalarda RSÜ'lerinin sahip olması istenen temel gereksinimleri tanımlanmıştır. RSÜ'ler için önemli standart ve değerlendirme kriterleri Bölüm 7'de incelenmiştir. Son bölümde çalışma özetlemiştir.

2. Rasgele Sayı Üreteçlerinin Tarihsel Gelişimi

Rasgele sayıların kriptolojik uygulamalarda merkezi kullanımı; Vernam şifreleme sistemi olarak bilinen tek bir ayrıcalıklı veya (exclusive-OR) kapısı kullanılarak, açık metin (plaintext) olarak tanımlanan veri dizisinin anahtar olarak adlandırılan rasgele sayılar kullanılarak karıştırılması protokolüdür. Koşulsuz güvenli şifreleme protokolü olarak ifade edilen bu dizi (stream) şifreleme protokolünün pratik uygulamalarda da güvenli olabilmesi için her bir anahtarın (rasgele sayının) sadece tek bir defa kullanılması ve sayıların gerçek rasgele olması istenmektedir.

Bilgisayar tabanlı rasgele sayı üreteçlerinin tasarım ve analizi için önemli kaynaklardan biri Knuth'un "The Art of Programming" isimli klasik eseri olmuştur [12].

Ripley 1983 yılında yayınladığı çalışmasında; kişisel bilgisayar kullanıcılarının uzmanlar tarafından geliştirilen rasgele sayıları üretmek için kullandıkları kütüphanelere erişimindeki problemleri adreslemiştir [13]. Ripley çalışmasında küçük kişisel bilgisayarlar tarafından sağlanan yetersiz rasgele sayı üreteçlerini kullanıcı programları ile yer değiştirerek üstel, normal ve Poisson dağılımına sahip diziler üreten etkili yöntemler geliştirmiştir.

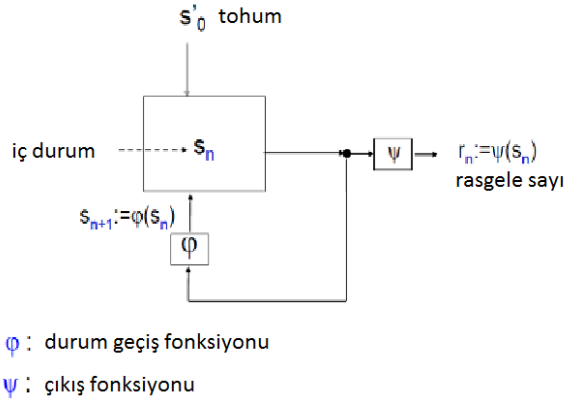
1990 yılında L'Ecuyer [14] orta seviyedeki bir bilgisayar kullanım bilgisine sahip bir kullanıcı için düzgün rasgele değişkenler üretilmesi problemini çözmüştür. Yine 1990 yılında James tarafından yapılan bir çalışmada Monte Carlo hesaplamaları için sözde rasgele sayı üreteçlerinin pratiği ile ilgilenmiştir [15]. Bu yıl yapılan bir diğer çalışmada ise Lagarias sözde rasgele sayı üreteçleri, tek yönlü fonksiyonlar ve gizli anahtarlı şifreleme sistemleri arasındaki bağlantıyı ortaya koyarak sayı teorisini temel alan sözde rasgele sayı üreteçlerini tanılamıştır [16]. Lagarias çalışmasında ayrıca bu üreteçlerin kripto analizi üzerine elde ettiği sonuçları özetlenmiştir.

1991 yılında Zeng ve arkadaşları akan şifreleme sistemleri için temel problem olan kısa uzunluklu bir rasgele anahtardan tahmin edilemez uzunluklu ikili sinyallerin üretilmesi problemini ele almıştır [17].

Rasgele sayı üreteçleri üzerine kapsamlı bir değerlendirme çalışması ise Ritter tarafından gerçekleştirilmiştir [18].

