İNTERNET'TE GÜVENLİK VE SALDIRI SEZME SİSTEMLERİ

Yusuf Levent ŞAHİN

Yüksek Lisans Tezi

Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği-Bilişim Ana Bilim Dalı

Ağustos-2005

JÜRİ VE ENSTİTÜ ONAYI

Yusuf Levent ŞAHİN'in "**İnternet'te Güvenlik ve Saldırı Sezme Sistemleri"** başlıklı **Bilgisayar Mühendisliği-Bilişim** Anabilim Dalındaki, Yüksek Lisans tezi ..... tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	**İmza**
Üye (Tez Danışmanı)	**: Prof. Dr. Yaşar HOŞCAN**	**.....**
Üye	**: Prof. Dr. Can AYDAY**	**.....**
Üye	**: Yard. Doç. Dr. Yusuf OYSAL**	**.....**

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun …… tarih ve …….. sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

İNTERNET'TE GÜVENLİK VE
SALDIRI SEZME SİSTEMLERİ

Yusuf Levent ŞAHİN

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı-Bilişim

Danışman: Prof. Dr. Yaşar HOŞCAN

2005, 89 sayfa

Bu çalışmada, İnternet'te güvenlik kavramı ve yeni bir güvenlik teknolojisi sayılabilecek saldırı sezme sistemleri işlenmiştir. Bunun için bilgi güvenliği ile ilgili kavramlar açıklanmış ve İnternet bağlantısına sahip sistemlerde bilgi güvenliğinin karşı karşıya olduğu tehditler anlatılmıştır. Sistemlerin ve bilgilerin İnternet'ten gelebilecek olan tehditlere karşı güvenliğinin sağlanması için kullanılan araçlara ve tekniklere değinildikten sonra saldırı sezme sistemleri ayrıntılı bir şekilde incelenmiştir. Son olarak bir saldırı sezme sistemi yazılımı başarımının belirlenmesi amacı ile uygun bir ağ ortamına uygulanarak test edilmiştir.

Anahtar Kelimeler: **İnternet'te güvenlik, bilgi güvenliği, saldırı sezme sistemleri.**

ABSTRACT

Master of Science Thesis

SECURITY ON THE INTERNET AND INTRUSION DETECTION SYSTEMS

Yusuf Levent ŞAHİN

Anadolu University
Graduate School of Sciences
Information Technology Program

Advisor: Prof. Dr. Yaşar HOŞCAN

2005, 89 pages

In this study, the concept of security on the internet and intrusion detection systems, considered a new security technology, are mentioned. For this reason the concepts about information security are explained and threats of information security on the systems that has internet connection are described. After explaining the tools and methods used to protect systems and information from the threats coming over the internet, intrusion detection systems are studied in detail. Finally, an intrusion detection system was tested to determine its performance in a suitable network environment.

Keywords: **Security on the internet, information security, intrusion detection systems.**

TEŞEKKÜR

Değerli katkılarından dolayı tez danışmanım Prof. Dr. Yaşar HOŞCAN'a, yardımlarından dolayı Mustafa Murat TOPÇU ve Hurşit Cem SALAR'a teşekkür ederim.

Yusuf Levent ŞAHİN

Ağustos-2005

İÇİNDEKİLER

TABLOLAR DİZİNİ

ŞEKİLLER DİZİNİ

1. GİRİŞ

İnternet hızla yaygınlaşarak hayatın bir çok alanında kullanılır hale gelmiştir. Kişiler, kurumlar ve organizasyonlar İnternet'i bilgiye ulaşma ve iletişim amaçlı olarak yoğun bir şekilde kullanmaya başlamışlardır. Bilginin gizliliğinin ve bütünlüğünün son derece önemli olabildiği günümüzde veriler kimi zaman ülkeler arasında kablolar üzerinde akmaktadır. Önemli bilgilerin bulunduğu pek çok sistem fiziksel anlamda çok uzakta olsalar dahi saniyeler içinde ulaşılabilir haldedirler. Bu durum, bir çok problemi de beraberinde getirmektedir. Bahsedilen problemlerden en büyüğü bilginin güvenliğinin sağlanmasıdır. Bu tezin amacı da bilgi güvenliğini tehdit eden unsurların ve bu tehditlerin en aza indirgenmesine yönelik uygulamaların incelenmesidir.

Tezin ikinci bölümünde, İnternet'te güvenlik sorununa değinilecek, güvenlik anlamında korunacak olan unsurlar belirtilecek ve kullanılan güvenlik stratejileri hakkında bilgi verilecektir.

Üçüncü bölümde, İnternet bağlantısının getirdiği tehditler üzerinde durulacaktır. Bu bağlamda öncelikle yaygın olarak kullanılan saldırı yöntemlerine örnekler verilecek daha sonra da yayılmak için çoğunlukla İnternet ortamını kullanan virüsler, Truva atları ve solucanlar gibi zararlı yazılımlar işlenecektir.

Dördüncü bölümde, üçüncü bölümde değinilip örnekleri verilen tehditlerden korunmak için kullanılan güvenlik duvarları, veri şifreleme teknikleri ve antivirüs yazılımları hakkında bilgiler verilmiştir. Saldırı sezme sistemleri de bu bölüm altında yer alabilecek olsa da daha ayrıntılı bir şekilde incelenmek üzere bir sonraki bölüme bırakılmıştır.

Yukarıda da bahsedildiği gibi tezin beşinci bölümü, saldırı sezme sistemlerinin ayrıntılı bir şekilde incelenmesine ayrılmıştır. Bu bağlamda öncelikle saldırı sezme sisteminin tanımı yapılmış, konu ile ilgili genel kavramlar verilmiş, saldırı ve saldırı sezme terimlerine değinilmiş, kullanılma nedenleri anlatılmıştır. Saldırı sezme sistemlerinin çeşitli açılardan sınıflandırılması yapıldıktan sonra verimliliklerini belirleyen unsurlar belirtilmiştir. Daha sonra da

maruz kalabilecekleri üç önemli saldırı örneği verilmiştir. Son olarak saldırı sezme sistemlerinin geleceği ile ilgili yorumlara yer verilmiştir.

Altıncı bölümde ise bir saldırı sezme sistemi yazılımı uygun bir ağ ortamına uygulanmış ve saldırı sezme sistemlerinin verimliliğini belirleyen unsurlar baz alınarak başarım değerlendirmesi yapılmıştır.

2. İNTERNET'TE GÜVENLİK

Tezin bu ilk bölümünde öncelikle İnternet'te güvenlik sorununun oluşumu ve nedenlerine değinilmiş, ardından da bu açıdan korunması gereken unsurlar ele alınmıştır. Bu şekilde İnternet'te güvenliği sağlamanın amacı ve gerekli olma nedenleri ortaya konulmaya çalışılmıştır. Daha sonra bu sorunun çözümü için kullanılan yaklaşımlar ve stratejiler hakkında bilgi verilmiştir.

2.1. İnternet'te Güvenlik Sorunu

İnternet'in doğuşu ve gelişimi, ulaştığı noktalar ve üstlendiği görevler göz önüne alındığında, tüm bu olanların aslında kısa bir süre zarfında gerçekleştiğini söylemek mümkündür. Özellikle kullanımının 1985' ten sonra büyük yaygınlık göstermesi, bazı konularda hazırlıksız yakalanılmasına, bunun sonucunda da gerekli alt yapı çalışmaları yapılmadan çözümlerin üretilmesine sebep olmuştur. Bu da tabi en başta bu konularda standartların oluşturulmasına yol açmıştır. Bu konuların başında da güvenlik gelmektedir.

Güvenlik, hemen her türlü bilgisayar ağında düşünülmesi gereken unsurların başında gelirken, İnternet gibi herkesin kullanımına açık bir ortam için çok daha önemli şeyler ifade etmektedir. İster resmi, ister ticari olsun, kuruluşların büyük bir kısmı İnternet'e ulaşmak, İnternet üzerinde kendilerine ait sayfaları görmek ve bazı bilgileri insanlara bu mekanizma aracılığıyla ulaştırmak istemektedirler. Hatta bazı durumlarda, şirketin değişik yerlerde bulunan birimlerini kapsayacak bir İntranetin kurulması düşünülürken bile, İnternet'ten yararlanılması söz konusu olmaktadır. Ancak yukarıda da belirtildiği gibi esasında tüm bunlar, bir miktar risk alınması anlamına gelmektedir. Çünkü, İnternet ortamı yeterince güvenli bir ortam değildir. Buna rağmen kuruluşlar değişik amaçlar için İnternet'i kullanmaktadırlar. Bu bazen aktarılan verinin duyarlılığının üst düzeyde olmamasından bazen de, değişik güvenlik mekanizmalarını bir arada kullanmak suretiyle, yapılan çalışma için yeterli güvenilirlikte bir ortamın sağlanmış olmasından kaynaklanmaktadır.

İnternet üzerinden gerçekleştirilecek bilgi transferleriyle ilgili olarak yaşanabilecek güvenlik sorunlarını kabaca şöyle gruplayabiliriz;

- Gizli bilgilerin, üçüncü kişiler tarafından öğrenilmesi
- Transfer edilmekte olan bilgilerin, başkalarınca değiştirilip yanıltıcı bilgilerin hedefe ulaşmasına neden olunması,
- İletişim halinde bulunulup bilgilerin karşılıklı olarak alınıp verileceği durumlarda, karşıdaki bilgisayarın kimliğinden emin olunamaması.

Bunun yanında, transfer edilmiyor olsa da, İnternet üzerinden ulaşılabilir bilgilerin istenmeyen kişilerin eline geçmesi, silinmesi veya zarar görmesi, yine yaşanabilecek sorunlardır. Ayrıca, gerek donanım gerek yazılım kaynaklarının, yine istenmeyen ve izin verilmeyen kişilerce, İnternet'in sunduğu olanaklardan faydalanmak suretiyle kullanılması, yaşanabilecek sorunlardır. Bütün bunlar da, güvenlik konusundaki eksikliklerin nelere sebep olabileceği konusunda fikir verebilir. Hele de kullanılan bilgilerin ve yapılan işin hassasiyeti arttıkça, güvenlik konusunun da önemi artacaktır [1].

2.2. İnternet'te Güvenlik Açısından Korunacak Unsurlar

İnternet'e bağlanıldığında üç şey riske atılmış olur. Bunlar;

- Veriler: Yerel ağda bulunan ve İnternet üzerinden ulaşılabilir bilgisayarlar üzerinde bulunan veriler
- Kaynaklar: İnternet üzerinden ulaşılabilir olan bilgisayarların kendileri.
- Saygınlık: Faaliyet gösterilen alana bağlı olmak üzere, ticari veya toplumsal saygınlık.

2.2.1. Veriler

Ağ güvenliğinde verilerin üç önemli unsuruna dikkat edilmelidir. Bunlar gizlilik, bütünlük ve kullanıma hazırlıktır [2].

Gizlilik

Gizlilik, bilginin sadece bilgiye erişim hakkı olan kişilere açık tutulmasıdır. Sadece bilgiye erişim hakkı olan kişilerin bilgiye erişmesini sağlayacak bir sistem, çok katı kontrollerin konulmasını gerektirir. Kişiler güvenilir ve gizli ilgilerine, sadece yaptıkları işte kullanılması gerektiği zaman erişmelidirler. Bilgi ve kaynaklara erişim hakkının sadece ihtiyaç duyanlara verilmesi kavramına erişim kontrolü denir.

Erişim kontrolünü sağlamada kullanılan, en yaygın yöntem şifre kullanımıdır. Şifrenin çaldırılması ise en yaygın olarak bilinen güvenlik gediğidir. Şifre güvenliğini sağlamak için akıllı kart ve tek kullanımlı şifre aletleri kullanmak gereklidir. Bu kaynakları yetkisiz erişime karşı engelleme konusunda alınacak önlemlerin ilk aşamasıdır.

Kuvvetli bir şifre politikası oluşturmak çok zor bir şey olmamakla birlikte kesinlikle oluşturulmalı ve çalışanlarda bu konuda eğitilmelidir.

Erişim kontrolünün diğer bir tarafı, kurumun bilgisayar ağlarında kimliği doğrulanan kurum çalışanının kaynak erişim yetkilerine sınırlamaların getirilmesidir. Örneğin, tüm insan kaynakları çalışanları, kişi ve kurum elemanları ile ilgili adres ve doğum günü bilgilerine erişim yetkisine sahip olabilir; fakat tazminat bilgilerine sadece bir kısım yetkili kişiler erişebilir. Aynı zamanda, bazı kullanıcılara bu bilgiler üzerinde sadece görüntüleme yetkisi verilirken, diğer kullanıcılara hem görüntüleme hem de güncelleme yetkisi verilecek durumlarla da karşılaşılabilir. Bu tipik bir erişim denetleme senaryosudur.

Bütünlük

Bütünlük ölçütü, bilginin beklenmeyen yollarla değiştirilmemesini garanti eder. Bütünlük kaybına insan hataları veya kasıtlı kurcalamalar neden olur. Şüphesiz ki, herhangi bir işte yanlış bilgilerin kullanılması istenmeyen sonuçlar doğurabilir. Uygunsuz bir şekilde değiştirilmiş verinin işe yaramaz veya tehlikeli olması gerçeği, verinin doğruluk ve tutarlılığının korunması için, çok büyük çabanın sarf edilmesini gerektirmektedir.

Verinin geçerliliği çok önemli ise bunu garanti edecek kontrollerin tasarlanması ve verinin doğruluğunun kontrol edilmesi gereklidir. Eğer veri bir şekilde çalınmışsa veya yetkisiz kişiler tarafından değiştirilmişse, bunun tespit edilmesi sistemin bütünlüğü açısından elzemdir. Şifreleme, verinin yetkisiz kişiler tarafından erişilmesi ve değiştirilmesini, veriyi gizli bir biçime çevirerek, engellemede kullanılan bir yöntemdir.

Olgunlaşmış bir bilgi güvenliği politikası, etkin önlem alma ve tepki oluşturma işlevlerini içermelidir. Etkin önlem alma, kuvvetli güvenlik denetimlerinin konulmasını gerektirirken, tepki oluşturma bu kontrollerin kayıtlanması ve izlenmesini gerektirmektedir. İyi bir sistem yöneticisi, sistem içerisindeki hareketleri kayıt dosyalarından izleyerek, gerçekleşen olaylardan sonuçlar çıkarıp, yeni önlemler oluşturmalıdır.

Kullanıma Hazırlık

Kullanıma hazırlık, kaynakların silinmesi veya erişilemez hale gelmesini engelleyen ölçüttür. Bu ölçüt, sadece bilgi değil, diğer tüm teknoloji altyapısı ve ağa bağlı makineler için geçerlidir. Bu gerekli kaynakları kullanamama durumu, hizmet dışı kalma olarak adlandırılır. Kasıtlı saldırıların çoğu, veri çalmaktan çok bu tip olup, politik veya ekonomik sebepli olarak başlatılır. Elektronik posta adreslerinin protesto amaçlı olarak yasa dışı gereksiz postalarla doldurulması veya bir İnternet bankacılık sitesinin hizmet dışı bırakılması gibi biçimleri ile karşılaşılır.

Bilgisayar ağının veya sistemin fiziksel güvenliğinin sağlanması, hazır bulunabilirliği sağlama yöntemlerinden biridir. Önemli makine ve veri kaynaklarına fiziksel erişim kısıtlanarak, sistemin hazır bulunabilirlik sorunu azaltılabilir. Aynı şekilde elektronik bilgi ağı dışarıdan gelecek tehlikelere karşı, güvenlik duvarı olarak adlandırılan ürünlerle korunmalıdır. Güvenli geçit, iki bilgisayar ağı arasında hangi tip verinin akacağını düzenler ve kısıtlar.

