HAVACILIK VE UZAY TEKNOLOJİLERİ DERGİSİ
OCAK 2005 CİLT 2 SAYI 1 (65-70)

BİLGİ GÜVENLİĞİ VE ÖZCÜK FONKSİYONLARI UYGULAMALARI

Hakan AYDIN
Kara Kuvvetleri Komutanlığı
Bilgi Sistemleri Dairesi Başkanlığı
Bakanlıkar/ANKARA
haydin @kkk.tsk.mil.tr

Fuat İNCE
Hava Harp Okulu Komutanlığı
Havacılık ve Uzay Teknolojileri Enstitüsü
Bilgisayar Mühendisliği A.B.D.
Yeşilyurt / İSTANBUL
fuat.ince@superonline.com

ÖZET

Bilgisayar ağlarının, özellikle INTERNET'in, yaygın olarak kullanımı sonucu, ağ ve bilgi güvenliği konuları hayati önem kazanmıştır. Bu yazıda önce bilgi güvenliğinin önemine ve belirgin saldırı türlerine değinilmektedir. Bilgi güvenliğinin öğelerinden olan mesaj bütünlüğü ve mesaj doğrulaması için yaygın kullanılan MAC (message authentication code) yanında, özcük fonksiyonlarının (hash functions) da kullanılabileceğine işaret edilmektedir. Güvenli Özcük Algoritması (Secure Hash Algorithm - SHA-1) ele alınarak mesaj bütünlüğü ve mesaj doğrulaması amacıyla anahtarlı özcük değeri yöntemi ortaya konmaktadır. SHA-1 özcük algoritmasına dayanan anahtarlı özcük algoritması anlatılmakta ve uygulamalarla gösterilmektedir.

Anahtar Kelimeler: *Bilgi Güvenliği, Özcük Fonksiyonları, Güvenli Özcük Algoritması (SHA-1)*

ABSTRACT

The widespread use of computer networks, in particular the INTERNET, has made network and information security a subject of vital importance. The paper first discussess the importance of information security in the light of different attacks. The subject of message integrity and authentication is reviewed. In addition to message authentication codes, the use of hash functions for that purpose is presented. The SHA-1 hash algorithm is explained and a method of keyed hash function is introduced. This method, based on the use of the SHA-1 algorithm is explained and demonstrated with applications.

Keywords *: Information Security, Hash Functions, Secure Hash Algorithm (SHA-1).*

1. GİRİŞ

Bilgisayar ağlarının yaygın olarak kullanımı ile birlikte, sistemler üzerinde bulunan dosya ve bilgiler paylaşıma açık hale gelmiş, sistemlere uzaktan ve çok sayıda noktadan erişmek mümkün olmuş, kullanıcı sayıları binleri hatta milyonları bulmuştur. Bilgi sistemlerine ve bu sistemler tarafından işlenen verilere yönelik güvenlik saldırıları hızla artmaktadır. Bu gelişmeler sonucunda ortaya çıkan bilgi güvenliği ihtiyacı ve kavramı hayati önem kazanmıştır.

Veri gizliliği ve bütünlüğüne karşı saldırılar ciddi, ve giderilemeyecek kayıplara yol açabilir. Güvenlik duvarları, yedekleme vb. olağan önlemler alınmış olsa da organizasyonları saatlerce veya günlerce çalışamaz duruma getirebilir. Saldırının düzeyine ve risk edilen bilgilerin türüne bağlı olarak, saldırılar değişen önemde kayıplarla neden olur, veya sistemi tamamen işlevsiz hale getirebilir. Bu saldırılardan sonra sistemin kurtarılma maliyeti yüzlerce dolardan başlayıp, milyonlarca dolara varabilir. Saldırıları önlemenin en etkin yoluna karar vermek için, olabilecek saldırı türlerini iyi incelemek ve anlamak gerekir.