3. Rasgele Sayı Üreteçlerinin Sınıflandırılması

Gerçek dünya RSÜ'leri iki temel sınıfa ayrılmaktadır. Birinci sınıf gerekirci RSÜ'leri (sözde veya sahte rasgele sayı üreteçleri olarak da bilinmektedir) içermektedir. Gerekirci RSÜ bir tohum değerinden başlayıp algoritmik olarak rasgele sayıları üretmektedir. İkinci sınıf RSÜ'leri ise gerçek RSÜ'leridir ve kendi içerisinde fiziksel gerçek RSÜ ve fiziksel olmayan gerçek RSÜ olarak ikiye ayrılmaktadır. Fiziksel gerçek RSÜ'ler elektronik devrelerin gerekirci olmayan etkilerini (Zener diyottaki gürültü, yarı iletkendeki termal ısı, serbest çalışan osilator gibi) veya fiziksel deneyleri (radyo aktif bozulma arasında geçen zaman, kuantum rasgele süreçler gibi) kullanır. Fiziksel olmayan gerçek RSÜ'leri ise gerekirci olmayan olaylarda (sistem zamanı, hard disk arama zamanı, RAM içeriği, kullanıcı etkileşimleri gibi) ortaya çıkmaktadır [3].



Şekil-1: Gerekirci RSÜ'nün genel tasarım mimarisi

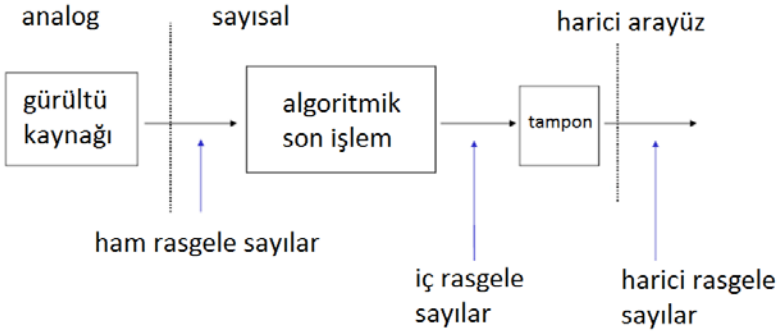
4. Gerekirci RSÜ Tasarımları

Şekil 1'de gerekirci RSÜ'nin genel tasarım mimarisi gösterilmiştir. $r_1, r_2, \dots, r_n \in R$ rasgele sayıları $s_n \in S$ gerekirci RSÜ'nin iç durumlarını göstermektedir. Burada S ve R sonlu kümeleri sırasıyla gerekirci RSÜ'nin durum uzayı ve çıkış uzayı olarak adlandırılmaktadır. $\psi: S \to R$ çıkış geçiş fonksiyonu, mevcut s_n iç durumundan r_n rasgele sayısını üretir. Ardından s_n durumu φ durum geçiş fonksiyonu kullanılarak $s_{n+1} := \varphi(s_n)$ biçiminde güncellenir. İlk iç durum değeri olan s_1 değeri tohum değeri olan s_0 kullanılarak $s_1 := \varphi(s_0)$ biçiminde veya daha karmaşık tasarımlar kullanılarak güncellenir. s_0 tohum değerinden tüm $s_1, s_2, \dots, s_n$ iç durumlarının ve $r_1, r_2, \dots, r_n$ rasgele sayıların tahmin edilebileceği açıktır. Bu yüzden güvenlik öncelikli uygulamalarda tohum değeri rasgele seçilmeli ve durum geçiş fonksiyonu ile çıkış fonksiyonunun yeterince karmaşık olması gerekmektedir [3-6].

Gerekirci RSÜ'lerinin dezavantajı çıkış değerlerinin tohum değeri aracılığıyla tamamen belirlenebilmesi ve gelecek rasgele sayıların sadece mevcut iç duruma bağlı olmasıdır. Bu yüzden iç durum cihaz aktif olmasa bile korunmalıdır. Özellikle gerekirci RSÜ tasarımları bilgisayar veya akıllı kartlar üzerinde gerçekleştiriliyor ise mevcut iç durum sonraki oturum için tehlikeli olabilir. Gerekirci RSÜ uygulamaların avantajları ise herhangi bir adanmış donanıma ihtiyaç duymadıklarından daha ucuz olmalarıdır.