Diğer yönden hazır bulunabilirlik, kaynakların ihtiyaç duyulduğu zaman ve yerde kullanılabilirliğini garanti etmelidir. Bu nedenle sistemin kırılması

durumunda anında geri dönmeyi sağlayacak yedek sunucuların ve veri yedeklerinin saldırganlardan arındırılmış bir ortamda, tüm güvenlik önlemleri alınmış bir şekilde tutulması gereklidir.

Bir kurumun güvenlik stratejisi yukarıda belirtilen üç ölçüt doğrultusunda değerlendirilmeli ve kurumun ihtiyaçlarına göre ölçütler değişik seviyelerde incelenmelidir. Örneğin milli savunma sistemlerinde güvenilirliğe daha çok önem verilmesi gerekirken, fon aktarım sistemlerinde bütünlüğe daha çok önem verilmelidir [3].

2.2.2. Kaynaklar

İnternet'e bağlanarak riske giren unsurlardan biri bilgisayar kaynaklarıdır. Zarar verilmeyecek olsa dahi bu kaynaklar başkaları tarafından kullanılmamalıdır [2]. Başka insanların, bir kuruluşa ait bilgisayardaki sabit diskte yer alan boş alanları kendi amacı için kullanmak istemesi, her ne kadar mevcut verilere zarar vermeyecek bir şey de olsa istenecek bir şey değildir. Bunun gibi diğer kaynakların da (işlemci zamanı, bellek, ...) başkaları tarafından kullanılması, kaynakların gerçek sahibi tarafından kabul edilecek bir durum değildir [1].

2.2.3. Saygınlık

İnternet'e açılan herkesin ve kurumların saygınlığının İnternet üzerinde korunması önemlidir. Oluşabilecek güvenlik problemleri kişi ve kurumların aleyhine olup kötü etkiler doğurabilir. İnternet üzerinde bir kurum ya da kişi adına izinsiz işlem yapan bir kişi, başka bir kişinin adını ya da ünvanını kullanır. Zarar verme, kötüye kullanma durumunda asıl kişinin saygınlığı zarar görecektir.

İnternet üzerinden dünyaya açılmayı düşünen kişi ya da kurumların, güvenlik politikası içinde, saygınlığının korunması için kişilere düşen güvenlik tedbirlerini alması gerekir. Ve düzenli olarak bu tedbirlerin izlenmesi şarttır [2].

2.3. Güvenlik Konusuna Yaklaşımlar

Bilgi güvenliği konusundaki yaklaşımlar şu şekilde listelenebilir:

- Güvenlik için önlem almama
- Belirsizlik yolu ile güvenliği sağlama
- Konak güvenliği
- Ağ güvenliği

2.3.1. Güvenlik için önlem almama

Güvenlik konusunda kullanılabilecek en basit ve kolay yol olarak bu yaklaşım verilebilir. Kullanılan işletim sistemi ve diğer ürünlerin basit olarak sağlamış olduğu güvenlik unsurlarının dışında bir çalışmanın yapılmadığı bu yaklaşım, ağ saldırılarının yoğun olarak yaşandığı günümüzde pek uygun olmayacaktır.

2.3.2. Belirsizlik yoluyla güvenliği sağlama

Bu güvenlik yöntemi, başkalarının sistem hakkında bir şey bilmemelerine, dolayısıyla neye nasıl saldıracaklarını kestirememelerine dayanır. Bu yönteme dayalı güvenlik yönteminin uzun süre sorunsuz çalışması nadir olarak görülür. Bu yöntemi benimseyenler genellikle küçük boyutlu çalışan, fazla tanınmayan kuruluşlardır. Dolayısıyla saldırganlar açısından iyi bir hedef olmayacaklarını düşünürler, bu açıdan da böyle bir güvenlik yönteminin kendileri için yeterli olacağını düşünürler. Oysa ki bazı saldırganlar için kırılan sistemin büyüklüğü veya ünü önemli değildir, kırılan sistemin sayısı önemlidir. Bu da bu tür sistemlerin onlar için kolay hedef olmaları sonucunu doğurur. Yeterli zaman ayrılarak, sisteme ilişkin bilgilerin bir saldırgan tarafından tahmin etme ve deneme yöntemleriyle ortaya çıkarılması mümkün olabilmektedir. Bu nedenle bu yaklaşım, güvenli bir çalışma ortamı sağlamaz.

2.3.3. Konak güvenliği

Belki de en çok kullanılan güvenlik yaklaşımı konak güvenliği yaklaşımıdır. Bu yaklaşımın esası, her bilgisayarın sahip olduğu güvenlik problemlerinin ayrı ayrı ele alınıp buna karşı önlem alınmasına dayanır. Bu yöntemin dezavantajı, bilgisayar sayısının artması durumunda yapılacak işlemin fazla zaman alacak olması, her bir bilgisayarla ilgili ayrı bir stratejinin gerçekleştirilmesinin zor olmasıdır.

Gerçekten de konak güvenliği, çokça kullanılan bilgisayar güvenlik yaklaşımı olmasına karşın, özellikle değişik yapılara sahip bilgisayarların bulunduğu ortamlar için kullanışlı olmayacaktır. Çünkü her bilgisayar kendi işletim sistemine sahip olacak, her işletim sistemi de değişik güvenlik problemlerini barındıracaktır. Bu da, bir konak için geliştirilmiş güvenlik mekanizmasının doğrudan diğer bilgisayarlara taşınmasına engel olacaktır. Tüm bilgisayarların aynı olup, aynı işletim sistemini çalıştırdığı düşünülse bile, işletim sisteminin değişik sürümleri için de aynı sorunlar geçerli olacaktır. Daha da ileri gidilerek, hem sitede yer alan tüm bilgisayarların aynı olduğu, hem de üzerlerinde aynı işletim sisteminin aynı sürümünün çalıştırıldığı durum ele alınsın. Bu durumda da, her bilgisayarın çalıştıracağı servislerin farklı olmasından kaynaklanacak güvenlik sorunları farklılıkları olacak, yine aynı güvenlik konfigürasyonunu tüm makinelere taşımak mümkün olamayacaktır.

Ayrıca konak güvenliği konusundaki başarı, bu bilgisayarları kullanacak kişilerin iyi niyeti ve yetenekleri ile doğrudan alakalıdır. Sistemdeki bilgisayar sayısının artması, bunları kullanacak kişi sayısını da arttıracak, bu da, bu konudaki kaygıların artmasına sebep olacaktır.

Konak güvenliği, nispeten küçük ve dolayısıyla bilgisayar sayısı az olan sistemler için uygun olabileceği gibi, ileri düzeyde güvenlik ihtiyacı duyulan yerlerde de kullanılabilir. Aslında tüm sistemler, kullandıkları güvenlik sistemi ne olursa olsun, bununla birlikte bir miktar konak güvenliğini de kullanmak zorunda kalacaklardır. Örnek olarak bir sonraki bölümde değinilecek olan Ağ Güvenliği yaklaşımının kullanıldığı bazı sistemlerde, bazı bilgisayarlar için daha ileri

güvenlik çalışmalarının yapılması gerekebilecek, bunun için de konak güvenliği yaklaşımından faydalanılabilecektir.

2.3.4. Ağ güvenliği

Sistemlerin büyümesi ve sistemi içerisindeki birimlerin farklı özelliklere sahip olması durumunda konak bazında güvenlik yaklaşımının kullanılması çok zor olacaktır. Bu da sistemlerin ağ güvenliği yaklaşımına dönmelerine sebep olmuştur. Bu yaklaşım, ağa ulaşmaların tamamı, ağ içerisinde yer alan farklı makinelere ve bunların sunduğu hizmetlere taleplerin bütünü kontrol altında tutar. Her bir makineyle ayrı ayrı ilgilenilmez. Bu yaklaşımda kullanılan ürünler; bir kuruluşa ait ağı, tüm ağdan ayıran güvenlik duvarları, mümkün olduğu kadar güçlendirilmiş kullanıcı belirleme mekanizmaları ve özel gizliliğe sahip olup da dış ağ üzerinden iletilmesi gereken bilgilerin başkaları tarafından ele geçirilip kullanılmasını engelleyecek olan şifreleme olarak sıralanabilir.

Ağ güvenliği yaklaşımı, güvenlik konusunda müthiş olanaklar sunabilmektedir. Bir güvenlik duvarı kullanmak suretiyle yüzlerce, binlerce ve hatta on binlerce bilgisayarın bulunduğu bir sistemi, bilgisayarlardaki konak güvenliği seviyelerini dikkate almaya gerek kalmadan dış dünyadan korumak mümkün olabilecektir [1].

2.4. Güvenlik Stratejileri

İnternet üzerindeki sistemlerin güvenliğini sağlamak için kullanılan yedi farklı strateji bulunmuştur. Bu stratejiler şunlardır:

- Gereksiz yetki vermeme
- Kademeli savunma
- Güvenlik kademelerinde farklı ürün kullanımı
- Denetim noktası
- En zayıf bağlantıyı baz alma
- Genel katılımı sağlama
- Karmaşık olmayan sistem yapısı

Bu bölümde bu stratejiler anlatılacaktır.

2.4.1. Gereksiz yetki vermeme

Gerektiğinden fazla ayrıcalık ve yetki vermeme, sadece bilgisayar ve ağ güvenliğinde değil, her türlü güvenlik çalışmasında öncelikle göz önüne alınan unsurlardandır. Bununla kastedilen, sistem üzerinde yer alan herhangi bir objenin, sadece kendisi ile ilişkilendirilmiş işlemi yapmak için gerekli olan yetki ve ayrıcalığa sahip olmasının gerektiğidir, daha fazlasına değil.

Konu İnternet üzerinde düşünülürse, şunlarla karşılaşılacaktır; tüm kullanıcıların, tüm İnternet servislerine ulaşmaları gerekmeyebilir. Bu durumda, kullanıcılara erişmeyecekleri servislere ilişkin yetki vermenin olumlu bir katkısı olmayacaktır. Buna karşılık burada değinilmekte olan güvenlik prensibine ters düşülecektir. Yine kullanıcıların tamamının sistem dosyalarını değiştirme ve hatta bazı durumlarda okumaları gerekmeyecektir. Kullanıcıları büyük bir kısmının normal çalışmalarını yapmaları için bilgisayarlarını süper kullanıcı şifresine ihtiyacı olmayacaktır. Hatta bazı durumlarda, sistem yöneticilerinin, tüm sistemin süper kullanıcı şifrelerini bilmesi gerekmeyebilecektir. Bir sistem, bir başka sistemde yer alan dosyalara ulaşma yetkisi olmadan da, gerekli işlemleri yapabiliyor olabilir. Tüm bu anlatılanların bir arada değerlendirilmesiyle, sistem üzerindeki her hangi bir objenin, gerekli çalışmalarını yapabilmesi için ihtiyaç duyacağı yetkiler belirlenmeli ve sadece bunlar o objeye verilmelidir.

Bazı durumlarda, objelere ayrıcalıklı durum vermek gerekebilir. Böyle durumlarda bile durum analiz edilerek, değişik çözümler aranmalıdır. Özellikle ayrıcalıklı yetki verilecek obje bir programsa ve bu program çok karmaşık bir yapıya sahipse, potansiyel bir saldırı hedefi durumuna gelecektir. Bu programa ulaşacak kişiler, bu programın sahip olduğu yetkilerden faydalanarak, pek çok şey yapabileceklerdir.

Gereksiz yetki vermeme konusunda çalışıyorken iki sorunla karşılaşılabilir. Bu sorunlardan biri, tasarlanmaları ve gerçeklenmeleri açısından böyle bir uygulamaya müsaade etmeyecek objelere sahip olunabileceğidir. Bir

program açısından olaya bakılırsa, programın yaptığı işler itibariyle süper kullanıcı yetkisi gerektirmiyor olmasına rağmen, başka bir yetki ile çalıştırmak mümkün olmayabilir [4].

2.4.2. Kademeli savunma

Genel anlamdaki güvenlik için de geçerli olacak bir diğer önemli güvenlik prensibi, bir birini yedekleyen, birden fazla kademeden oluşan bir güvenlik mekanizması yaratmaktır. Bu kademelerin mümkün olduğu kadar bir birleriyle bağlantısız olmaları sağlanarak, birinde meydana gelecek sorunun, diğer kademeleri etkilememesi de, göz önüne alınması gereken önemli unsurlardandır.

Güvenlik duvarları, ağ güvenliği konusunda çokça kullanılmalarına karşın, bunların aşılabilecekleri unutulmamalıdır. Buna önlem olarak da, ek bazı tedbirler alınmalıdır. Önemli olan bilgisayarda konak güvenliği'nin sağlanması, kullanıcıların eğitilerek, bunlardan kaynaklanacak sorunların en aza indirilmesi gibi tedbirler, güvenlik duvarının aşılması durumunda da, güvenliğin tamamen kaybolmasını önleyecektir.

2.4.3. Güvenlik kademelerinde farklı ürün kullanımı

Güvenlik mekanizmasını kademeli yapıyorken, her güvenlik kademesi için farklı bir ürünün ve yaklaşımın kullanılması, kademelerin birbirlerinden bağımsız olmalarını sağlayacak, bu da bir kademeyi aşmış bir kişinin, bir sonraki kademeyi kolaylıkla aşmasına engel olacaktır.

Değişik şirketler tarafından üretilmiş ürünleri bir arada ve güvenliğin farklı kademelerinin elemanları olarak kullanmak, bir üründe olabilecek hataların, diğer kademe tarafından telafi edilmesi imkanını verecektir. Tabi bu yaklaşım beraberinde bazı zorlukları da getirecektir [1].

2.4.4. Denetim noktası

Denetim noktası adı verilecek, gözlemenin ve kontrolün kolay olacağı bir dar kanal kullanılarak muhtemel saldırıları fark etmek ve önlemek daha kolay olacaktır.

Genel güvenlik yaklaşımlarında sıkça kullanılan bu yöntem, bilgisayar ağları ile ilgili güvenlik çalışmalarında da kullanılmaktadır. Güvenlik duvarları, korunması istenen ağ ile İnternet arasındaki tek geçiş noktasıdır. Dolayısıyla da, ağa saldırmak isteyen kişi kesinlikle güvenlik duvarı üzerinden geçmek zorundadır. Burada gerekli kontrolleri yapmak ve gerekli önlemleri almak yoluyla, güvenliği sağlayabiliriz.

Bu prensibin etkin bir şekilde uygulanması, gelebilecek saldırıların tamamının denetim noktası üzerinden geçecek olması ile mümkündür. Başka bir deyişle, eğer saldırgan saldırmak istediği ağa başka yerlerden dolaşarak da ulaşabilecekse, denetim noktasının ve orada kurulacak güvenliğin çok anlamı olmayacaktır [4].

2.4.5. En zayıf bağlantıyı baz alma

Güvenliğin temel tanımlarından biri; 'bir zincir en zayıf halkası kadar güçlüdür'. Saldırıda bulunmak isteyenler öncelikle savunmanın en zayıf halkasını bulmaya çalışacaklar, daha sonra da konsantrasyonlarını burası üzerine yoğunlaştıracaklardır. Bu nedenle de güvenlik sisteminin zayıf noktaları bulunmalı, buradaki güvenlik sorunları giderilmelidir. Eğer sorun giderilemiyorsa burası sürekli kontrol altında tutularak, olabilecek bir saldırıya anında müdahale edilmelidir. Güvenlik sorunları arasında bir ayrım yapılmamalı, zayıf olduğu düşünülen bir nokta üzerinde yoğunlaşılırken, diğer taraftaki problemler tamamen unutulmamalıdır [4].