Dikkate alınan tehditlerin tümüne karşı etkin bir koruma olmadan, bir bilgi sistemi veya ağının herhangi bir bölümü, saldırılara veya yetkisiz aktivitelere karşı açık olabilir. Böyle bir durumda yönlendiriciler, anahtarlar ve ana bilgisayarlara profesyonel korsanlar, rakip şirketler veya şirket çalışanları tarafından müdahale edilebilir, büyük zararlara neden olunabilir.

2. BİLGİ GÜVENLİĞİ, SALDIRILAR VE BİR SENARYO

Günümüzde bilgisayar güvenliği ve ağ güvenliği, beraberce bilgi güvenliği kapsamında ele alınmaktadır. Bilişim sistemlerinin ve bu sistemler tarafından işlenen verilerin, bütünlüğünü, gizliliğini ya da kullanılabilirliliğini tehdit eden her türlü eylem ve hareketlere güvenlik saldırısı denir. Yetkisiz veya kötü niyetli kişilerin, bilgi sistemlerinden yararlanmaları, bilgi çalmaları, gizli bilgilere ulaşmaları, kayıtlı verilere ulaşıp onları kullanmaları, değiştirmeleri, bozmaları, silmeleri, bilgiye yetkili erişim ve kullanımı engellemeleri, ve benzeri zarar verici eylemler, bilgi güvenliğine karşı değişik tür saldırıları oluşturur. Bunların yanında arıza, kaza ve doğal afetler de bilgi güvenliğini tehdid eder. Bilgi güvenliği tüm bunlara karşı sistemi her an yetkili kullanıma açık ve kontrol altında tutmak ve bilgiyi korumak için alınan önlem ve yapılan çalışmaları kapsar.

Bir başka deyişle bilgi güvenliği bilginin gizliliğini, bütünlüğünü (integrity), erişilebilirliğini, kaynağın doğruluğunu, ve zamansal ilişkileri güvence altında tutmak etmektir. Sayısal imza da kaynak doğruluğu kavramının geniş kapsamı içinde yer alır.

Bilgi güvenliği hem dışarıdan, hem de içeriden yapılabilecek güvenlik saldırılarına karşı duyarlı olmak zorundadır. Genellikle literatür ve piyasadaki önlemler dış saldırılara karşı geliştirilmiştir. Güvenlik duvarları, virüs tarayıcılar, şifreli mesajlaşma gibi teknikler dışarıdan gelen saldırılara karşı tasarımlandığından iç kaynaklı saldırılara karşı etkin olmayabilir. Organizasyon içerisinden yapılan saldırılar, daha zor farkedilmeleri ve daha zor önlenebilmeleri nedeniyle daha zarar verici olabilirler. Birçok bilinen yöntem iç saldırılara karşı yetersizdir.

Bilgi sistemlerine başarılı nüfuz sonucu, güvenliğe yönelik, özellikle zaman bombası, mantık bombası veya daha henüz antivirüs programı olmayan değişik bir virüs, sistemlere bulaştığında, sorun virüs tarama programları ile çözülemeyebilir. Virüs etkisini gösterdiğinde her şey bitmiş zarar gerçekleşmiş olabilir.

Bilgi güvenliğine yönelik tehditler olarak; virüsler, kurtçuklar, Truva atları, mantık bombaları, tuzak kapıları, yetkisiz erişim saldırıları, pasif dinleme saldırıları, sosyal mühendislik, bakteriler, mesaj içeriğinin açığa çıkartılması, trafik analizi, kimlik değiştirme, tekrarlama, mesaj içeriğinin değiştirilmesi, hizmet engellemesi ve parola saldırıları sayılabilir. Bilgi güvenliğine yönelik tehditler hakkında ayrıntılı bilgi için bakınız [13] .

Bilgi güvenlik araçları olarak; şifreleme, antivirüs yazılımları, güvenlik duvarları, mesaj doğrulama, sayısal imza, özel sanal ağlar, vekil sunucular, saldırı tespit sistemleri vb araçlar kullanılmaktadır.