5. Gerçek RSÜ Tasarım Mimarisi

Şekil 2'de bir gerçek RSÜ'nin genel tasarım mimarisi gösterilmiştir. Tasarımın kalbini gürültü kaynağı oluşturmaktadır. Gürültü kaynağı tipik olarak elektronik devreler (gürültülü diyot veya serbest çalışan osilatör) veya fiziksel deneyler (radyoaktif bozulma veya ışığın kuantum etkisi) ile gerçekleştirilir. Gürültü kaynağı sürekli zamanlı analog sinyaller üretmekte ve aynı adımda bu değerler periyodik olarak sayısallaştırılarak ikili (binary) değerlere dönüştürülmektedir. Sayısal değerler, sayısallaştırılmış analog sinyaller olarak adlandırılırlar. Gerçek RSÜ'lerin içerebileceği potansiyel zayıflıkları indirgemek için sayılar değerlere çeşitli algoritmik son işlem (post-process) yöntemleri uygulanabilir. Ancak zayıflıkları indirgeme işlemi çoğu zaman basit bir dönüşümden daha fazlasına (bağımlılıkları eşiklemek veya veri sıkıştırma işlemi gibi) gereksinim duyduğundan bu işlemler RSÜ'nün çıkış hızını düşürmektedir. Bu yüzden gerçek RSÜ tasarımlarında güçlü gürültü kaynakları kullanılarak algoritmik son işleme ihtiyaç duymadan iç rasgele sayıları doğrudan çıkışa veren mimariler tercih edilmektedir [3-5].



Şekil-2: Gerçek RSÜ'nün genel tasarım mimarisi

6. Kriptolojik Rasgele Sayı Üreteçlerinin Temel Karakteristikleri

Bu bölümde kriptolojik uygulamalarda kullanılacak rasgele sayıların gereksinim duyduğu temel karakteristikler tartışılmıştır. İdeal olarak rasgele sayılar tanımlı olduğu aralıkta düzgün dağılıma sahip ve birbirinden bağımsız olmalıdırlar. Bu ideal bir RSÜ'nün karakteristiğidir. Ancak bu karakteristik bir matematiksel tasarımdır. Gerçek hayattaki uygulamalarda kullanılan rasgele sayıların ideal olması çok zordur. İdeal rasgele sayıları üretmek mümkün olsa bile bunu kanıtlayabilmek diğer önemli bir problemdir.

Çizelge-1: Kriptolojik rasgele sayı üreteçleri için temel gereksinimler

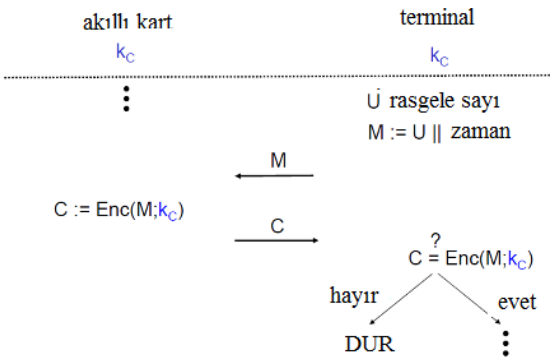
No	Gereksinim Açıklaması
(G1)	Rasgele sayılar herhangi bir istatistiksel zayıflık göstermemelidir.
(G2)	Rasgele sayıların alt dizilerini bilmek öncül ve ardıl rasgele sayıların hesaplanmasına veya tahminine izin vermemelidir.
(G3)	RSÜ'nin iç durum değerinin bilinmesi durumunda veya iç durum değeri bilinmese bile tahmin edilebilme olasılığı ile önceki rasgele sayıları hesaplayabilmek mümkün olmamalıdır.
(G4)	RSÜ'nin iç durum değerinin bilinmesi durumunda veya iç durum değeri bilinmese bile tahmin edilebilme olasılığı ile gelecek rasgele sayıları hesaplayabilmek mümkün olmamalıdır.