2.4.6. Genel katılımı sağlanma

Kurulmuş olan güvenlik sistemi ne kadar iyi olursa olsun, çalışanların gerekli yardımı ve dikkati olmadığı sürece, güvenliğin sağlanmış olduğu söylenemez. Bu nedenle de güvenlik ile ilgili yapılan çalışmalarda tüm çalışanların katılımı sağlanmalı, güvenliğin gerekliliği kavratılmalıdır.

2.4.7 Basitlik

Basitlik iki nedenden dolayı güvenlik stratejileri arasında sayılabilir. Birincisi, basit nesneleri anlamanın kolay olmasıdır. Dolayısı ile de böyle

nesnelerin çalışmasının güvenli olup olmayacağını anlamak kolay olacaktır. İkincisi ise; karmaşık bir yapının, içinde hatalar barındıracak daha çok kısım içerebilecek olmasıdır [4].

3. İNTERNET BAĞLANTISI İLE GELEBİLECEK TEHDİTLER

Bu bölümde İnternet bağlantısı ile gelebilecek tehditler üzerinde durulacaktır. Söz konusu tehditler, yaygın saldırı yöntemleri ve genellikle kaynağı İnternet olan zararlı programlar olarak iki başlık altında incelenecektir.

Hem saldırı yöntemleri hem de zararlı programlar konuları üzerinde durulmasındaki amaç bu tezin konusu olan saldırı sezme sistemlerinin kullanımı ile önlenebilecek güvenlik sorunlarına örnek vermektir. Dolayısı ile bütün saldırı yöntemlerini veya bütün zararlı program örneklerini inceleme amacı taşınmamıştır.

Bu bölümde sadece bu tehditler üzerinde durulmuş ve saldırı sezme sistemleri ile önlenmeleri konusu bölüm 4'e bırakılmıştır.

3.1 Yaygın Saldırı Yöntemleri

Saldırı yöntemleri araştırıldığında yaygın olarak kullanılan yöntemlerin ağ paketlerini dinleme, ip taklidi, şifre saldırıları, "fragmentasyon" saldırıları, "ortadaki adam" saldırıları, "hizmet dışı bırakma" saldırıları, tcp sıra numarasının tahmini, kaynak yönlendirme, icmp saldırısı, uygulama katmanı saldırıları olduğu görülmüştür. Bunların dışında kullanılan pek çok yöntem olduğu görülmüş ancak burada hepsine yer verilmemiştir.

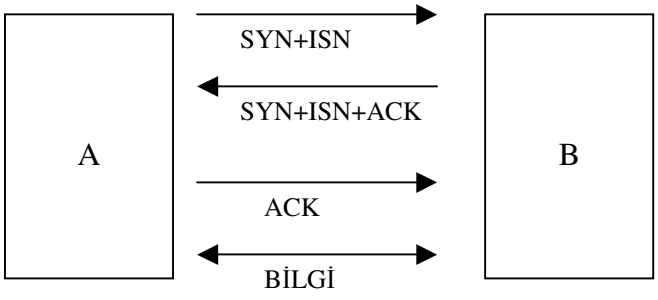
3.1.1 Ağ paketlerini dinleme

Paket dinleyiciler, bir bilgisayar ağı üzerindeki bütün trafiği izleyen araçlardır. Bu araçlar kendilerine gönderilmiş paketleri aldıkları gibi ağ üzerindeki başka konaklara gönderilmiş olan paketleri de kabul ederler. Bu yönleri ile sadece kendilerine gönderilmiş olan paketleri kabul eden standart ağ konaklarından farklı çalışırlar. Paket dinleyicilerinin sağladığı güvenlik tehdidi onların açık metin olarak yazılmış şifreleri, kullanıcı adlarını veya diğer hassas bilgileri de içerebilen bütün ağ trafiğini dinleyebilmelerinden kaynaklanmaktadır. Teorik olarak paket dinleyicileri sezmek çok zordur. Çünkü bu araçlar pasiftirler ve sadece bilgi toplarlar [5].

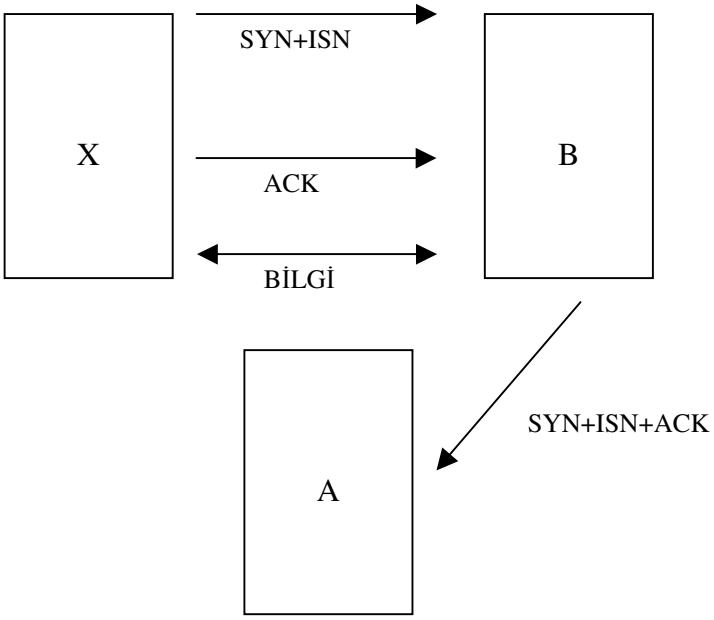
3.1.2 IP taklidi

Bilgisayarlara izinsiz erişim sağlamak için kullanılan bir tekniktir [6,8]. Bu saldırının amacı bir makinenin IP adresini ele geçirmektir. Bilgisayar saldırganının asıl saldırı noktasını saklamasını sağlamakta ya da iki makine arasındaki güvenilir ilişkiden faydalanmaktır [7]. Saldırgan, erişim sağlamak için kaynak adresi değiştirilmiş paketler oluşturur. Bu yöntem IP adreslerine göre yetkilendirme yapan uygulamaları suistimal ederek, saldırganı, hedef bilgisayar üzerinde yetki sahibi yapar [9].

Şekil 2.1 ve Şekil 2.2 de bu duruma örnek verilmiştir. A ve B isimli iki bilgisayar üçlü onaydan sonra veri transferine başlamaktadır. Ağı dinleyen bir saldırgan bu iki bilgisayarın bu yöntemle birbirine bağlandıklarını görebilir. A makinesi herhangi bir nedenle kapatıldığında saldırgan X adındaki bilgisayarını kullanarak A'nın yerine geçer. Önce B bilgisayarına SYN ve kendi ISN'ini yollar. B bilgisayarı bu paketin A dan geldiğini sanar ve SYN ile kendi ISN'ini A bilgisayarına yollar. Bundan sonra X bilgisayarı B'nin yolladığı ISN'i tahmin edip B'den A'ya giden pakete cevap olarak tahmini SYN ve ACK yollar. Aksi bir durum olmazsa X ile B arasında bağlantı kurulur ve X, B bilgisayarı üzerinde A ile B arasında şifre gerektirmeden kullanılabilecek servisleri kullanma hakkını elde eder.



Şekil-3.1 TCP bağlantısında üçlü onay



Şekil 3.2. TCP bağlantısında IP Spoofing

3.1.3. Şifre saldırıları

Her ne kadar paket dinleme ve IP taklidi teknikleri erişim için gerekli olan şifre bilgilerini ele geçirmede kullanılabilse de genellikle kullanıcı hesapları ve şifrelerini tanımlamak için şifre saldırıları tercih edilir. Bu saldırıları gerçekleştirmek için, ardarda birçok şifrenin denenmesi, truva atı programları, IP taklidi ve paket dinleyicileri gibi bir çok farklı yöntem kullanılabilir [3].

Sistemlerde kullanıcı adlarının ve şifrelerinin bulunduğu bir dosya bulunmaktadır. Bilgisayara herhangi bir şekilde girebilen saldırganlar bu dosyalarda bulunan şifreleri şifre kırma programlarının yardımıyla öğrenebilirler. Saldırganlar şifreyi tahmin etmeye de çalışabilirler.

3.1.4. Paket parçalama saldırıları

İnternet Protokolünün bir parçası olan paket parçalama birçok İnternet güvenlik duvarının aşılabilmesini de beraberinde getiren bir mekanizmadır.

Bazı durumlarda İnternet Protokolü'nün mevcut paketi parçalara ayırması gerekebilir. Eğer aktarılmak istenen paket, bulunulan ağ parçası için çok büyük ise, bu yola başvurmak zorunlu olacaktır. Parçalama sonucu oluşan her paket, ayrı bir IP başlığı ve gövdesine sahip olacaktır. Başlık kısmı tüm küçük paket parçaları için aynı olacaktır. Bu yeni küçük paketlerin gövde kısmı ise orijinal paketin belli bir kısmını içerecektir.

Paket parçalamanın paket filtreleme açısından doğurabileceği sorun, her ne kadar her parçada başlık kısmı birbirine çok benzese de, ilk paket hariç diğerlerinde gövde kısmında yer alan bilgilerin hangi protokole ait olduğu bilgisinin yer almayacak olmasıdır. Bu da filtrelemede bu bilginin kullanıldığı durumlarda işlemin yapılamaması sonucunu doğuracaktır.

Bu durumu bir örnekle açıklamak yerinde olacaktır. Saldırılan konağın önünde 23 numaralı TELNET portuna erişim izni vermemekte ancak 80 numaralı http portuna erişim izni vermekte olsun. Saldırgan, başlangıç paketinin hedef port numarasını 80 olarak belirlediği paket parçaları oluşturur. Bu, gönderdiği başlangıç paketinin ve o paketin devamı olan diğer paketlerin güvenlik duvarını aşmasını sağlar. Saldırgan bu paketlerden birisinin birinci ofsetini kurarak paketin başlangıç paketinin üzerine yazılmasını sağlayabilir. Başlangıç paketinin yerine geçecek olan bu paketin hedef portu 25 olarak belirlenmiş ise IP paketi aslında yasaklanmış olan 25 numaralı telnet portuna gitmiş olur ve hedef konak ile telnet bağlantısı kurulmuş olur [10].

3.1.5. "Ortadaki adam" saldırıları

Bu saldırının en temel amacı iki makine arasındaki trafiği değiştirmektir. İletişim esnasındaki veriyi durdurmak, değiştirmek yada yoketmek olabilir [11].

Ortadaki adam saldırısı, saldırganın ağ üzerindeki paketlere erişimini gerektirir. Böyle bir duruma örnek olarak ağınızla diğer ağlar arasında iletilen tüm bilgi paketlerine erişim kazanabilecek, İnternet servis sağlayıcınızda çalışan biri gösterilebilir. Böyle saldırılar genellikle ağ paket dinleyicileri, yönlendirme ve iletişim protokolleri kullanılarak gerçekleştirilir. Böyle saldırıların olası

kullanımları, bilgi hırsızlığı, dahili ağ kaynaklarına erişim için mevcut durumun ele geçirilmesi, mevcut ağ yapısı hakkında bilgi elde etmek için trafiğin izlenmesi, hizmetin engellenmesi, iletilen bilginin bozulması ve ağ oturumlarına yeni bir bilginin eklenmesi olabilir [3].

Bu yöntemde saldırgan, birbiri ile iletişim kurmak isteyen iki konak arasında girerek bu konaklar arasındaki bilgi trafiğinin kendi üzerinden geçmesini sağlar. Bu konuma ulaşan saldırgan paketlerin içeriğini görebilecek potansiyele ulaştığı gibi paketlerin içeriğini değiştirerek önemli zararlar verebilir.

3.1.6. "Hizmet dışı bırakma" saldırıları

Bu saldırı tipinde birincil amaç hedef bilgisayarın aşırı görevlendirilmesini sağlamak ve bilgisayarın üzerindeki görevlerin üstesinden gelememesini sağlayıp hizmet dışı bırakmaktır.

Konu ile ilgili kaynaklara bakıldığında bir çok hizmet dışı bırakma saldırısı örneği olduğu görülmektedir. Bu örneklerden en çok karşılaşılanı saldırılacak bilgisayar ağına aşırı miktarda ICMP ECHO paketi gönderilmesidir. Eğer bu paket yeterli miktarda gönderilebilirse bütün ağ trafiğini kaplayacak ve ağın çalışmasını aksatabilecektir. Bu saldırıyı gerçekleştirmek için bir tek bilgisayar kullanılır ise hedef ağı dolduracak miktarda paket gönderilemeyebilir. Yeterli sayıda paket göndermek için, birden fazla bilgisayarı aynı anda kullanarak yapılan ve dağıtık hizmet dışı bırakma saldırısı adı verilen bir yöntem de ortaya çıkmıştır [12].

3.1.7. TCP sıra numarasının tahmini

TCP sıra numarası saldırısı TCP bağlantısındaki üç aşamalı el-sıkışma sırasına dayalıdır. Bu saldırıda saldırgan TCP paket sırası numaralarını tahmin edebilir. Bu sayede sunucu geçerli bir istemci ile bağlantı kurulduğunu sanar.

TCP güvenilir bağlantı tabanlı bir protokoldür. İki konak arasında gidip gelen paketleri doğru sırada üst katmanlara aktarmaktan sorumludur. TCP bu

sıralama işlemini doğru bir şekilde yapabilmek için paketlerdeki sıra numarası alanını kullanır.

Saldırganın sıra numarası tahmini saldırısı yapabilmesi için öncelikle biri hedef olan iki konak arasındaki ağ trafiğini dinlemesi gerekmektedir. Sonra saldırılacak olan konağa kaynak IP numarası güvenilir bir konağın IP si olan bir paketler gönderilmeye başlanır. Gönderilen paketlerin sıra numaraları hedef konağın beklediği numaralar olmak zorundadır. Bunun yanı sıra paketler hedef konağa, güvenilir konağın gönderdiği paketlerden daha önce ulaşmalıdır. Bunu başarmak için genellikle güvenilir sisteme çok sayıda veri paketleri gönderilerek bir çeşit servis dışı bırakma saldırısı uygulanır [13].

Bu şekilde bağlantı ele geçirildikten sonra hedef bilgisayarın üzerinde IP numarası taklit edilen bilgisayarın hakları elde edilmiş olur.

3.1.8. Kaynak yönlendirme

IP protokolünün Kaynak Yönlendirme olarak adlandırılan ve IP paketlerinin izlemesi gereken yolu tanımlayan bir seçeneği vardır.Bu yol paketlerin izlemesi gereken bir dizi yönlendirici IP adresinden meydana gelmektedir [11].

Kaynak yönlendirme saldırısında, saldırgan yerel ağdan bir ip adresi kullanarak hedef konağa bağlantı kurar ve mesaj paketleri güvenilen bir kaynaktan geliyormuş gibi, yönlendiriciye ve onun arkasındaki yerel ağa erişebilir [3].

Bu saldırının önüne geçmek için basitçe kaynak yönlendirme opsiyonunu kullanan paketlerin kabul edilmemesi yeterli olacaktır [14].

3.1.9. ICMP saldırısı

ICMP, TCP/IP protokol grubunun basit ağ yönetim protokolüdür ve kötüye kullanım için zengin bir potansiyel olarak görülür [15].