Bu yazıda önerilen teknik, bir tür mesaj doğrulama tekniği olmakla birlikte, uygulandığında programların ve dosyaların bütünlüğünü güvence altına alan, onlara truva atı veya benzeri virüslerin bulaşmadığını kanıtlar. Tipik olarak şöyle bir senaryo için bir önlem oluşturur.

İçeriden kötü niyetli ancak uzman bir kullanıcı bir dosyaya veya programa, zaman bombası veya mantık bombası türü bir truva atı yerleştirebilir. Bulaşmış program, truva atı aktif olmadığı sürece normal çalışacak, herhangi bir anormallik görülmeyecektir. Bu program hesap makinası gibi çok kullanışlı bir program olabilir. Truva atı aktive olup harekete geçince de artık çok geç olacaktır. Truva atı aktif hale geldiğinde saldırganlar sisteme tuzak kapıları ile ulaşabilir, bu sayede gizli bilgileri çalabilecekleri gibi silebilir, değiştirebilir ve hatta sistem kontrolünü ele geçirebilirler.

Bu truva atına karşı bir anti virüs programı var olabilir veya olmayabilir. Önemli olan, antivirüs programı olmasa bile bulaşıklık durumunu, truva atı (yani virüs) harekete geçmeden bulmaktır. Her an "*kullandığımız bu program orijinal halinde midir yoksa bütünlüğü bozulmuş mudur* ?" sorusuna kesin yanıt verebilmek gerekir. Bu yazıda anlatılan çalışma özellikle bu senaryo ve soruyu konu etmektedir.

3. MESAJ DOĞRULAMA VE ÖZCÜK FONKSİYONLARI

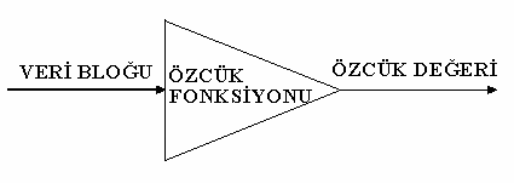
Mesaj veya dosya veya herhangi bir veri setinin ilk özgün halini koruduğu, eklenti, eksiltme veya hiç değişikliğe uğramadığı güvencesi verilmesi, genelde mesaj doğrulaması (authentication) olarak bilinir. Bu amaçla mesaj doğrulama kodu (MAC, message authentication code) kavramı ve algoritmaları geliştirilmiştir. Mesaj doğrulama kodu (Message Authantication Code - MAC), gizli anahtar kullanımı ile mesajın bir fonksiyonu olarak hesaplanan ve mesaja eklenen bir koddur. Bu yöntemde, sabit uzunluktaki bir kod mesaja eklenir. Mesaj üzerinde herhangi bir değişiklik yapılmaz ve mesajla birlikte üretilen bu kod gönderilir. Üretilen bu kod mesaj doğrulama kodu olup, mesajdan üretilmiş sabit büyüklükte bir koddur. Büyüklüğü genelde çok kısadır. MAC elde etmenin bir yolu, verinin bir yetkilinin sayısal imzasını taşımasıdır.

Dosya, resim, metin, mesaj gibi veri bloklarının özgünlüğünü korumasının diğer bir yolu da, özcük fonsiyonları kullanılmasıdır. Özcük fonksiyonları, herhangi uzunluktaki bir dosya, resim, metin, mesaj gibi veri bloklarını belirli bir özcük algoritmasına tabi tutarak, sabit uzunlukta sayısal bir özcük değeri üreten fonksiyonlardır. Herhangi uzunluktaki bir dosya,

resim, metin, mesaj gibi veri seti özcük fonksiyonu girdisi olabilir. Ancak çıktı özcük değeri sabit uzunluktadır.