Kriptolojik uygulamalarda rasgele sayı üretmek amacıyla kullanılan rasgele sayı üreteçlerinin belirli güvenlik gereksinimlerini sağlamaları gerekmektedir. Bu gereksinimler Çizelge 1'de listelenmiştir [3, 5]. Çizelge 1'de listelenen gereksinimlere olan bağlılık ele alınan uygulamanın ihtiyaç duyduğu güvenlik düzeyine göre değişmektedir. Hassas kriptolojik uygulamalar (G1) ve (G2) gereksinimlerini mutlaka sağlamalıdır. Gerekirci RSÜ'ler için (G3) ve (G4) gereksinimleri ek gereksinim olarak temel alınmaktadır. Gerçek RSÜ'ler için ise (G2) gereksiniminin sağlanmasının ardından (G3) ve (G4)'ün otomatik olarak sağlandığı kabul edilmektedir.

6.1. (G1) Gereksiniminin Önemi

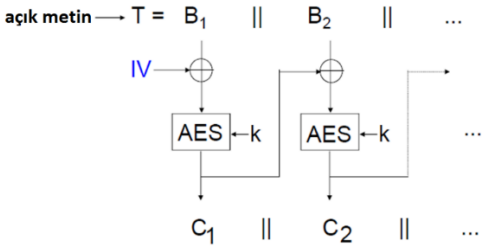
Şekil 3'de basitleştirilmiş bir soru/cevap protokol mimarisi gösterilmiştir. En basit anlamda protokol; iletişim kuracak tarafların nasıl davranması gerektiğini söyleyen sonlu adımlar kümesidir (bir algoritmadır). Şekil 3'de iletişim kuracak taraflar olan akıllı kart ve terminal başlangıçta gizli bir k_c anahtar değeri üzerinde anlaşmışlardır. Tasarlanan protokolde terminal bir RSÜ üreteci aracılığı ile U rasgele sayısını üretmiştir. U rasgele sayısının ardına zaman bilgisi eklenerek bir M karakter dizisi oluşturulmuştur. M değeri akıllı karta soru olarak gönderilir ve ardından akıllı kartın cevabı olan $C = Enc(k_c, M)$ değeri alınır. Eğer terminal tarafında

hesaplanan C değeri ile akıllı kartın gönderdiği C değeri aynı ise işleme devam edilir. Aksi takdirde işlem sonlandırılır. Şekil 3'deki protokolde aynı rasgele sayılar defalarca kullanılıyor ise saldırgan tekrarlama saldırı olarak adlandırılan yöntem ile sistemin güvenliğini tehdit edebilmektedir. Bu tip saldırıları önlemek için tasarım mimarisinde kullanılacak $U_1, U_2, \ldots$ rasgele sayılarının büyük olasılıkla birbirinden farklı olması gerekmektedir.



Şekil-3: Basit bir soru/cevap protokolü

Diğer bir protokol yapısı Şekil 4'de gösterilen blok şifreleme mimarisidir. Blok şifreleme algoritmaları açık (orijinal) metinleri ardışık bloklara böler ardından her bloğu şifreleyerek şifreli metin bloklarına dönüştürmektedir. Blok şifre sistemlerinin en büyük dezavantajı açık metindeki birbirinin aynısı olan blokların, şifreli metinde de birbirinin aynısı olmasıdır. Bu problemi gidermek için çeşitli işlem modları kullanılmaktadır. Şekil 4'deki tasarımda kullanılan işlem mod'unda her bir şifreli bloğun çıktısını bir sonraki bloğa girdi olarak verilmiştir. Ancak ilk bloğun şifrelenmesi için kullanılan giriş olmadığı için IV (Initialization Vector) sembolü ile gösterilen başlangıç vektörleri kullanılmaktadır. IV'ler genellikle rasgele sayı üreteçlerinden sağlanmaktadır. Şekil 4'deki protokol için hem tekrar hem de korelasyon saldırılarını önlemek için IV rasgele sayılarının düzgün dağılıma sahip olması gerekmektedir.