ICMP protokolünde saldırı bakımından odak nokta, ICMP yeniden yönlendirme mesajıdır. Bu mesaj yönlendiriciler tarafından, en iyi yol üzerindeki konakları belirtmek için kullanılır. ICMP protokolünde de RIP protokolünde olan saldırılara benzer saldırılarla karşılaşılabilir ICMP saldırılarının zorluğu kurulmuş bir bağlantının varlığıdır. Bu durumda konağın yönlendirme çizelgesine yasal olmayan yönlendirme bilgileri eklemek mümkün değildir. Varolan bir bağlantı için yönlendirme değişikliklerini kabul etmeyerek, bu saldırı tipine savunma sağlanabilir. Fakat bu, bağlantıda başarım düşüklüğüne yol açabilir [3].

3.1.10. Uygulama katmanı saldırıları

Bu tür saldırılar uygulamalar içerisindeki çeşitli zayıflıklardan yararlanırlar. Yine de bir kısmı tiplerine göre sınıflandırılabilir.

Ayarlama sorunları

Uygulamalardaki öncelikli güvenlik sorunlarının başında ayarlama hataları gelmektedir. İki tip hata vardır: ön kabullü (default) kurulum ve hatalı ayarlama.

WEB sunucuları gibi yazılımlar ön kabullü kurulumda saldırganlar için gizli bilgilere erişim sağlar. Örneğin kaynak veriye ulaşmak için dinamik sayfalar üzerinde betikler çalıştırabilirler. Bundan başka bir kurulum ön kabullü kullanıcı adı/şifre ile yönetim ara yüzü sağlayabilir. Böylece saldırgan sitede istediği her şeyi değiştirebilir.

Ana zayıflıklar yanlış tanımlarla ve parametrelerle oluşturulmuş erişim listeleridir. Böylece saldırgan özel sayfalara ve veritabanlarına erişebilir.

Hatalı tanımlamaya klasik örnek Lotus domino web sunucusunda sıklıkla rastlanır. Bu sunucuyu kurarken, Lotus tanım veritabanı hiçbir erişim listesine sahip değildir. Açıkça, eğer names.nsf Lotus veritabanı web tarayıcısı üzerinden kontrol edilmeden erişilebilirse, tüm Lotus kullanıcı isimleri gibi birçok bilginin alınması mümkündür.

Hatalar

Kötü bir program yazılımı her zaman hatalar içerir. Bunlar en önemli zayıflıklardır. Keşfedildikleri zaman dinamik sayfaların kaynak kodlarını ele geçirmek, servisleri kullanılmaz hale getirmek, makinenin kontrolünü almak gibi amaçlarla komutlar çalıştırmaya müsaade eder. Bu hatalardan en bilineni ve en ilginç olanı bellek taşmasıdır.

Bellek Taşması

Bellek taşması kötü programlamanın sebep olduğu bir zayıflıktır. Argüman olarak bir değişkenin boyutuna bakılmadan bellek içerisinde bir fonksiyona kopyalanması sonucu ortaya çıkar. Eğer ki değişken, bellek için hafızada ayrılmış yerden büyükse bellek taşması gerçekleşmesi için yeterlidir. Değişkene parçalı bir program geçirerek patlayacaktır. Eğer ki saldırgan başarılı olursa saldırılan uygulamanın hakları ile hedef makinede uzaktan komut çalıştırabilecektir.

Betikler

Kötü betik programlaması da sıklıkla sistem güvenliğini etkilemektedir. Örneğin Perl betiklerinin içerisinde web yolu dışındaki dosyaları okumaya izin verebilecek ya da izinsiz komutlar çalıştırmaya müsaade edebilecek birçok zayıflık bulunmaktadır [7].

3.2. Zararlı Programlar

İnternet kaynaklı olabilecek zararlı programların başında virüsler, tuzak kapıları, mantık bombaları, bakteriler, Truva atları ve solucanlar gelmektedir.

3.2.1. Virüsler

Virüs, kendini çoğaltabilecek biçimde özel olarak yazılmış koddur. Kendini taşıyıcı programa ekleyerek bir bilgisayardan diğerine yayılmaya çalışır. Donanıma, yazılımlara veya verilere zarar verebilir [16].

Bu programlar (ya da virüs kodları) çalıştırıldığında programlanma şekline göre bilgisayarınıza zarar vermeye başlar. Ayrıca, tüm virüs kodları (bilinen adıyla virüsler) bir sistemde aktif hale geçirildikten sonra çoğalma (bilgisayarınızdaki diğer dosyalara yayılma, ağ üzerinden diğer bilgisayarlara bulaşma vb gibi) özelliğine sahiptir.

Bilgisayar virüslerinin popüler bulaşma yollarından birisi "virüs kapmış bilgisayar programları" dır. Bu durumda, virüs kodu bir bilgisayar programına (sözgelimi, sık kullanılan bir kelime işlemci ya da beğenerek oynanılan bir oyun programı) virüsü yazan (ya da yayan) kişi tarafından eklenir. Böylece, virüslü bu programları çalıştıran kullanıcıların bilgisayarları virüs kapabilirler. Özellikle internet üzerinde dosya arşivlerinin ne kadar sık kullanıldığını düşünülürse tehlikenin boyutları daha da iyi anlaşılabilir.

Virüslenmiş program çalıştırıldığında bilgisayar virüs kodu da, genellikle, bilgisayarınızın hafızasına yerleşir ve potansiyel olarak zararlarına başlar. Bazı virüsler, sabit diskin ya da disketlerin "boot sector" denilen ve bilgisayar her açıldığında ilk bakılan yer olan kısmına yerleşir. Bu durumda, bilgisayar her açıldığında "virüslenmiş" olarak açılır. Benzer şekilde, kendini önemli sistem dosyalarının (MSDOS ve windows için COMMAND.COM gibi) ardına kopyalayan virüsler de vardır [17].

Tipik bir virüsün yaşam seyri, şu dört aşamadan geçer;

Uyku evresi: Virüs işsizdir. Sonunda bir olayla, mesela belli bir tarihte, başka bir dosyanın veya programın varlığıyla ya da disk kapasitesinin bazı limitleri aştığında virüs aktif hale gelir. Bütün virüsler bu aşamalara sahip değildir.

Yayılma evresi: Virüs, özdeş kopyasını disk üzerinde başka bir programa ya da belli bir sisteme yerleştirir. Etkilenen her bir program şimdi kendiliğinden yayılma aşamasına girecek bir virüs süresine sahiptir.

Harekete geçme evresi: Virüs amaçları doğrultusunda fonksiyonlarını harekete geçirir. Harekete geçme evresi, uyku evresindeki bazı olayların aktif hale gelmesinden sonra oluşur.

Çalışma evresi: İşlev gerçekleştirilir. Bu işlev ekranda zararsız bir mesaj olabileceği gibi, veri dosyalarına ve programlara zarar da verebilir.

Çoğu virüs işlerini, özel işletim sistemlerine ya da bazı durumlarda özel bir donanım platformuna özgü yöntemlerle yapar. Bu nedenle virüsler işlerini, özel sistemlerin detay avantajları ve zayıflıkları üzerinde tasarlar

Virüs yapısı

Hastalıklı bir program virüs kodu ile başlar ve şöyle çalışır; Kodun ilk çizgisi ana virüs programına bir zıplamadır. İkinci çizgisi ise özel bir belirleyicidir ki bu işaret bir programa daha önceden virüs bulaşıp bulaşmadığını kontrol amacı ile kullanılır. Program çalıştırılmak üzere çağırıldığında kontrol derhal ana virüs programına geçer. Virüs programı öncelikle virüs bulaşmamış dosyaları seçer ve onlara bulaşır. Sonra virüs zararlı işlevlerini uygulamaya başlayabilir. Bu faaliyet her zaman olabilir ya da belli zamanlarda harekete geçen mantık bombası olabilir. Sonunda virüs, kontrolü orijinal programa aktarır. Bu hastalanma süreci hızlı olursa, kullanıcı hastalıklı ve hastalıksız programın yönetimi arasındaki farkı muhtemel olarak fark etmeyebilir.

Bahsedilen virüsler kolaylıkla ortaya çıkarılabilir. Çünkü bir programın hastalıklı versiyonu normal versiyonuna göre daha uzundur. Virüslerin bu basit yolla ortaya çıkarılmasının bir yolu, yönetilebilir dosyayı sıkıştırarak hastalıklı ve hastalıksız versiyonları eşit uzunluğa getirmektir [18].

Virüs çeşitleri

Virüsler, boot sektör virüsleri, program virüsleri, ve makro virüsleri olmak üzere üç'e ayrılırlar.

Boot sektör virüsleri

Bu virüsler kendilerini taşınabilir disketlere oradan da sabit disklerin boot sektörüne kopyalarlar. Bilgisayar çalıştırıldığında veya yeniden başlatıldığında sabit diskten virüs programı yüklenir. Bu tip virüsler ancak virüs bulaşmış olan disketlerden gelir. Paylaşılan dosya veya çalıştırılabilir programlarla bulaşmaz. Bu virüs türü günümüzde azalmaktadır çünkü günümüz bilgisayarları açılmak için boot diskine ihtiyaç duymamaktadır.

Program virüsleri

Bu virüsler kendilerini programların çalıştırılabilir dosyalarına eklerler. Genellikle bir programın .EXE veya .COM uzantılı dosyalarına bulaşırlar. Bununla birlikte program çalıştırdıkları zaman başka dosya türlerine de bulaşabilirler. Virüs içeren bir program çalıştırıldığı zaman, virüs kendini bilgisayarın belleğine yükler. Virüs bellekteyken çalıştırılan programlara bulaşır [19].

Makro virüsleri

Son yıllarda gözlenen virüs sayısında dramatik bir artış vardır. Bu artışın asıl nedeni yepyeni bir virüsün hızla yayılımıdır ki bu da "Makro Virüs" dür. National Computer Security Agency-NESA ya göre makro virüsleri, bütün virüslerin 2/3'ünü oluşturmaktadır.

Makro virüsleri, özellikle aşağıdaki nedenlerden dolayı tehdit edicidirler:

- Makro virüsü programdan bağımsızdır. Bütün makro virüsleri Microsoft Word yazılımının bütün dökümanlarını etkiler. Herhangi bir donanım platformu ve işletim sistemi tarafından desteklenen Word bile etkilenebilir.
- Makro virüsleri, dökümanları ve çalıştırılmayan kod parçalarını etkiler. Zaten bilgisayara yüklenen bilgilerin çoğu, programdan çok dökümandır.
- Makro virüsleri kolay yayılabilir. En kolay yol elektronik postadır.

3.2.2. Tuzak kapıları

Tuzak kapıları, programlarda bulunan gizli güvenlik açıklarıdır. Kendisini bilen herhangi birinin, bilgisayara standart güvenlik prosedürlerinden geçmeden girebilmesini sağlarlar. Pek çok tuzak kapısı saldırısı önemli hasarlara sebep olabilecek niteliktedir. Tuzak kapıları kullanılarak, programlara, önemli bilgilere veya sunucu bilgisayara yüksek erişim seviyesi sağlanılabilir. Aslında programcılar tarafından programları derlemek ve test etmek için meşru olarak kullanılırlar. Tuzak kapılarının meşru anlamda kullanılmalarının bazı sebepleri şunlardır;

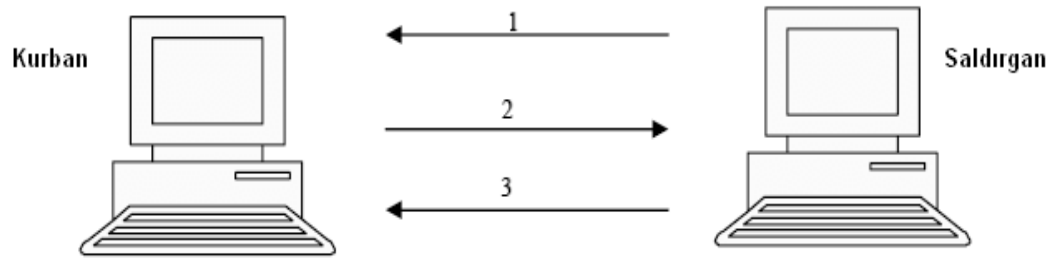
1. Programı test etme işlemini kolaylaştırırlar.
2. Hata durumunda programa ulaşma amaçlı kullanılırlar.
3. Programı hatalardan arındırmak için kullanılırlar.

Fakat ileride legal olmayan yollardan erişim sağlamak için gayrı meşru olarak da kullanılabilirler. Bu anlamda tuzak kapıları ahlaksız programcılar yüzünden yetkisiz erişim elde etmek için kullanıldığında tehdit oluşturmaktadırlar.

Tuzak kapıları arka kapı olarak da adlandırılırlar. Saldırganlar, arka kapıları kullanarak sisteme güvenlik kontrolü prosedürlerini atlayarak ve korsanlık için zaman harcamadan girebilirler [20].

3.2.3. Mantık bombaları

Mantık bombaları programlar içine yerleştirilmiş kodlardır. Önceden tanımlanmış olayların gerçekleşmesi durumda tetiklenirler. Bu kodlar, programa veya işletim sistemine gizlice yerleştirilerek önceden belirlenmiş bazı şartların sağlanmasıyla zarar verici faaliyetlerde bulunurlar.



Şekil 3.3. Ön tanımlı şartlar sağlandığında saldırganı uyaran mantık bombası

Bir mantık bombası, Şekil 3.3 de gösterildiği gibi, kendisini içeren kelime işlemci gibi herhangi bir programı kullanan ve İnternet bağlantısı bulunan bir kullanıcıdan saldırgana saldırıya hazır bulunulduğuna dair mesaj gönderebilir. Böyle bir durum için bombanın saldırıyı yapmadığına ancak saldırgana saldırının başlaması için gerekli durumun oluştuğunu söylediğine dikkat edilmelidir [20].

3.2.4. Bakteriler

Bakteriler, sistem kaynaklarını tüketmek için kendi kopyalarını üreten programlardır. Dosyalara zarar vermezler. Temel amaçları kendilerini çoğaltmaktır. Tipik bir bakteri programı çoklu görev özellikli sistemler üzerinde devamlı olarak kendinden iki kopya yapmaktan fazla bir şey yapmazlar. Bu iki kopya da kendilerinden ikişer tane yapar ve bu böyle devam eder. Bakteri genellikle işlemcinin, belleğin ve diskin kapasitesini doldurana kadar üstel olarak çoğalır [21].

3.2.5. Truva atları ve casus yazılımlar

Bilgisayar terminolojisinde "Truva atı" zarar verici program içeren herhangi bir yazılımdır. Truva atlarının gizlenmek için en sevdikleri yazılım türü, bilgisayar oyunlarıdır. Sıklıkla el değiştiren oyunlar, Truva atlarının etkisinin de hızla yayılmasına neden olurlar. Truva atlarını gizleyen yazılımlar her zaman oyunlar olmayabilir; bazen gerçekten işleyen gösterişli programların, örneğin bir grafik çizim programının içine gizlenmiş Truva atları bulunabilir. Diğer yandan çok yetkin olmayan "bilgisayar korsanları" tarafından geliştirilmiş kaba saba yazılımlar içine yerleşmiş Truva atları ile de sıklıkla karşılaşılmaktadır [22].

Truva atları, zararsız programlar gibi görünürler. Bulaştığı program düzgün çalışmaya devam eder. Yapacağı işleri arka planda çalıştırdığı için kullanıcı hissetmez. Sistemde fark edilmeleri oldukça zordur [23].

Truva atı taşıyan uygulama çalıştırıldığında virüslerde olduğu gibi önce truva atının bulunduğu bölüm çalışır. Truva atının özelliklerine göre bazı işlemler yapılır ve sonra asıl programın çalıştırılması sağlanarak kontrol devredilir. Truva atları aktive olduklarında öncelikle asıl görevlerini gerçekleştirirler. Bunlar genellikle güvenlikle ilgili görevlerdir; sabit diskte kredi kartı bilgisi arama, arka kapı açma vs. olabilir.