Özcük algoritmaları sonucu ortaya çıkan özcük değeri, mesajın sayısal bir parmak izidir. "Bugün hava çok iyi" gibi bir ifadenin özcük değeri "h7tfd8Fr" olabileceği gibi, bilgisayarımızdaki "resim.gif" gibi bir resim dosyasının özcük değeride "a8jkdİ0" gibi bir değer olabilir. Özcük fonksiyonu ile oluşan özcük değeri anlamsız bir bilgidir. Özcük fonksiyonları tek yönlüdür. Bir diğer deyişle girdi verildiğinde çıktı tektir ve kolayca bulunur. Ancak bir özcük değeri verildiğinde fonksiyon ters çalıştırılıp onu üretmiş olan girdi bulunamaz. Ayrıca girdideki en küçük bir değişiklik çıktıyı olduğu gibi değiştirir.

Özcük fonksiyonlarının öncelikli kullanım alanı, kazaen, kasıt olmadan oluşmuş hataların farkına varmaktır. İletişim veya depolama hataları gibi doğal nedenlerden, veya yetkisiz bir işlem sonucu oluşmuş hatalardan dolayı içeriği değişmiş bir veri setinin doğrulaması önceden ve sonradan alınacak özcük değerlerinin karşılaştırması ile yapılabilir. (Şekil 3.1.).



Şekil 1. Dosya, resim, metin, mesaj gibi veri bloklarının özcük algoritmasına tabi tutulması işlemi

Bir veri bloğunu doğrulamanın, yani iletim sonucu, yetkisiz erişim ve depolama hataları vb. gibi sebeplerden dolayı içeriğin değişmediğinin kanıtlanması, veri bloğunun önceden bir özcük değerini yaratmakla başlar. Mesajın özcük değeri, alıcı tarafından tekrar üretilir. Eğer iki değer eşitse doğrulama sağlanmıştır.

İyi bir özcük fonksiyonunun temel özellikleri olarak şunları sayabiliriz.

1. Özcük algoritması, her hangi bir uzunluktaki veri setinde çalışabilmelidir.
2. Çıktı, yani özcük değeri sabit uzunlukta olmalıdır. Bu uzunluğun veya özcük değeri uzayımızın yeterli büyüklükte olması uygulama durumuna göre ve aşağıdaki koşulların yerine getirilmesi için zorunludur. Örneğin özcük değerinin 128 bit olması, (özcük uzayının 2^{128} olması) yeterli görülebilir. Bazı uygulamalarda bunun 160, 192 veya 256 bit olması istenebilir.
3. Verilen herhangi bir girdi değeri x için, özcük değğeri H(x) kolayca hesaplanabilmelidir.
4. Özcük algoritmalarından elde edilen özcük değerinden, orjinal veriyi elde etmek mümkün olmamalıdır. Tek yön özelliği olarak tanımlanabilen bu özellik enformasyon teorisine göre genelde kendiliğinden oluşur.
5. Bir mesaj M ve özcük değeri H (M) verildiğinde, aynı özcük değerini veren bir başka mesaj M' kolayca bulunamaz. H(M) = H(M') ilişkisini sağlayacak M' bulunamaz demek, algoritmik yolla veya makul bir zamanda bulunaz demektir. Geriye kalan tek yol kaba kuvvettir.
6. Aynı özcük değerini veren iki ayrı mesaj bulunamaz. Ayni biçimde bunun algoritmik bir yolu yoktur. Tek yolu olan kaba kuvvetin de yeterli zamanda yanıt vereceği şüphelidir.

Yukardaki iyi özcük algoritması olma koşullarını yerine getiren bazı algoritmalar ortaya çıkarılmış ve standart sayılabilecek kullanıma girmişlerdir. Bunlara örnek olarak SHA-1 (Secure Hash Algorithm-1), MD2 (Message Digest 2), MD5 (Message Digest 5) gibi algoritmalar gösterilebilir.