Şekil-4: Blok şifreleme mimarisi

Sonuç olarak herhangi bir rasgele sayı üreteci iyi istatistiksel özelliklere sahip olmalıdır. Bu özelliği kontrol etmek için (G1) gereksinimi tanımlanmıştır. (G1) gereksinimi genellikle özel istatistiksel test araçları ile kontrol edilmektedir. İstatistiki rasgelelik testleri genellikle hipotez testleri olarak da bilinirler. Testlerin hipotezi kabul veya ret etmesine göre test sonuçları başarılı veya başarısız olarak sonuçlandırılır. Bu testlerden en yaygın kullanılan test araçları NIST, FIPS ve Diehard olarak bilinmektedir.

6.2. (G2) Gereksiniminin Önemi

(G1) gereksinimi hassas kriptolojik uygulamalar için yetersiz kalmaktadır. Bazı RSÜ'leri iyi istatistiksel özelliklere sahip olmasına rağmen saldırgan rasgele sayıların bilinen küçük bir alt dizisini kullanarak rasgele sayıların tüm dizisini kestirebilir. Saldırganın bazı rasgele sayılara ulaşabileceği birçok uygulama için gerçekçi bir olasılıktır. Örneğin oturum anahtarlarının üretiminde aynı RSÜ kullanılıyor ise soru cevap açık olarak iletileceği için birçok rasgele sayı elde edilebilir. Diğer bir örnek uygulamada rasgele seçilen oturum anahtarı ile gizlenen mesajı açma yetkisine sahip bir kullanıcı en azından bir anahtara sahip ve ayrıcalıklı bir saldırgan

pozisyonundadır. Bu yüzden kriptolojik olarak güvenli RSÜ tasarımları için (G2) gereksiniminin sağlanması önerilmektedir. (G2) gereksinimi saldırganın rasgele sayıların alt dizilerini bilmesi ile öncül ve ardıl rasgele sayıların hesaplanmasına veya tahminine izin verilmediğinin sağlanıp sağlanmadığının kontrol edilmesiyle ilgilidir. Bu gereksinimin sağlanması için rasgele sayı üretecinde kullanılan yapıların karmaşık olması gerekmektedir. Bu karmaşıklığı sağlamak için genellikle literatürde yaygın biçimde çalışılmış ve güvenlik özellikleri iyi bilinen kriptolojik yapıların kullanılması önerilmektedir.

Örneğin literatürde en yaygın biçimde kullanılan gerekirci RSÜ tasarımı olan Geri Beslemeli Kaydırmalı Doğrusal Yazmaç (Linear Feedback Shift Register, LFSR) yapısı $a_{n+t+1} = c_1 a_{n+t} + \cdots + a_{n+1} (mod\ 2)$

özyinelemli formülü ile tanımlanmaktadır. LFSR tasarımları küçük olmayan t değerleri için iyi istatistiksel özellikler göstermektedir. Bu yüzden (G1) gereksinimini sağlamaktadır. Ayrıca etkili bir biçimde çok hızlı olarak uygulanabilir. Fakat LFSR yapısı GF(2) üzerinde doğrusal bir yapıya sahip olduğundan eğer t çıkışa (rasgele sayıların bilinen küçük bir alt dizisine) sahip herhangi biri bu bilgiden tüm rasgele diziyi elde edebilir. Sonuç olarak LFSR tasarımı (G2) gereksinimini sağlamaz ve kriptolojik uygulamalar için uygun değildir.