İkincil görevler ise belli şartların oluşması durumunda yapılacaklardır. Şartlar; tarih ve zaman, asıl görevin tamamlanması, kullanıcı tarafından gerçekleştirilen bazı şartlar vs... Şartların gerçekleşmesi durumunda; dosyaların silinmesi, formatlama, ekrana çıkan bazı mesajlar, müzik çalınması gibi etkiler ortaya çıkabilir. Bunların tamamı truva atının özelliklerine bağlıdır.

Yeni Truva atlarının bir çoğu bilgisayar korsanlarının kullandığı araçlar olarak geliştirilmiştir. Truva atının olduğu bilgisayarlarda açılan güvenlik açıkları sayesinde bilgisayar korsanları bu bilgisayara zarar verici bir çok faaliyette bulunabilmektedirler. Günümüzde çok popüler olan "servis dışı bırakma" saldırılarında da bu bilgisayarlar kullanıcıların haberi olmadan kullanılabilmektedirler.

Truva atları mutlaka EXE türü program olmak zorunda değildirler. VBS veya JS olan, HTML koda gömülü olan bir çok truva atı mevcuttur. Bu nedenle, nereden gelecekleri belli değildir.

3.2.6. Solucanlar

Son yılların en popüler zararlı programlarından biri olan solucanlar, virüslere benzer özelliklere sahip oldukları için virüslerle karıştırılmaktadırlar. Birçok yerde de virüs olarak anılmaktadırlar. Virüslerle olan temel farkları yayılmak için bir dosyaya bulaşmak zorunda olmamalarıdır. Solucanlar bir

anlamda bulaşmadan çoğalır ve yayılırlar. Yayılmak için ağları kullanırlar. Bu ağ kurumsal bir ağ olabileceği gibi İnternet de olabilmektedir [24].

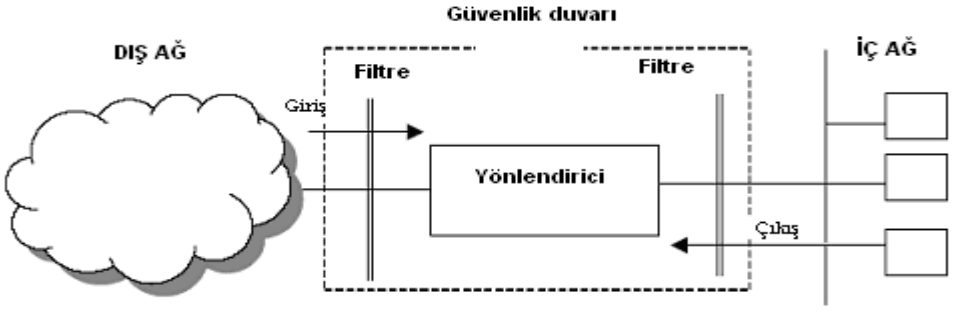
4. İNTERNET'TE GÜVENLİĞİ SAĞLAMA

Bu bölümde İnternet bağlantısı ile gelebilecek olan ve 2. bölümde örnekleri verilen tehditlere karşı alınan önlemler anlatılmıştır. Öncelikle İnternet bağlantısını kısıtlayarak daha güvenli kılmayı amaçlayan Güvenlik Duvarları üzerinde durulmuş, daha sonra da iletilen verilerin güvenliğini sağlamada kullanılan Şifreleme teknikleri incelenmiştir. Ayrıca kaynağı çoğunlukla İnternet olan zararlı programların temizlenmesi amacını taşıyan yaklaşımlara da yer verilmiştir. Son olarak da IPSec teknolojisi kısaca anlatılmıştır.

Bu tezin konusu olan ve İnternet bağlantısını güvenli hale getirmesi amaçlanan Saldırı sezme sistemleri, daha ayrıntılı bir şekilde incelenmek üzere Saldırı Sezme Sistemleri başlığını taşıyan 4. bölüme bırakılmıştır.

4.1. Güvenlik Duvarı

Güvenlik duvarlarının görevi İnternet üzerinden gelebilecek olası saldırılara ve tehdit unsurlarına karşı aktif bir güvenlik sistemi oluşturmaktır. Bu görevi, sadece izin verilen servislerin veya ağ sistemlerinin, sunulacak veya kullanılacak kaynak veya sistemlere ulaşıp ulaşamayacaklarını kontrol ederek yaparlar. Gerektiğinde iç ağ ortamları için kullanılan özel IP adresleme sistemlerini İnternet üzerinde var olan genel IP adresleme sistemlerine çevirerek iç ağlarda kullanılan IP adreslerini gizler ve güvenlik sağlarlar. Bu şekilde servis, protokol, ip adresi veya kullanıcı bazında kısıtlamalar ve filtreleme işlemlerini dışarıdan gelen veya içeriden dışarıya çıkan istekler için iki taraflı olarak uygulamak mümkün olur [25].



Şekil 4.1. Güvenlik Duvarı

4.1.1 . Güvenlik duvarı çeşitleri

Güvenlik duvarları, ağ protokolündeki çeşitli katmanlarda filtreleme yapabilirler. Üç ana kategoriye ayrılırlar: paket filtreleyici güvenlik duvarları, devre düzeyinde güvenlik duvarları ve uygulama düzeyinde güvenlik duvarları. Bunların hepsi kontrol ettikleri protokol katmanına göre karakterize edilmişlerdir [26].

4.1.1.1. Paket filtreleyici güvenlik duvarları

Paket filtreleme en basit paket izleme metodudur. Paket filtreleyici güvenlik duvarı tam olarak isminin çağrıştırdığı işi yapar – paketleri filtreler. En yaygın kullanımı yönlendirici veya iki evli ağ geçitlerindedir. Paket filtreleme işlemi şu şekilde yapılmaktadır: her bir paket ateş duvarından geçtiği esnada paket başlığı bilgisi önceden tanımlı kurallar veya filtreler doğrultusunda incelenir. Kabul etme veya reddetme kararı bu karşılaştırmanın sonuçları doğrultusunda verilir. Her bir paket diğer paketlerden bağımsız bir şekilde incelenir. Paket filtreleyen ateş duvarı genelde filtreleme ağ katmanında veya nakil katmanında yapıldığı için ağ katmanı ateş duvarı olarak da adlandırılır.

4.1.1.2. Devre düzeyinde güvenlik duvarları

Bu tip güvenlik duvarları, dışarıdan bilgi akışına sadece içerideki bilgisayarlardan istek geldiğinde izin verir. Dışarıya giden isteklerin kaydı tutulur ve sadece isteğe karşılık gelen cevaba izin verilir. Bu tip bir güvenlik duvarının en önemli avantajlarından biri, dışarıdakilerin bütün bir ağı sadece güvenlik duvarının adresi olarak görmesidir. Böylelikle, ağın gerisi korunmuş olur. Örneğin, güvenlik duvarının arkasında bulunan bir bilgisayar ağında bulunan bir A bilgisayarı, dışarıdaki bir B bilgisayarına ait web sayfasını görüntülemek istesin. A bilgisayarı, B bilgisayarına ait web sayfası için istek gönderdiğinde, bu istek güvenlik duvarı tarafından yakalanır ve kaydedilir. B bilgisayarı isteği güvenlik duvarından alır ve web sayfasını geri göndermeye başlar. Paketler güvenlik duvarına ulaştığında, gelen paketler A'nın isteğiyle karşılaştırılır. Sonuçta, pakete izin verilir veya paket çöpe atılır.

Bunun en büyük avantajı, içeriden istek gelmediği takdirde dışarıdan içeriye girilmesine izin verilmemesidir. Güvenlik duvarı açana kadar bütün portlar kapalıdır. Ana dezavantajı ise, diğer filtreleme yöntemleriyle birleştirilmediği takdirde içeriden gelen herhangi bir veri isteğine izin vermesidir [27].

4.1.1.3. Uygulama düzeyi güvenlik duvarları

"Vekil Sunucu" olarak da anılan bu güvenlik duvarları, devre düzeyindekilere benzer şekilde, ağa giriş ve çıkış için tek geçiş olarak davranırlar. En önemli fark ise bilgiyi ele alış şekillerindedir.

Devre düzeyindeki güvenlik duvarları adres ve port bilgisine bakarken, uygulama düzeyi güvenlik duvarları paketleri daha detaylı olarak inceler ve içeriğe bakar. Bu yöntemi kullanan güvenlik duvarları, yaygın veri türlerinin güvenlik duvarından geçmesine izin vermeden önce vekil sunucu uygulamaları çalıştırırlar.

Bunun iki önemli avantajı vardır. İlki, dış kaynak ile güvenlik duvarı arkasındaki bilgisayarlar arasında doğrudan bağlantıya izin vermemesi, diğeri ise verinin içeriğine bakılarak filtreleme yapılabilmesidir.

Örneğin, bu güvenlik duvarları kullanılarak, sadece hangi kullanıcıların web sayfasına erişebileceği değil, aynı zamanda hangi web sayfalarına erişebilecekleri de belirlenebilir. Uygulama düzeyindeki güvenlik duvarları sundukları kontrol düzeyleri nedeniyle çok güvenlidirler fakat önemli konfigürasyon gereksinimleri vardır. Ayrıca, paketleri geçirmekte de, çalışan vekil sunucu uygulamaları nedeniyle, diğer güvenlik duvarlarına göre yavaştırlar. İstemci bilgisayarlara da dışarıdaki kaynaklara erişim için ayrıca vekil sunucu konfigürasyonları yapılması gerekir [28].

Bir istemci, uygulama düzeyindeki bir güvenlik duvarına Telnet, FTP, HTTP gibi TCP/IP uygulama protokollerini kullanarak iletişim kurmak istediğinde, güvenlik duvarı geçerli kullanıcı asıllama ve yetkilendirme bilgisini ister. Basit olarak bu bilgi bir kullanıcı adı ve bir şifreden oluşur. Fakat, vekil sunucu Internet'ten erişilebilecek bir konumdaysa, tek kullanımlık şifre gibi güçlü

asıllama mekanizmaları kullanılmalıdır. Eğer vekil sunucu kullanıcı bilgilerini kabul ederse, daha önceden ayarlanmış bir şekilde SMTP veya LDAP sunucusuna bağlanabilir veya kullanıcıya, bağlanmak istediği uzaktaki sistemin adı sorulur. Daha sonra, vekil sunucu bu sisteme yeni bir TCP bağlantısı kurar. Bu ikinci TCP bağlantısı da kurulduktan sonra, vekil sunucu iki bağlantı arasındaki veri akışını kontrol eder. Bu kontrollerle veriler için çeşitli sınırlamalar getirmek mümkündür. Örneğin, FTP trafiği sadece yetkili kişilere sınırlı kapasitelerle açılabilir.

Kimlik tanımlama için gerekli bilgiler yerel olarak vekil sunucuda tutulabileceği gibi başka bir güvenlik sunucusunda da tutulabilir. İkinci yaklaşım, değişik güvenlik yönetim birimlerini ve ağa erişim sunucularını (NAS-Network Access Server) bir noktada toplamak açısından tercih edilebilir. Bu tip bir yaklaşımda güvenlik sunucusuna bağlanmak için bazı protokoller kullanılmaktadır. Örneğin, Livingston Enterprises firmasına ait olan RADIUS ile Cisco firmasına ait olan TACACS.

Bir sunucuya bağlanmadan önce uygulama düzeyinde bir ağ geçidine bağlanmak işlemi saydamlaştırılabilir. Buna "saydam vekillik" denir. Bu iki anlamda kullanılmaktadır:

- Daha geleneksel anlamıyla, saydam vekillikte bir kullanıcı doğrudan kaynağa bağlanmakla bir vekil sunucu üzerinden bağlanmak arasında bir zorluk yaşamaz. Bunun yerine istemci yazılımları bunu sağlayacak şekilde modifiye edilmelidirler. SOCKS bu yaklaşımı kullanmaktadır.

- Diğer anlamında ise, istemci yazılımı vekil sunuculardan haberdar olmak zorunda değildir. Erişim yönlendiricileri (Access Routers) herhangi bir isteği vekil sunucuya yönlendirmek için ayarlanırlar. Bu kullanım günümüzde gitgide yaygınlaşmaktadır.

Bu güvenlik duvarları da devre düzeyindeki güvenlik duvarları gibi internet paylaşımı ile bütünleşmiştir. Bu tip güvenlik duvarları genellikle büyük bilgisayar ağlarını korumak için iş amaçlı kullanılmaktadırlar [29].

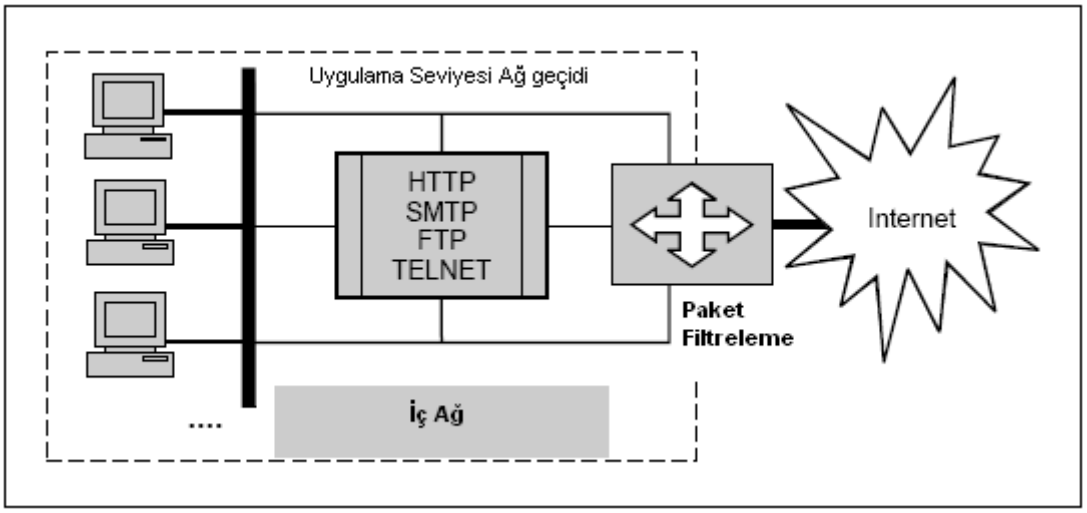
4.1.2. Güvenlik duvarı konfigürasyonları

Pratikte bir güvenlik duvarı genellikle paket filtreleyici ve uygulama düzeyi güvenlik duvarının kombinasyonudur. Buna dayalı olarak, olası üç güvenlik duvarı konfigürasyonu vardır [30]. Bu güvenlik duvarı konfigürasyonları şunlardır;

- Tek evli korumalı konak mimarisi
- Çift evli korumalı konak mimarisi
- Perdelenmiş alt ağ mimarisi

4.1.2.1. Tek evli korumalı konak mimarisi

Tek evli korumalı konak mimarisi'nde servisler, yerel ağa sadece bir arayüzü bulunan bir konak üzerinden verilir. Bu konak, bir yönlendiriciye bağlıdır ve bu yönlendirici üzerinden dış dünyaya açılır. Bu mimaride güvenlik için kullanılacak öncelikli mekanizma paket filtrelemedir. Korumalı konak, yerel ağda yer alır. Denetleme yönlendiricisi üzerinde yer alan paket filtreleme kuralları, İnternet üzerinde yer alan bilgisayarların yerel ağda sadece korumalı konağa bağlantı yapmalarına müsaade edecek şekilde gerçeklenirler. Ayrıca, sadece izin verilen servislere ulaşmasına müsaade ederler. Dışarıdan gelecek tüm servis istekleri korumalı konağa yönlendirileceğinden, bu konağın üst düzeyde güvenliğinin sağlanması gerekir. Aynı zamanda korumalı konak, dış ağda yer alan sunuculara bağlanarak istekte bulunma işlemini de yerine getirir.