Özcük fonksiyonları, elektronik postalar, elektronik fon transferi, yazılım dağıtımı, veri saklanması, veri bütünlüğü garantisi, veri kaynağı doğrulaması, Web güvenliği kapsamında oturum doğrulama (session authentication), dosya sistemi bütünlük kontrolü (File System Integrity Checking), mesaj doğrulama, IP güvenliği (IP Security) ve sayısal imza da kullanılmaktadır.

4. SHA-1 ÖZCÜK ALGORİTMASI

Amerika Birleşik Devletleri Hükümeti, Federal Bilgi İşlem Standartları Yayımları'ndan (Federal Information Processing Standarts Publications - FIPS) birisi olan "FIBS PUB 180-1" numaralı yayım, SHA özcük algoritmasının (FIPS 180) teknik olarak yeniden gözden geçirilip düzeltilmiş bir sürümü olup, SHA-1 özcük algoritması hakkında teknik bilgi vermektedir (http://www.itl.nist.gov).

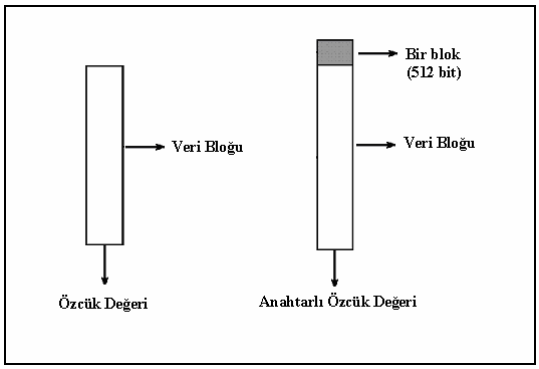
Sayısal imza standartı (Digital Signature Standart - DSS) gereğince, SHA-1 özcük algoritması, hem gönderici ve hem de alıcı tarafından , sayısal imzayı hazırlamak maksatlı ve doğrulama maksatlı olarak kullanılabilir. SHA-1 algoritması, verilen bir mesajın özcük değerini hesaplar. Bu değere karşılık gelen bir mesajın veya veri bloğunun bulunması veya aynı mesaj özcük değerini üretecek iki farklı mesajın bulunabilmesi olası değildir. SHA-1 özcük algoritması ile ilgili bilgilere "**http:// www.itl.nist.gov / fipspubs / fip180-1.htm"** kaynağından ulaşmak mümkündür.

Bu yazıda, SHA-1 algoritmasının, arkasından sayısal imza uygulaması gelmese de anahtarlı bir uygulama ile mesaj bütünlüğü amacına hizmet edebileceği gösterilmektedir.

5. ANAHTARLI ÖZCÜK DEĞERİ YÖNTEMİ

Güvenlik açısından bakıldığında tek başına bir özcük algoritması içerden uzman saldırıya karşı koymakta yetersiz kalır. Çünki organizasyon içi kullanımdaki bir özcük fonksiyonunu uygulaması bu kötü niyetli uzmana da açıktır. Bu uzman sisteme bir truva atı koyduktan sonra yeni özcük değerlerini üretebilir ve değiştirebilir. Burada açıklanan anahtarlı özcük metodu bu sakıncanın önüne geçmektedir. Anahtarlı Özcük Yöntemi özetle şu şekilde çalışmaktadır.

Normal olarak özcük fonksiyonları, veri bloklarını bit bazında okumakta ve özcük fonksiyonunun çalışma algoritmasına göre özcük değeri üretmektedirler. Bu yöntemde ise veri blokları yine okunmaktadır. Ancak okunan bu veri bloklarının üstüne, anahtar olarak belirlenen yeni bir blok eklenmektedir. Eklenen ve bu uygulamada 512 bitten oluşan bu veri bloğu, sistem güvenlik sorumlusunun rassal bir biçimde ürettiği, tahmin edilemeyen, sadece kendisinin istediği ortamda sakladığı, anahtar olarak nitelendirilebilecek bir veri bloğudur (Şekil 4.1.).