Ama rasgele sayıları üretmek için tasarımda Enc: $\{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^n$ bir blok şifre, $\{0,1\}^n$ açık metin $\{0,1\}^m$ anahtar olsun. $S = \{0,1\}^n \times \{0,1\}^m$ ve $R = \{0,1\}^n$ olmak üzere $s_n = (r_n, k)$ için k gizli kalmak üzere $\psi(r_n, k) = r_n$ ve $s_{n+1} = (Enc(r_n, k), k)$ biçiminde bir RSÜ mimarisi kullanılıyor ise güçlü blok şifreler için Enc ve Enc^{-1} fonksiyonlarında seçilen açık metin saldırısı bulmak pratik olarak mümkün olmadığı için tohumlama sürecinin garanti edilmesi ile k'nın tahmin edilemeyeceği garanti edilmiş olunur. Bu yolla eğer blok şifre olarak AES veya 3DES kullanılıyor ise RSÜ'nün (G2) gereksinimini sağladığı varsayılabilir.

6.3. (G3) Gereksiniminin Önemi

Eğer RSÜ bilgisayar veya bir akıllı kart üzerinde gerçekleştirilmiş ise saldırgan başarılı bir donanım saldırısı uygulayarak mevcut s_n iç durum değerini elde ettiğini varsayarsak s_n iç durumunu takip ederek $r_n, r_{n+1}, \ldots$ rasgele sayıları elde edilebilir. Bu yüzden birçok uygulama RSÜ'ler için (G3) ek bir gereksinime ihtiyaç duyar.

Çizelge 1'de verilen tanıma göre (G3) gereksinimi $\varphi: S \rightarrow S$ biçiminde bir tek yönlü durum geçiş fonksiyonu istemektedir. Tek yönlü fonksiyonlarda x ve x' girdileri, y ve y' çıktıları göstermek üzere $\varphi(x) = y$ değeri bilindiğinde, x'i hesaplamak sonlu zamanda mümkün olmamalı ve y biliniyor iken $\varphi(x') = y$ olacak bir x' bulmak zor (hesaplamak sonlu zamanda mümkün değil) olmalıdır. Bu durumda blok şifreleri temel alan RSÜ tasarımları (G3) gereksinimini sağlamaz çünkü $r_{n-1} = Enc^{-1}(r_n, k), r_{n-2} = Enc^{-1}(r_{n-1}, k), \ldots$ eşitliği yazılabilir. Ancak $S = R = \{0,1\}^n$ olmak üzere φ ve ψ fonksiyonları hash fonksiyonları ile tanımlanırsa her iki fonksiyonda tek yönlü özelliğini sağladığından RSÜ (G3) gereksinimini sağlamaktadır.

6.4. (G4) Gereksiniminin Önemi

Belirli uygulamalarda saldırgan RSÜ'nün mevcut iç durumunu bilebilir (örneğin RSÜ'nün bilgisayarda yazılım olarak gerçekleştirilmesi gibi) ve kullanıcıdan habersiz sonraki rasgele sayıları üretebilir. Örneğin $s_n \rightarrow \varphi(s_n)$ ile iç durumu güncelleyerek $r_n = \psi(s_n)$ değerini hesaplayan RSÜ tasarımlarında (G4) gereksinimi sağlanmaz. Dolayısıyla (G4) gereksiniminin sağlanması için ek bir girişe ihtiyaç duyulmaktadır. Tasarımın güvenliği ise ek girişin rasgeleliğine bağlıdır. Fiziksel kaynaklardan gelen düzenli ek girişler (G4) gereksinimini sağlamaktadır.

(G3) ve (G4) gereksinimlerinin gerekirci RSÜ'lere özel gereksinimler olduğu unutulmamalıdır. Gerçek RSÜ'ler için ise eğer (G2) gereksinimi sağlanıyor ise (G3) ve (G4) gereksinimleri otomatik olarak sağlandığı kabul edilmektedir.

7. Rasgele Sayı Üreteçleri için Değerlendirme Rehberleri

Bir değerin "rasgele" olabilmesi için bu değerin tanımlı olduğu küme içerisinden keyfi olarak seçilmesi gerekmektedir. Benzer şekilde bir karakter dizisinin "rasgele" olabilmesi için rasgele değerlerin dizisinden oluşması gerekmektedir. Ancak "rasgele olmayan" bir karakter dizisinin dağılımı incelendiğinde düz bir dağılıma sahip olmadığı gözükmektedir. Bu yüzden rasgeleliği test etmek için dizinin olasılık dağılımı incelenmektedir.