Şekil 4.2. Tek evli korumalı konak mimarisi

Denetleme yönlendiricisi üzerinde yer alan paket filtreleme kurulumu, iki farklı şekilde davranabilir;

- Yerel ağda yer alan konakların, belirlenmiş servisler için dış ağa bağlanabilmelerine müsaade eder. Bu işlemi yaparken de paket filtrelemenin getirdiği güvenliği kullanır.
- Yerel konakların dış ağa yapacakları tüm bağlantıları yasaklayarak, bu tür isteklerin tamamının korumalı konak üzerinde yer alan vekil sunucular üzerinden yapılmasını zorlar.

Bu iki yöntem bazı durumlarda birlikte kullanılabilir. Bazı servisler için, paket filtreleme mekanizması kullanılmak suretiyle doğrudan dışarıya ulaşıma izin verilirken bazılarının sadece vekil sunucu üzerinden çalışmasına müsaade edilebilir.

Bu mimaride çift evli güvenlik duvarı mimarisi'nden farklı olarak dış ağa ait olan paketler, yerel ağa geçebilmektedirler. Dolayısıyla bu mimari daha riskli görünebilir. Ama, çift evli güvenlik duvarı mimarisi'nde de meydana gelecek bir sorun neticesinde dışarıdan gelecek tüm paketlerin içeriye geçmesi mümkün olabilecektir. Ayrıca bir yönlendiriciyi saldırılara karşı korumak, bir konağı korumaktan daha kolay olacaktır çünkü bir yönlendiricinin yapacağı işlemler

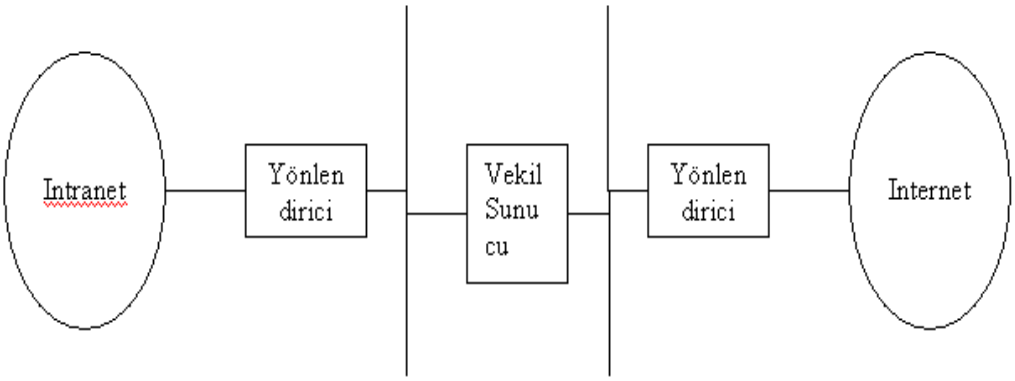
sınırlıdır ve bunlarla ilgili olarak güvenliği sağlamak da, bir konağa ilişkin güvenliği sağlamaya göre daha kolay olacaktır.

Bütün bunlara karşın bu mimarinin de bazı dezavantajları vardır. En başta, bir saldırganın korumalı konağa ulaşması durumunda, ağın geri kalan konaklarına ulaşılması için önünde her hangi bir engel kalmamaktadır. Aynı şekilde, Denetleme Yönlendiricisi'nde meydana gelecek her hangi bir sorun durumunda saldırganın tüm ağa ulaşması için önünde herhangi bir engel kalmayacaktır. Bunların dikkatle göz önünde bulundurulması gerekir.

4.1.2.2. Çift evli korumalı konak mimarisi

İki ağ arayüzlü ve IP yönlendirme özelliği etkisizleştirilmiş, uygulama düzeyinde çalışan bir ağ geçidinden ve filtreleme yapabilen (perdeleyici) yönlendiriciden oluşur. Bu yönlendirici sayesinde, içerisinde değişik sunucuların bulunduğu, perdelenmiş bir alt ağ oluşturulur.

Bu yapıda vekil sunucu tarafından izin verilmeyen hiçbir hizmet kabul edilmez. Dışarıdan gelen hiçbir paketin vekil sunucuyu geçmeden korunan ağa girmesi mümkün değildir. Güvenlik duvarı (ağ geçidi) DNS bilgilerini saklar ve dışarıdan hiç kimse korunan ağdaki ip adreslerini ve isimleri bilemez.



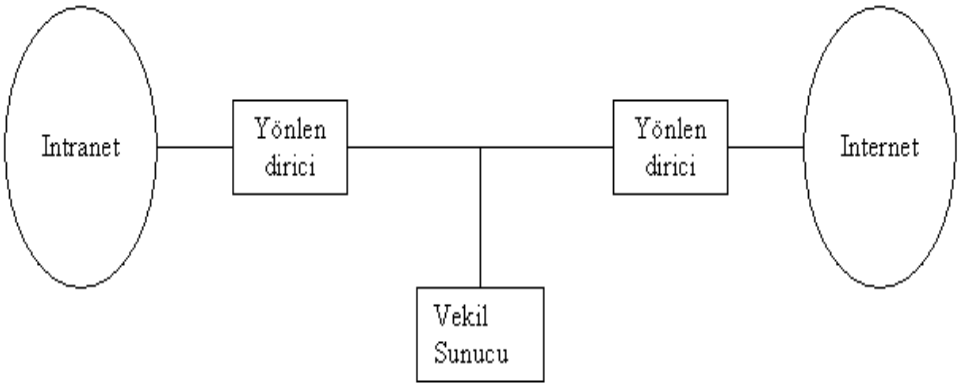
Şekil 4.3. Çift evli korumalı konak mimarisi

Bu yapının basit bir örneğinde vekil sunucu TELNET, FTP ve merkezi e-posta hizmetlerine izin verebilir. Güvenlik duvarında gerekli asıllama ve yetkilendirme bilgileri tutulabilir. Ayrıca, erişim kayıtları da tutularak saldırı aktiviteleri izlenebilir.

Bu yapıda, uygulama ağ geçidi ile yönlendirici arasına başka hizmetler veren sunucular yerleştirilebilir. Bu sunucular, dışarıya IP adresi ve ismi verilmeden, uygulama ağ geçidi üzerinden dışarı bağlanabilecekleri gibi (eğer vekil sunucu bu hizmet desteğini veriyorsa), doğrudan dışarıdan gelen istekleri de alabilirler. İkinci durumda diğer hizmetleri veren sunuculara yapılacak saldırılar uygulama ağ geçidinde takılacaklarından korunan ağa bir saldırı olamaz. Ancak, korunan ağdaki istemcilerin diğer sunuculardan hizmet alabilmek için uygulama ağ geçidinden geçmek zorunda olmaları ağ geçidi üzerinde yoğun bir trafik oluşturur ve bu da performans düşüklüğüne neden olur. Ayrıca, bazı hizmetler veren sunucular (LOTUS Notes, SQLnet gibi) için proxy desteği bulunmadığından, korunan ağdan bunlara erişim yapılamaz ki bu da bu tip yapıların dezavantajlarındandır.

4.1.2.3. Perdelenmiş alt ağ mimarisi

Bu yapıda perdelenmiş bir alt ağ oluşturmak için iki perdeleyici yönlendirici kullanılır. Bunların arasında kalan alana perdelenmiş alt ağ veya silahsızlandırılmış bölge (DMZ – Demilitarized Zone) denir. Bu bölgede birkaç tane uygulama ağ geçidi ve istenirse diğer hizmetleri veren bazı sunucular bulunur. Korunan ağdan dışarı çıkmak isteyenler, yönlendiricilerden ilki tarafından uygulama ağ geçidine yönlendirilirler. Eğer dışarı çıkmak değil de DMZ'de bulunan diğer sunuculardan hizmet almak istiyorlarsa, yönlendirici tarafından uygulama ağ geçidine uğramadan bu sunuculara yönlendirilirler. Bu, çift-evli güvenlik duvarı konfigürasyonuna göre daha esnek bir yapı sağlar. Ayrıca, uygulama ağ geçidi üzerinden yükü azaltarak performans artışını sağlar. Ancak, DMZ'de bulunan diğer sunucuların çok iyi korunması gerekir.



Şekil 4.4. Perdelenmiş alt ağ mimarisi

4.1.3. Güvenlik duvarının olumlu etkileri

Doğru şekilde uygulandığında güvenlik duvarları ağa gelen ve ağdan giden trafiği kontrol edebilir. Yetkisi bulunmayan veya dış ağdaki kullanıcıların iç ağa ve servislere erişimlerini engelleyebilir. Aynı zamanda iç kullanıcıların da dış veya yetkileri bulunmayan ağa veya servislere erişimlerini engelleyebilir. Departmanlar veya diğer özel ağlar servislerin erişim kontrollerini sağlamak amacı ile birçok güvenlik duvarı yapılandırılabilir.

Güvenlik duvarları kullanıcılardan kimlik bilgilerini talep edecek şekilde yapılandırılabilir. Bu ağ yöneticilerinin belirli kullanıcıların belirli servislere ve kaynaklara erişimini kontrol etmesine olanak sağlar. Kimlik doğrulama ayrıca ağ yöneticilerinin kullanıcı aktivitesini ve izinsiz giriş denemelerini izlemesine olanak sağlar.

Güvenlik duvarları denetleme ve kayıt tutma olanakları sağlayabilir. Güvenlik duvarlarını bu şekilde yapılandırarak gerekli bilgiler ileriki günlerde incelenip analiz edilebilir. Güvenlik duvarları ayrıca topladıkları bilgilerden çeşitli istatistiklerde oluşturabilir. Bu istatistikler ağ erişimi ve kullanımı ile ilgili güvenlik kararlarını vermekte oldukça faydalı olabilir.

Bazı güvenlik duvarları güvenilir iç ağları güvenilmez olan dış ağlardan ayırmada kullanılır. Ek katman güvenliği servisleri istenmeyen taramalardan koruyabilir.

4.1.4. Güvenlik duvarının olumsuz etkileri

Güvenlik duvarı çözümlerinin birçok faydası olmasının yanında negatif etkileri de bulunmaktadır.

Güvenlik duvarları bazı ağlarda trafik darboğazına sebep olabilir. Bütün ağ trafiğinin ateş duvarı üzerinden geçmesi zorunlu kılındığı durumlarda ağ trafiğinde tıkanıklık yaşanma ihtimali oldukça fazladır.

Ağlar arası geçişin sadece güvenlik duvarı üzerinden yapıldığı durumlarda eğer güvenlik duvarı doğru yapılandırılmazsa ağlar arasındaki trafik akışında problemler yaşanır.

Ağ servislerine veya kaynaklarına erişim hakkı kısıtlanan kullanıcılarda veya erişim hakkı olup da gerekli şifrelerini hatırlayamayan kullanıcılarda güvenlik duvarları memnuniyetsizliğe yol açabilir.

Güvenlik duvarları fazla olan ağlarda yönetim sorumluluğu arttığı gibi herhangi bir problem olması durumunda bu problemin kaynağını bulmakta zorlaşabilir. Eğer ağ yöneticileri uyarıları ve kayıtları incelemek için yeteri kadar zaman ayırmazlarsa güvenlik duvarının gerçekte işini yapıp yapmadığı hakkında kesin bir bilgiye sahip olamazlar. Bütün güvenlik duvarları devamlı yönetimsel desteğe, genel bakıma, yazılım güncellemelerine, güvenlik yamalarına ihtiyaç duymaları yöneticiler üzerine ek bir yük getirmektedir [31].

4.2. Veri Şifreleme

Şifreleme/deşifreleme (encryption-decryption) bir bilgisayar ağında veya kişisel bilgisayarlarda haberleşme ya da dosya güvenliğini sağlamak için kullanılır. Bu nedenle günümüzde bilgisayarlarda ya da bilgisayar ağlarında şifrelemenin önemi gün geçtikçe artmaktadır [32].

Şifreleme, bilginin alıcı haricindeki kimse tarafından anlaşılmayacak bir şekilde çevrilmesidir. Deşifreleme ise, özel bir anahtar yardımıyla anlamsız bilgiye, şifrelenmeden önceki anlamlı halinin geri verilmesidir. Şifreleme ve deşifreleme kriptografi algoritması olarak adlandırılan matematiksel işlevlerce

gerçekleştirilir. Şifreleme yönteminin kuvveti algoritmanın bilinmezliği ile değil, kullanılan anahtarın uzunluğu ile ilgilidir. Şifrelenen veri anahtar kullanımı ile rahatça açılmakta iken, anahtarın bilinmemesi durumunda verinin elde edilmesi matematiksel işlemlerin yoğunluğu açısından imkansızdır [3].

Simetrik anahtarlı ve asimetrik anahtarlı şifreleme olmak üzere iki tür şifreleme algoritması vardır [33].

4.2.1. Simetrik anahtarlı şifreleme

Gizli anahtarlı şifreleme ya da tek anahtarlı şifreleme olarak da adlandırılır. Tek bir anahtarın hem şifreleme hem de şifre çözme amacıyla kullanıldığı daha geleneksel bir yöntemdir [34].

Simetrik anahtarlı şifreleme, verinin şifrelenmesinde ve deşifrelenmesinde herhangi bir gecikmeye neden olmaz. Simetrik anahtarlı şifrelemede bir anahtar ile şifrelenen veri diğer bir anahtarla açılmadığından, anahtarın gizli tutulması durumunda bir derece kimlik doğrulama sağlanır.

Simetrik anahtarlı şifrelemede, simetrik anahtarın gizli tutulması vazgeçilmez şarttır. Simetrik anahtarın gizli tutulmadığı durumlarda, hem verinin güvenilirliği, hem de kimlik doğrulama ölçütleri tehlikeye girer. Başkasına ait bir simetrik anahtar saldırganın hem kişiye ait gizli bilgilere erişmesini, hem de anahtar sahibinin kimliği ile başkalarına veri göndermesini sağlar [3].

Simetrik anahtarlı şifreleme algoritması olan bazı örnekler şunlardır;

- Veri Şifreleme Standardı (Data Encryption Standart – DES)
- Üçlü DES
- Uluslararası Veri Şifreleme Algoritması (International Data Encryption Algorithm - IDEA)

Veri Şifreleme Standardı (DES) algoritması

Milli Standartlar Bürosu tarafından yayınlanmış DES algoritmasının aşamaları aşağıda sıralanmıştır [3].