Şekil 2. Özcük değeri ve Anahtarlı özcük değeri

SHA-1 algoritması analizlerinden, mesajdaki herhangi bir değişikliğin özcük değerini tamamen ve tahmin edilemeyecek biçimde değiştirdiği, özcük değerine bakarak geri dönüşümün (mesajdaki değişikliği bulmanın) mümkün olmadığı bilinmektedir. Bu durumda anahtar olarak eklenen 512 bit blok, özcük değerini tahmin edilemeyecek biçimde değiştirecektir. Özcük anahtarının uzunluğu 512bitten küçük de olabilir. Ancak özcük uzunluğu olan 160 bitten küçük olamaz ve bu değerin küçük olmamasında yarar vardır.

Kötü niyetli bir saldırgan, mesajda bir değişiklik yaparsa, doğrulama için kullanılan özcük değerini de değiştirmek zorundadır. Eğer özcük değeri düz (anahtarsız) hesaplanıyorsa uzman bir iç saldırgan değiştirdiği mesaja uyan özcük değerini de doğru olarak hesaplayabilir ve böylece mesaj değişikliği açığa çıkmaz. Ancak özcük hesaplanmasında anahtar kullanılırsa ve bu anahtar sadece güvenlik sorumlusunda bulunursa, o zaman kötü niyetli kişi değiştirdiği mesaj için doğru özcük değerini üretemez ve yaptığı değişiklik ilk kontrolde farkedilir. Kötü niyetli kişinin anahtarı doğru tahmin etme olasılığı 2^{-512} dir. Anahtarlı özcük değerini doğru tahmin etme olasığı da 2^{-160} dır. Bu değerler hemen hemen tüm uygulamalarda yeterli güvenlik sağlarlar. Yapılan uygulamada anahtar değeri 512 bit alınmış olsa da, güvenlik açısından bu değer 160 bit üstünde herhangii bir uzunlukta olabilir.

Tipik anahtarlı özcük değeri uygulaması şöyledir:

Sistem yöneticisi veya sistem güvenlik sorumlusu, sadece kendinin sahip olduğu 512 bit uzunluğunda bir rastgele bit dizisi oluşturur. Özcük anahtarı diyebileceğimiz bu dizi, bir küçük "flash memory" veya bir disket üzerinde bulunabilir. Ama sistemde başkalarının okuyabileceği bir yerde bulunamaz. Sorumlu kişi, bütünlüğü garanti altına alınmak istenen her veri parçası (dosya, vb.) için bir anahtarlı özcük değeri üretir ve bunları saklar. Değişmemesi gereken bu veri parçaları için istenen zaman aralıkları ile yine anahtarlı özcük değeri üretir. Eğer önceki anahtarlı özcük değeri ile yeni değer aynı ise veri bütünlülüğü devam etmektedir. İki değer değişikse veri bütünlülüğü bozulmuştur.

Bu uygulamanın bir başka yararı da antivirüs programı henuz bulunmayan zaman veya mantık bombalarına karşı etkin olmasıdır. Bilgilere bilinen tip virüsler bulaştığında çözüm mümkün olurken, daha henüz antivirüs programı olmayan değişik bir virüs bulaştığında, çözüm konusunda problem bulunmaktadır. Anahtarlanmış özcük değeri ile bu problem de aşılmaktadır. Henüz imzası tespit edilmemiş virüs veya belli bir zamanda aktif olmak üzere sisteme bulaşmış zaman bombası, orijinal veriye ait olan anahtarlı özcük değerinin değişmesine neden olacağından ilk kontrolde ortaya çıkacaktır.