Rasgelelilik olasılıksal bir özelliktir öyle ki, rasgele dizinin özellikleri olasılıksal olarak tanımlanabilir. Rasgele sayıların olasılıksal özelliklerini değerlendirmek için birçok istatistiksel test vardır. Bu testler dizinin rasgele olup olmadığının anlaşılmasını sağlayacak örnekleri tespit etme sürecinde kullanılmaktadır. Birçok istatistiki test olduğu için tüm testleri geçen bir üreteç için bile rasgele diyemeyiz. Çünkü yeni ortaya koyulacak bir test için üretecin başarısız olma olasılığı bulunmaktadır. Bu yüzden istatistiksel testlerin sonuçları iyi yorumlanmalıdır.

İstatistiksel testlerin temel çalışma prensibi, belirli bir geçersiz hipotezi (H0) test etmek için bir formül ortaya koymaktır. Burada amaç, test altındaki geçersiz hipotez test edilen dizinin rasgele olduğudur. Geçersiz hipotez ile ilgili olarak alternatif hipotez (Ha) dizinin rasgele olmadığıdır. Her uygulanan test için, geçersiz hipotezin kabul veya ret edilmesi yönünde bir karar veya sonuç ortaya çıkar. Her bir test için uygun bir rasgelelilik istatistiği seçilmiş olmalı ve geçersiz hipotezin kabul edilmesi veya ret edilmesin belirlenmesinde kullanılmalıdır. Rasgelelilik varsayımı altında, böyle bir istatistiğin olası değerlerinin dağılımı vardır. Geçersiz hipotez altında bu istatistiğin teorik referans dağılımı matematiksel yöntemler ile belirlenebilir. Bu referans dağılımından kritik bir değer belirlenir. Test süresince, test istatistiksel değeri verileri hesaplanır. Test istatistiksel değeri kritik değer ile karşılaştırılır. Eğer test istatistiksel değeri kritik değeri geçiyor ise, rasgelelik için geçersiz hipotez ret edilir. Aksi takdirde, geçersiz hipotezi ret edilemez (örneğin, hipotez kabul edilir).

Kaynak [7] ve [8]'deki değerlendirme rehberleri RSÜ'ler için önerilen özellikleri doğrulayacak değerlendirme kriterlerinin tanımlayıp açıklamaya çalışmaktadır. Kaynak [7, 9, 10, 11]'de gerekirci RSÜ tasarımları için kabul edilmiş değerlendirme rehberleri ve standartlar verilmiştir.

Fiziksel RSÜ'ler için kabul edilmiş tasarımları özelleştirmek çok daha güçtür. Çünkü güvenlik belirlenen gerçekleştirime bağlı olarak değişmektedir [8]. Belirlenen gerçekleştirmeleri ölçecek değerlendirmelere gereksinim vardır. İstatistiksel kara kutu testleri RSÜ'nün güvenliğini garanti etmez.

8. Sonuç ve Tartışma

Rasgelelik kriptolojik uygulamalar için temel bir karakteristiktir. Bu yüzden birçok uygulamada kullanılan rasgele sayı üreteçlerinin kalitesi tüm sistemin güvenliğini etkilemektedir. Ancak literatürdeki birçok çalışmada istatistiki rasgelelik ile kriptografik rasgelelik kavramlarının karıştırıldığı ve birçok RSÜ tasarımında sadece istatistiki testlerin kullanılarak eksik değerlendirmelerin yapıldığı görülmüştür. Bu çalışmada bu eksikliğin giderilebilmesi için öncelikle RSÜ'ler sınıflandırılmış ardından gerekirci ve gerçek RSÜ'leri için genel tasarım mimarileri ve değerlendirme kriterleri verilmiştir.