1. Mesaj 64 bit uzunluğunda parçalara bölünür.
2. Bu bölmeler bir başlangıç permütasyonundan geçirilir.
3. 56 Bitlik anahtar kullanılarak 16 adet 48 bitlik anahtar elde edilir.
    a. 56 bitlik anahtar üzerinde permütasyon geçirilerek, 2 adet 28 bitlik anahtar elde edilir.
    b. Her iki bölme, 1, 2, 9 ve 16. aşamalarda sola doğru bir, diğer aşamalarda 2 bit döndürülür.
    c. Her bir bölme ayrı ayrı permütasyondan geçirilip, birinci anahtarın 9, 18, 22 ve 25'inci, ikinci anahtarın 35,38, 43 ve 54'üncü bitleri elenir ve 48 bitlik anahtar elde edilir.
4. DES aşamaları gerçekleştirilmek üzere, 16 adet 48 bitlik anahtarın her biri kullanılır.
    a. 64 bitlik giriş değeri 32 bitlik iki parçaya bölünür.
    b. DES aşamasına giriş değerinin sağ tarafı, aşama çıkış değerinin sol tarafı olur.
    c. Aşamaya giriş değerinin sağ tarafına parçalayıcı işlevi, aşamaya ait anahtarla uygulanır.
    d. Parçalayıcı işlevin sonucu, aşamaya giren değerin sol tarafı ile aşamadan çıkan değerin sağ tarafının XOR işleminden geçirilmesidir.
5. Dördüncü aşama 16 kez tekrar edilir.
6. Sonucun sol ve sağ parçaları yer değiştirir.
7. Son permütason gerçekleştirilir.

Üçlü DES algoritması

Üçlü DES basitçe DES algoritmasının başka bir modudur. Toplam anahtar uzunluğu 192 bit olan üç tane 64 bitlik anahtar kullanır [35].

Bu algoritmaların aşamaları şu şekildedir:

1. Açık metin, anahtar 1 kullanılarak şifrelenir.
2. Birinci aşamada elde edilen şifrelenmiş metin, anahtar 2 ile deşifrelenir.
3. İkinci aşama sonucunda elde edilen metin, anahtar 3 ile şifrelenir.

Uluslararası Veri Şifreleme Algoritması (IDEA)

İsviçre'de, ETH Zurich'de, Xueja Lai tarafından geliştirilen IDEA, 128 bitlik anahtar kullanarak, 64 bitlik açık metini 64 bitlik şifreli parçalar halinde şifreleyen bir algoritmadır. Feistel yapısının ilkel bir genellemesini temel alan IDEA, bir çıkış çevrimi tarafından takip edilen, hesapsal olarak birbirinin aynı 8 aşama içerir.

Uzun zamandır bilinen IDEA algoritması, uzun zamandır üzerinde incelemeler yapılmasına karşın herhangi bir saldırı teşebbüsüne maruz kalmamıştır. Biham, Shamir, ve Biryukov tarafından IDEA'ya gerçekleştirilen saldırı teşebbüsü, algoritmanın ancak dördüncü seviyesine kadar ulaşmış, algoritmanın 8 aşamalık toplam bölümü yine güvenli olarak kalmıştır [3].

Amerika ve birçok Avrupa ülkesinde patentlenmiş olan IDEA algoritmasının kabaca aşamaları aşağıdadır:

1. Şifrelenecek mesaj 64 bitlik parçalara bölününür. Herbir açık metin parçası, şifreli parça haline dönüşecektir.
2. 64 bitlik parçaları, 16 bitlik 4 bloğa ayrılır.
3. 52 adet 16 bitlik anahtar oluşturmak üzere, 128 bitlik bir anahtar kullanılır.
4. Tek sayılı aşamada dörtlü, çift sayılı aşamalarda ikili kümelerde anahtar kullanılarak 17 aşama gerçekleştirilir.
5. 16 bitlik bloklar, 64 bitlik şifreli bloklar olarak birleştirilir.

RC4

RSA Veri Güvenlik Firması ve Ron Rivest tarafından geliştirilen RC4 algoritması, biri kaynak kodunu haber gruplarına postalayana kadar ticaret sırrı olarak kullanılmıştır. Çalışma hızının çok yüksek olması güvenlik seviyesinin bilinmemesine rağmen bazı hız gerektiren uygulamalarda RC4 algoritmasının kullanımını uygun kılmıştır. İstenilen her büyüklükte anahtar uzunluğunu kabul eden RC4 aslında, bir psuedo rasgele sayı üreticisi ve bu sayı üreticisinin çıktısının, şifrelenecek veri ile XOR işlemine tabi tutulmasından ibarettir. Bu nedenle, aynı anahtar iki farklı veri kümesini şifrelemede kullanılmamalıdır [3].

4.2.2. Açık anahtarlı şifreleme

Açık anahtar tabanlı şifreleme sistemlerinin tarihi 1970'li yıllara dayanır. Diffie ve Hellman'ın [36] temellerini attığı bu sistemde zaman içinde birçok algoritma önerilmiştir.

Açık anahtarlı şifreleme ile ilgili ilk makalelerin ortaya atılmasına kadar olan süreçte kullanılan simetrik şifreleme sistemleri göz önünde bulundurulduğunda açık anahtarlı şifrelemenin gelişmesi, şifreleme tarihindeki en büyük devrimdir.

Açık anahtarlı şifreleme, gerçek anlamda daha önceki gelişmelerden radikal bir kopuştur. Açık anahtarlı şifreleme sistemlerinin en önemli noktaları matematiksel işlevler üzerine temellenmiş olmalarıdır, aslında açık anahtarlı şifreleme için matematiğin çözüm getiremediği bir takım durumları (örneğin çok büyük bir sayının iki asal çarpanının bulunmasının matematikte herhangi bir doğrudan çözümü olmaması gibi) kullanarak güvenlik sağlar. Daha da önemlisi, açık anahtarlı şifreleme algoritmaları, tek anahtar kullanan simetrik geleneksel şifreleme algoritmalarının tersine, iki ayrı anahtarın asimetrik kullanımını öngörür [37].

Açık anahtarlı şifreleme de biri açık anahtar, diğeri özel anahtar olarak adlandırılan bir anahtar çifti kullanılır. Bu anahtar çifti verinin imzalanması, şifrelenmesi ve kimlik doğrulamada kullanılır. Şifreleme yönteminde açık anahtar

herkese dağıtılırken, özel anahtar sadece sahibi tarafından bilinir. Açık anahtar ile şifrelenen veri sadece özel anahtar ile açılabilmektedir [3].

Açık anahtarlı şifreleme, simetrik anahtarlı şifrelemeye göre daha fazla matematiksel işlem gerektirir. Bu nedenle büyük uzunluklu verilerin şifrelenmesinde kullanılması uygun değildir. Bu tarz veri boyutlarında açık anahtarlı şifreleme ve simetrik anahtarlı şifreleme birlikte kullanılmalıdır. Çoğunlukla karşılaşılan yöntem, simetrik anahtarın değişiminde açık anahtarlı şifrelemenin kullanılması, verinin değişiminde ise simetrik anahtarlı şifrelemenin kullanılmasıdır.

Açık anahtarlı şifreleme kimlik doğrulamada da kullanılmaktadır. Kimliği doğrulanacak olan gönderici, özel anahtarı ile veriyi şifreler. Özel anahtar ile şifrelenen veri sadece açık anahtar ile açılabildiği için, göndericinin kimliği doğrulanır.

Açık anahtarlı şifreleme algoritmalarına örnek olarak RSA algoritması gösterilebilir.

RSA algoritması

RSA şifreleme sistemi, hem şifreleme hem de sayısal imza atma olanağı tanıyan açık anahtarlı bir şifreleme yapısıdır. Sistemi 1977 yılında Ronald Rivest, Adi Shamir ve Leonard Adleman geliştirmiştir. Sistem adını, geliştiricilerinin isimlerinin ilk harflerinden alır.

RSA algoritması şöyle çalışır: Öncelikle p ve q olmak üzere iki tane asal sayı üretilir. Bunların birbirleriyle çarpılmasıyla n, n=p.q, elde edilir. Bundan sonra n sayısından küçük ve (p-1).(q-1) sayısıyla 1 dışında herhangi bir ortak böleni bulunmayan bir e sayısı seçilir. Daha sonra (e.d-1) sayısının (p-1).(q-1) çarpımına tam olarak bölünmesini sağlayan bir d sayısı bulunur. e ve d değerleri, sırasıyla, açık ve gizli üs olarak adlandırılırlar. Açık anahtarı (n,e) çifti, gizli anahtarı ise (n,d) çifti oluşturur. p ve q sayıları ya yok edilmeli ya da gizli anahtar ile birlikte saklanmalıdır.

Gizli anahtar olan d sayısının, (n,e) sayılarından elde edilmesi zor bir işlemdir. Eğer bir kişi n sayısını çarpanlarına ayırarak p ve q sayılarını elde edebilirse gizli anahtarı da kolaylıkla bulabilir. Bu sebeple RSA sisteminin güvenliği çarpanlarına ayırma probleminin zorluğu temeline dayanır. Çarpanlarına ayırma işleminin kolay bir yönteminin bulunması, RSA algoritmasının kırılması anlamına gelir.

Örneğin; Ayşe'nin Barış'a m mesajını göndermek istediğini farz edelim. Ayşe, m mesajının üssünü alarak, c=me (mod n), c şifreli mesajını elde eder. Burada kullanılan (n,e) Barış'ın açık anahtarıdır. Bu işlemleri yaptıktan sonra Ayşe şifreli c mesajını Barış'a gönderir. Barış aynı şekilde gelen şifreli mesaj c'nin üssünü alır. m=cd (mod n) Ancak bu sefer kendi gizli anahtarını kullanır. e ve d arasındaki bağlantı Barış'ın mesajı doğru mesajı elde ettiğinin kanıtıdır. Barış'ın gizli anahtarını sadece Barış bildiği için, gelen şifreli mesajı da sadece Barış okuyabilir.

4.2.3. Kimlik doğrulama ve sayısal imzalar

İletişimin üçüncü şahıslar tarafından dinlenememesi, haberleşilen kişinin doğruluğunun denetlenebilmesi ve mesaj bütünlüğünün sağlanabilmesi gibi gereksinimleri karşılayabilmek için kullanılan kimlik doğrulama protokolleri ve sayısal imzalar ayrı başlıklar altında incelenecektir.

4.2.3.1. Sayısal imzalar

Şifreleme ve deşifreleme teknikleri verinin İnternet üzerinde iletiminde karşılaşılan sorunlardan sadece gözetlemeyi engellemektedir. Sayısal imzalar ise, değiştirme ve taklit etme sorunlarına çözüm getirmektedir.

Değişim sezme ve kimlik doğrulama teknikleri tek yönlü çırpma (one-way hash) olarak adlandırılan bir matematik işlevine bağlıdır. Tek yönlü çırpma işlevinin özellikleri şöyledir:

- Veriden elde edilen çırpı değeri tektir. Veride yapılan bir bitlik değişiklik çırpı değerini de değiştirir.
- Çırpı değerinden, çırpılan veri elde edilemez. Bu nedenle çırpma işlevi tek yönlü işlev olarak adlandırılmıştır.
- Sayısal imza, veriye yek yönlü çırpı işlevi uygulanarak elde edilen çırpı değerinin, kullanıcının özel anahtarı ile şifrelenmesi ile elde edilir.
- Veri tek yönlü çırpı işlevinden geçirilerek tek yönlü çırpı değeri elde edilir. Elde edilen çırpı değeri özel anahtar ile şifrelenir ve sayısal imza elde edilir. Elde edilen sayısal imza veri ile birlikte alıcıya gönderilir. Alıcı sayısal imzayı göndericinin açık anahtarı ile açar ve gönderici tarafından hesaplanmış tek yönlü çırpı değerini elde eder. Daha sonra elindeki veriden tek yönlü çırpı değerini hesaplar ve diğer çırpı değeri ile karşılaştırır. İki değerin eşitliği durumunda göndericinin kimliği doğrulanır [3].

4.2.3.2. Kimlik doğrulama

Kimlik doğrulama, bilgisayar ağları üzerinde iki tarafın birbirlerinin kimliklerinden emin olmasıdır. Bilgisayar ağlarında iletişim, istemci ile sunucu arasında gerçekleşir. İstemci kimlik doğrulama, istemcinin kimliğinin sunucu taradından tanımlanması olarak tanımlanırken, sunucu kimlik doğrulama ise, sunucunun kimliğinin istemci tarafından doğrulanmasıdır.

Sertifikalar, istemci ve sunucunun kimlik doğrulama dışında, elektronik posta mesajında kullanıcının kimliğini, “activex” ve “java applet” programlarının güvenilirliğini kanıtlamak için kullanılırlar.

İki tip istemci doğrulaması vardır:

- Şifre temelli kimlik doğrulama.
- Sertifika temelli kimlik doğrulama.

Şifre temelli kimlik doğrulama

Bu kimlik doğrulama yönteminde sunucu isim ve şifrelerden oluşan bir veri tabanı tutar. İstemcinin girmiş olduğu isim ve şifre, veritabanındaki bilgilerle karşılaştırılarak kimlik doğrulama işlemi gerçekleştirilir. Kimlik doğrulama istemcinin girmiş olduğu isim ve şifre çiftinin, sunucunun veritabanındaki çiftle aynı olduğu durumda başarıyla sonuçlanır. Tüm sunucu yazılımları şifre temelli kimlik doğrulamayı destekler.

Sertifika temelli kimlik doğrulama

Sertifika temelli kimlik doğrulamada istemci rasgele üretilmiş veriyi, sayısal imzası ile imzalar ve sertifikası ile birlikte sunucuya gönderir. Sunucu açık anahtarlı şifreleme tekniklerini kullanarak sayısal imzayı doğrular ve sertifikanın geçerliliğini onaylar.

4.3. Antivirüs Yazılımları

Virüslerin tehdidine karşı ideal çözüm önlemedir. Öncelikle virüsün sisteme girmesine izin verilmemelidir. Bu amacın genelde başarılması mümkün değildir, sadece virütik saldırıların sayısı azaltılabilir. Sonraki en iyi yaklaşım şunları yapmaktır;

- **Bulma:** Hastalık ortaya çıkınca, tespit edilir ve virüsün yeri belirlenir.
- **Tanımlama:** Bulma başarılırsa, hastalıklı programdaki virüs tanımlanır.
- **Yok etme:** Belirli virüs tanımlanınca, hastalıklı programdan virüsün tüm formları yok edilir veya programın orijinal durumu tekrar yüklenir. Virüsün bütün formları daha fazla yayılmaması için sistemden atılır.

Virüs ve antivirüs teknolojisindeki gelişmeler elden ele geçmektedir. İlk virüsler nispeten basit kodlu ve basit amaçlı olarak tanımlanabilir ve antivirüs yazılım paketleri ile temizlenebilirlerdi. Virüslerle ilgili uluslar arası yarış geliştikçe hem virüs programları hem de antivirüs yazılımları karmaşık hale geldiler.

4.3.1. Antivirüs yazılımlarının tarihsel gelişimi

Antivirüs yazılımlarının gelişimi 4 jenerasyona ayrılır;

- **Birinci jenerasyon:** Basit tarayıcılar.
- **İkinci jenerasyon:** Sezgisel/keşifsel tarayıcılar.
- **Üçüncü jenerasyon:** Aktif tuzaklar, hileler.
- **Dördüncü jenerasyon:** Tam özellikli koruma.

4.3.1.1. Birinci jenerasyon yazılımlar

İlk jenerasyon tarayıcıları virüsü tanımlamak için virüs imzasına ihtiyaç duyarlar. Belirli imza taşıyan tarayıcılar, bilinen virüslerin bulunmasında sınırlı işleve sahiptirler. Bir diğer birinci jenerasyon yazılımı türü de programların uzunlukları ile ilgili kayıt bulundurarak bu kayıt uzunluklarındaki değişimleri tararlar.

4.3.1.2. İkinci jenerasyon yazılımlar

İkinci jenerasyon yazılımlar belirli bir imzaya güvenmezler. Bu tür tarayıcılar daha çok sezgisel/keşifsel kuralları kullanırlar. Virüs olma ihtimali olan kodları tararlar. Virüslerin şifreleme eğilimine bakarlar ve buna göre şifreleme anahtarını bulurlar. Anahtar bulununca tarayıcı virüsü tanımlar ve bozulmayı önlerler.