6. SONUÇ VE DEĞERLENDİRME

Bilgi güvenliğinin vazgeçilmez unsurlarından ilki, bilgi bütünlüğü ve bunun doğrulamasıdır. Bilgi bütünlüğünü bozacak iç ve dış saldırılara karşı antivirüs programları, düz özcük değerleri veya klasik yöntem diyebileceğimiz Mesaj doğrulama kodu (Message Authantication Code – MAC) yöntemi kullanılabilse de, bu yöntemler bazı durumlarda yetersiz kalabilmektedir. Çünkü antivirüs programları yanlız bilinen virüslere karşı etkilidir. Düz özcük değerini ise içerideki saldırgan da üretebilir ve sistemi aldatabilir. Klasik yöntem olan MAC ise, RSA benzeri bir yöntemle imza gerektirdiğinden dolayı çok zaman almakta ve performansı düşürmektedir. Bu sorunlara karşı anahtarlı özcük değeri uygulaması önerilmektedir.

Normalde özcük fonksiyonları, veri bloklarını bit bazında okumakta ve özcük fonksiyonunun çalışma

algoritmasına göre özcük değerini elde etmektedirler. Anahtarlı özcük yönteminde ise, yine veri blokları okunmaktadır. Okunan bu veri bloklarının üstüne, anahtar olarak belirlenen bir blok, yani 512 bit olan bir veri bloğu eklenmektedir. Eklenen ve 512 bitten oluşan bu veri bloğu, tamamen rassal üretilen sadece kullanıcının bildiği ve istediği ortamda sakladığı, anahtar olarak nitelendirilebilecek bir veri bloğudur. Böylece bütünlüğü güvence altına alınmak istenen verilere 2^{-160} düzeyinde bir güvenlik sağlanmış olur.

Organizasyon içerisinden yapılan saldırılar sonucu, hem veriler ve hem de bu verilere ait düz özcük değerlerini değiştirmek mümkün iken, bu anahtarlanmış özcük değeri için geçerli değildir. Bu tip saldırılarda, aynı verilere ait daha önceden hesaplanmış anahtarlı özcük değerleride değişeceğinden dolayı, değişiklik fark edilebilecektir. Burada dikkat edilmesi gereken nokta, anahtarlanmış özcük değerinin sadece anahtar sahibi olan sistem güvenlik sorumlusu tarafından üretilebileceğidir. Fakat, verilere ait düz özcük değeri ise, özcük algoritmasına sahip olan içeriden kötü niyetli kullanıcı tarafından da kolaylıkla üretilebilir. Aynı şekilde, organizasyon dışarısından birileri bilgi sistemlerine nüfuz etmeyi başararak verilerde ekleme, silme, vb. değişiklikleri yaparsa, sistem yöneticisi tarafından daha önceden hesaplanmış anahtarlı özcük değerleri de değişeceğinden dolayı, değişiklik fark edilebilecektir. Bu çalışma ile ilgili daha ayrıntılı bilgi için bakınız [1]

KAYNAKLAR

[1] Aydın, H. 2003. Bilgi Güvenliği ve Özcük Fonksiyonu Uygulamaları, Yüksek Lisans Tezi, Hava Harp Okulu, HUTEN, Istanbul

[2] Çölkesen, R. 2001. *Network, TCP/IP, UNIX*. Papatya Yayınevi

[3] Edwards, L and Waelde, C. 2000. *Law and the Internet: a framework for electronic Commerce*. Hart Publishing, Oxford

[4] Grant, L. 1997. *Understanding Digital Signatures: Establishing Trust over the Internet and Other Networks.* McGraw-Hill.

[5] Kephart, J., Sorkin, G., Chess, D., and White, S. 1999. *Fighting Computer Viruses*.

[6] Menezes , A., Van, P., S. Vanstone. 1996, *Handbook of Applied Cryptography*. CRC Press

[7] Naor, M. 1989. Universal One-way Hash Functions and their Cryptographic Applications. *Proceedings 21th Annual ACM Symposium on Theory of Computing*, 1989, pp. 33-43

[8] Digital Signature Standard (DSS), NIST FIPS Publication. 1994 NIST, Washington DC

[9] Secure Hash Standard (SHS), NIST FIPS Publication. 1995 NIST, Washington DC.