Genel bir değerlendirme yapılırsa gerekirci RSÜ'nin güvenliği temel olarak iki faktöre bağlıdır. Bu faktörler ihmal edilebilecek kadar küçük bir olasılık ile gerekirci RSÜ'nin tohum veya herhangi bir iç durumunun bilinmesi ile tahmin edilecek sayıya ve çıkış fonksiyonu ile durum geçiş fonksiyonunun karmaşıklığına bağlıdır. Tipik olarak gerekirci RSÜ'lerin temel bileşeni kriptolojik temellerdir ve güvenliği bu kriptolojik temellere dayanmaktadır. Gerekirci RSÜ'ler temel aldıkları kriptolojik yapılar güvenli olduğu sürece pratik olarak güvenlidir.

Gerekirci RSÜ'lerin aksine gerçek RSÜ'lerin güvenliği ise üretilen rasgele sayıların tahmin edilemezliğine bağlıdır. Bu tahmin edilemezlik gürültü kaynağının entropisi ile yakından ilişkilidir. Entropi hesaplamasının doğru yapılması ile birlikte rasgele sayıların tahmini için yapılacak işin zamandan bağımsız olduğu yani rasgele sayıların teorik güvenliği sağlanmış olur. Gerekirci RSÜ'nün aksine bu güvenlik değerlendirmesi zamanla değişmez.

Gelecek çalışmalarda gerekirci ve gerçek RSÜ sınıflarının her biri için tasarım mimarisi detaylandırılacak; bu tasarım sınıflarının her biri için kapsamlı bir literatür özeti yapılıp, açık problemler listelenerek bu konuda çalışmayı planlayan araştırmacılar için bir başlangıç rehberi oluşturulmaya çalışılacaktır.

Kaynakça

1. Katz J., Lindell Y. Introduction to modern cryptography : principles and protocols, Chapman & Hall. (2008).
2. Paar C., Pelzl J., Understanding Cryptography A Textbook for Student and Practitioners, Springer. (2010).
3. Koç Ç. K., Cryptographic Engineering, Springer-Verlag. (2009).
4. Menezes A. J., Oorschot P. C., Vanstone S. A.. Handbook of Applied Cryptography. CRC Press, Boca Raton (1997).
5. Özkaynak F., Cryptographically secure random number generator with chaotic additional input, Nonlinear Dynamics (2014) 78 pp. 2015–2020.
6. Lagarias J. C., Pseudorandom Number Generators in Cryptography and Number Theory. Proc. Symp. Appl. Math., 42: 1990, pp. 115–143,
7. AIS 20. Functionality Classes and Evaluation Methodology for Deterministic Random Number Generators.
8. AIS 31. Functionality Classes and Evaluation Methodology for Physical Random Number Generators.
9. NIST. Security Requirements for Cryptographic Modules. FIPS PUB 140-2.
10. Marsaglia G. Diehard (Test Suite for Random Number Generators). http://www.stat.fsu.edu/pub/diehard/
11. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., Vo, S.: A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800–22rev1a (2010).
12. Knuth, D. 1981. *The Art of Computer Programming,* Vol. 2, *Seminumerical Algorithms.* 2nd ed. Addison-Wesley: Reading, Massachusetts.
13. Ripley, B. 1983. Computer Generation of Random Variables: A Tutorial. *International Statistical Review.* 51: 301-319.
14. L'Ecuyer, P. 1990. Random Numbers for Simulation. Communications of the ACM. 33(1): 85-97.
15. James, F. 1990. A review of pseudorandom number generators. Computer Physics Communications. 60: 329-344. North-Holland.
16. Lagarias, J. 1990. Pseudorandom Number Generators in Cryptography and Number Theory. Proceedings of Symposia in Advanced Mathematics. 42: 115-143.
17. Zeng, K., C. Yang, D. Wei and T. Rao. 1991. Pseudorandom Bit Generators in Stream-Cipher Cryptography. IEEE Computer. February. 8-17.
18. Ritter, T. 1991. The Efficient Generation of Cryptographic Confusion Sequences. Cryptologia. 15(2): 81-13