[10] Örencik, B. ve Çölkesen, R. 2002. Bilgisayar Haberleşmesi ve Ağ Teknolojileri. İstanbul: Papatya Yayıncılık.

[11] Preneel, K. 1996. Hash Functions Based on Block Ciphers and Quaternary Codes, Advances in Cryptology.

[12] Saka, Y. 2000, Bilgisayar Ağ Güvenliği ve Şifreleme. Yüksek Lisans Tezi. Muğla Üniversitesi, MUĞLA.

[13] Stallings, W. 1999. Cryptography and Network Security : Principles and Practice. Prentice Hall NJ.

[14] Stalings, W. 2000. Data and Computer Communications. Prentice Hall NJ.

[15] Stallings, W. 2000. Network Security Essentials. Prentice Hall NJ.

[16] Stephen, N. 2001, Network Intrusion Detection. Indianapolis, Ind. : New Riders.

[17] http://www.cisco.com/go/security/

[18] http://www.cert.org/

[19] http://www.computer.org/

[20] http://www.ieee.org/

[21] http://www.itl.nist.gov/fipspubs/fip180-1.htm Secure Hash Standart.

[22] http://www.itl.nist.gov/fipspubs/index.htm/ Federal Information Processing (FIPS) Standards Publications.

[23] http://www.Symantec.com/

ÖZGEÇMİŞLER

Yzb. Hakan AYDIN

1971 yılında Edirne/Havsa'da doğdu. İlkokulu ve ortaokulu Dazkırı/AFYON'da tamamladı. 1985 yılında Kuleli Askeri Lisesi'ne girdi. Kuleli Askeri Lisesi'nden 1989 yılında mezun oldu. Aynı yıl Kara Harp Okulu'na girmeye hak kazandı. Kara Harp Okulu Elektronik Mühendisliği bölümünden 1993 yılında Teğmen olarak mezun oldu. Piyade Subayı olarak çeşitli kıta görevlerinde bulundu. 1999–2000 yılları arasında Ege Üniversitesi Bilgisayar Mühendisliği'nde, 10.Dönem OBİ Subay Temel Kursu'nu bitirdi. Ekim 2001 yılında, Kara Kuvvetleri Komutanlığı Bilgi Sistemleri Dairesi'nde görev yaptığı sırada, Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsü Bilgisayar Ana Bilim Dalında Yüksek Lisans öğrenimi yapmaya hak kazandı ve 2003 yılında buradan mezun oldu. Halen Kara Kuvvetleri Komutanlığı Bilgi Sistemleri Dairesi'nde görev yapmaktadır.

Prof.Dr. Fuat İNCE

Akademik Unvanı :Prof. Dr.
Aldığı Dereceler :BS,Elektrik Müh.,Boğaziçi Ünv., 1968.
MS,Elektrik Müh., Illinois Ünv. 1969

PhD, Elektrik Müh., Illinois Ünv., 1973

İlgi Alanları : Yazılım Mühendisliği, Bilgi Teknolojileri, Görntü İşleme, Uzay Teknolojileri

Verdiği Dersler : (HUTEN'de)

- Yazılım Mühendisliği İleri Konular
- Uzay Teknolojilerine Giriş

Deneyimler :

- Halen Maltepe Üniversitesi, İstanbul, Bilgisayar Mühendisliği Bölüm Başkanı.
- Bilgi Teknolojileri ve Araştırma Enstitüsü Kurucu Başkanı (erken emeklilik nedeniyle görevden ayrıldığında, 120 çalışanıyla enstitü, milyonlarca dolarlık birçok uluslararası araştırma projesini yürütüp başarıyla tamamlamıştı).
- NATO Bilgi Teknolojileri Paneli üyesi (1998-2001).
- Birçok ulusal ve uluslararası komite görevlerinde bulunmuştur.
- Birçok profesyonel topluma üye.