Bir diğer yaklaşımları da bütünlük kontrolüdür. Basit bir kontrolden öte karmaşık bir fonksiyon kullanarak, virüsün aynı karışık kodu tekrar üretmesi engellenir.

4.3.1.3. Üçüncü jenerasyon yazılımlar

Üçüncü jenerasyon yazılımlar, hafızaya yerleşik olarak çalışan programlardır. Hastalıklı programlardan çok virüslerin yaptıkları hareketlerle ilgilenirler. Bu programların avantajları sezgisel kurallar ya da virüs imzaları ile uğraşmazlar. Bu tür programlar daha çok küçük hareketlerle ilgilenir ve onların bulaşma teşebbüslerine müdahale ederler.

4.3.1.4. Dördüncü jenerasyon yazılımlar

Bu jenerasyona ait olan antivirüs yazılımları, değişik antivirüs teknikleri içeren paketlerdir. Bu yazılımlar tarama ve aktif tuzak elemanlarını içerirler. Dahası böyle bir paket, virüslerin bir sistem içine girme yeteneklerini kısıtlayan ve bir virüsün dosyaları bozmaya geçmeleri için güncelleştirme yeteneğini kısıtlayan geçiş kontrol kabiliyetini kapsar.

Dördüncü jenerasyon yazılımları ile savunma alanını daha genel amaçlı bilgisayar güvenlik ölçülerine genişleten, geniş kapsamlı savunma stratejisi kullanılır [18].

4.3.2. İleri antivirüs teknikleri

Daha karmaşık antivirüs yaklaşımları ve ürünleri gelişmeye devam etmektedirler. Bu bölümde en önemli iki teknik incelenecektir.

4.3.2.1. Genel çözümleme

Genel çözümleme teknolojisi, hızlı tarama kullanarak, en karmaşık polimorfik virüslerin bile kolayca bulunmasını sağlar. Polimorfik bir virüs içeren bir dosyanın çalıştırılabilmesi için, virüsün kendini çözümlemesi gerekmektedir. Böyle bir yapıyı bulmak için yürürlükteki dosyalar genel çözümleme tarayıcısından geçirilirler. Bu tarayıcı aşağıdaki elemanları içerir.

CPU emülatörü: Emülatör, donanıma ait işlemcilerin ve bütün yazmaçların yazılım versiyonlarını içerir, böylece temel prosedür emülatörde yorumlanan programlardan etkilenmemiş olur.

Virüs imza tarayıcısı: Bilinen virüs imzalarını araştıran, hedef kodu inceleyen model.

Emülatör kontrol modülü: Hedef kodun yürürlüğe girmesini kontrol eder.

Her bir simülasyon başladığında, bir periyotta bir tane olmak üzere hedef kodda bulunan komutlar yorumlanır. Böylece çözümleme rutini bir kod içerirse bu bir virüsü belirtir ve o kod yorumlanır.

Yorumlama sırasında hedef kod, bilgisayara ve çevresine zarar veremez. Çünkü tamamen kontrollü bir çevrede yorumlanır.

Bir genel çözümleme tarayıcısı ile en zor tasarım konusu, her yorum çalışma zamanının ne kadar süreceğini saptamaktır. Tipik olarak, virüs elemanları bir program çalışmaya başladıktan sonra aktif olur. Tarayıcı belli bir programın izinden daha çok gittikçe, daha çok saklı virüs bulabilir. Bununla birlikte, antivirüs programı sınırlı bir zaman aralığında devam edebilir ve kullanıcıların şikayetlerinden önce harekete geçer.

4.3.2.2. Dijital bağışıklık sistemi

Bu sistemin gelişmesinin nedeni, İnternet temelli virüs yayılım tehditlerinin artmasıdır. Bu tehditler hakkında yeterli olacak açıklama yapıldıktan sonra dijital bağışıklık sisteminin geliştirici olan IBM şirketi çalışanlarının yaklaşımı incelenecektir.

Şu anda virüs tehdidi, yeni virüs ve mutasyonların nispeten yavaş yayılımları ile karakterize edilir. Antivirüs yazılımları, tipik olarak periyodik temelde güncelleştirilir ve bu problemin kontrolü için yeterlidir. Fakat İnternet teknolojisindeki iki ana eğilimin, virüs yayılım oranlarında artırıcı etkisi olmaktadır.

Kompleks posta sistemi: Microsoft Outlook gibi sistemler, birisine bir şey obje, almayı ve alınan objelerle çalışmayı basitleştirmektedir.

Taşınabilir program sistemi: Java ve ActiveX gibi yetenekler, programların kendilerini bir sistemden diğerine taşıma olanağı verir.

Bu Internet temelli yeteneklerin görevleri sonucunda bir takım tehditler oluşmaktadır. Bu tehditlere karşı tepki olarak IBM şirketi bir prototip dijital

bağışıklık sistemi geliştirmiştir. Bu sistem genel izleme ile virüs bulma sistemi sağlar. Bu sistemin tarafsızlığı, hızlı bir tepki süresini sağlayarak virüslerin tanıtımından sonra en kısa süre içinde sonuca ulaşır. Bir organizasyona yeni bir virüs girdiğinde, bağışıklık sistemi otomatik olarak onu yakalar, analiz eder. Sonrasında bu virüs bulunur, koruma altına alınır ve antivirüs programlarına bu virüs hakkında bilgi dağıtılır. Böylece başka bir yerde çalışmadan önce bulunabilir.

Dijital bağışıklık sisteminin başarısı, virüs analiz makinesinin yeni virüs zararlarını bulma yeteneğine bağlıdır. Sürekli olarak başıboş virüslerin, analiz ve denetimi ile tehditlerle karşılaşmamak için dijital bağışıklık yazılımlarının sürekli güncelleştirilmesi mümkün olabilmelidir [18].

4.4. İnternet Protokol Güvenliği (IPSec)

IPSec (IP Security), İnternet ortamında özel ağların güvenli bir şekilde haberleşmesini sağlamaktadır. IPSec, IETF (Internet Engineering Task Force) IPSec Çalışma Grubu tarafından geliştirilmektedir. Yakın gelecekte ağ güvenlik standartlarının önemli bir parçası olacaktır. IPv6 modelinde zorunlu olarak kullanılacaktır. Ağ katmanında sıkı kimlik doğrulama ve şifreleme yapmaya olanak tanımaktadır. Çift yönlü tünel kullanarak haberleşmeyi gerçekleştirmektedir. Açık anahtarlı veri şifrelemeyi desteklemektedir. Yalnız IP trafiğini destekler ve IP yığınına gömülü olarak çalışır.

Anahtar değişim protokolü olarak IKE (Internet Key Exchange) veya ISAKMP (Internet Security Association and Key Management Protocol)/Oakley protokolünü kullanmaktadır. Değişim protokolü olarak AH ve ESP değişim protokollerini desteklemektedir. Şifreleme algoritmaları olarak DES (Data Encryption Standart), IDEA (International Data Encryption Algorithm), Blowfish, RC5, 3DES, CAST, RSA gibi algoritmaların yanında özgün algoritmaların kullanımına da olanak tanımaktadır. MD5, SHA (Secure Hash Algoritm) ve Tiger gibi Hash algoritmalarını kimlik doğrulama işlemlerinde kullanmaktadır [2].

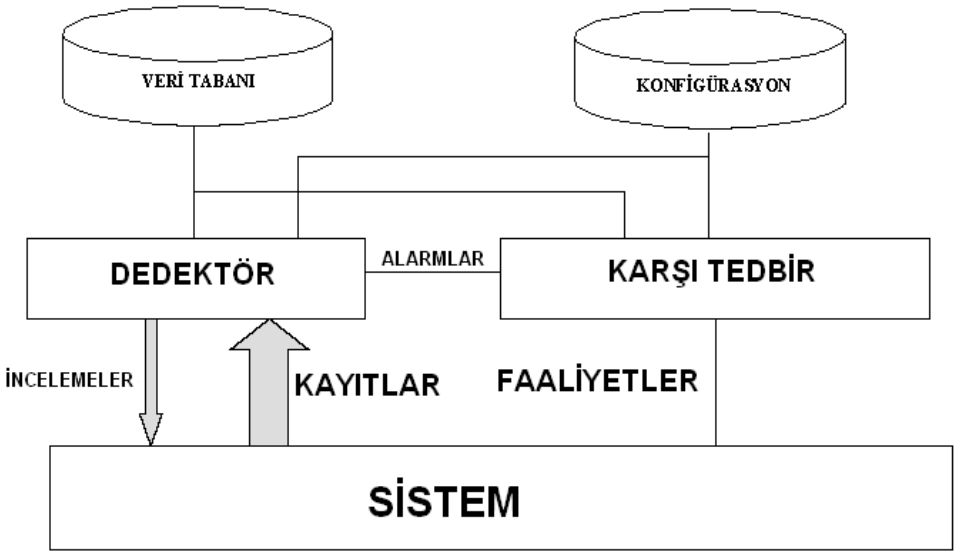
5. SALDIRI SEZME SİSTEMLERİ

Bu bölümde öncelikle Saldırı Sezme Sistemlerinin (SSS) tanımı üzerinde durulacak daha sonra SSS hakkında genel kavramlar verilecektir. Saldırı ve saldırı sezmenin tanımı yapılıp SSS'lerin kullanılma nedenleri belirtilecektir. SSS'lerin sınıflandırılmaları yapıldıktan sonra verimlilik unsurları anlatılacaktır. Son olarak SSS'lere karşı yapılacak olan önemli saldırılar ve SSS'lerin geleceği konularına değinilecektir.

5.1. Tanımı

SSS, bilgisayar sistemlerinde veya ağlarında gerçekleşen olayların güvenlik sorunu olmaları açısından gözlenip analiz edilmesini otomatikleştiren yazılım veya donanımlardır [38].

Bir SSS verilen ağdaki aktiviteleri dinamik olarak gözler ve bu aktivitelerin içinde bir saldırı belirtisi olup olmadığına dair karar verir. Bu anlamda korunacak olan sistemden gelen bilgileri işleyen dedektörler olarak tanımlanabilirler [39].



Şekil 5.1. Genel bir SSS'in mimarisi.

Bilgisayar ağlarına yapılan saldırıların arttığı son birkaç yılda çoğu organizasyonun güvenlik altyapısında gereklilik haline gelmişlerdir [38].

5.2. Genel Kavramlar

Her ne kadar SSS'ler yeni bir teknoloji olarak çocukluk döneminde olsa da olağanüstü hızda gelişim göstermektedir. SSS'ler ile ilgili kavramlar da SSS'lerin gelişimi gibi hızla artmaktadır. Bu bölümde bu kavramların okuyucuya aktarılması amaçlanmıştır.

Alarm

Alarm, SSS tarafından bir saldırının başladığını bildirmek üzere sistem operatörüne gönderilen uyarıdır. Bir saldırının ortaya çıkarılması halinde SSS analisti farklı metotlarla uyarır. Eğer SSS konsolu yerel bir monitör ise alarm normal olarak monitörde belirir. Bu uyarılarda ses de kullanılabilir. Uyarılar uzaktaki konsollara çeşitli yollarla gönderilebilir. Elektronik posta veya kısa mesaj bu yollara örnek olarak gösterilebilir.

Anomali

Çoğu SSS bilinen bir imzayla eşleşen bazı olaylar gerçekleştiği zaman uyarı verir. Bir anomali tabanlı SSS konağın veya ağın aktivitelerinin zamana bağlı olarak profilini çıkarır. Bu profilin dışına çıkan bir olay olduğu zaman SSS alarm verir. Buna örnek olarak bir kullanıcının bir anda yönetici yetkisine veya kök dizin ayrıcalıklarına sahip olması verilebilir.

Saldırı

Saldırı, bilgiye ulaşma, bilgiyi değiştirme veya hedeflenen ağ sisteminin amaçlanan işlevini yapmasını engellemek amacıyla bir sistemin delinmesi veya sistemin güvenliğinin sekteye uğratılması olarak tanımlanabilir.

Otomatik yanıt

Bazı SSS'ler saldırı durumunda uyarı üretebildiği gibi saldırılara karşı savunma da sağlayabilmektedirler. Bu farklı yollarla yapılabilmektedir: birincisi, yönlendiricileri ve güvenlik duvarlarını yeniden konfigüre ederek aynı adresle gelecekte veri alış verişini durdurabilmektedirler, ikincisi, ağa paket sokuşturarak bağlantının yeniden kurulmasını sağlayabilmektedirler. İki yöntemde de sorunlar vardır. Saldırganlar yeniden konfigürasyon aracını adres yanıltma metoduyla kendi avantajlarına kullanabilmektedirler. Bu sayede SSS yönlendiricileri ve güvenlik duvarlarını bu adresleri reddedecek şekilde konfigüre edebilmektedirler. Paket sokuşturma metodu, aktif bir ağ arayüzüne ihtiyaç duyar. Bu sayede de kendisini saldırıya açık hale getirir.

Suistimal

Her sistem açığı için bir suistimal söz konusudur. Suistimal, yapısal sistem zayıflıklarından avantaj sağlamak olarak düşünülebilir. Bir sisteme saldırmak için bir saldırgan, koddaki incinebilirliği sömürebilir.

Yanlış negatif

Yanlış negatif, bir saldırının veya olayın SSS tarafından ortaya çıkarılamaması veya analist tarafından olağan olduğuna karar verilmesi halidir.

Yanlış pozitif

SSS'in saldırı olmayan bir olayı saldırı olarak değerlendirmesi durumudur.

Fragmentasyon

Eğer bir paket çok büyük ise küçük parçalara bölünmek zorundadır. Paketler, maksimum taşınabilir ünite boyu farklı olan ağlardan geçebilmektedir. Örneğin Ethernet ağlarında maksimum taşınabilir ünite boyu 1500 iken Token Ring ağlarda bu 4464'tür. Bu nedenle eğer bir paket Token Ring ağda Ethernet ağına taşınırken daha küçük parçalara bölünmeli ve hedefe varınca tekrar oluşturulmalıdır. Saldırganlar bu olayı SSS'lere yakalanmamak için

kullanabilmektedirler. Bu teknikle yapılan birkaç servis dışı bırakma saldırısı bilinmektedir.

Sezgisellik

Sezgisellik terimi saldırıların yapay zeka kullanılarak ortaya çıkarılması anlamında kullanılmaktadır. Sezgiselliği kullanan SSS'ler hemen hemen on yıldır geliştirilmektedirler fakat henüz zararlı ağ trafiğini tam anlamıyla engelleyecek seviyede değildirler.

Bal kabı

Bal kabı, bir veya birden fazla incinebilir konağı simüle edebilen bir sistemdir ve saldırgan için kolay bir hedef teşkil eder. Bal kabının başka bir görevi yoktur. Bu nedenle bal kabı rolündeki sistemlere yapılan bütün bağlantı teşebbüsleri şüpheli sayılır. Diğer bir amacı da saldırganların dikkatlerinin bu sistem üzerinde toplanmasını sağlamayarak ortaya çıkarılmaları için gereken zamanın kazanılmasını sağlamaktır.

Olay yanıtlama

Ortaya çıkarılmış tehlike potansiyeli olan olaya SSS'in olay işleme prosedürlerine göre velilen ilk tepkidir.

Gelişigüzellik

Normalde bir SSS ağ ara yüzü sadece bir konağa giden veya bir konaktan gelen verileri görür. Bu, gelişigüzel olmayan tanımına uyar. Bir ağ ara yüzü gelişigüzel yapılarak bulunulan ağ segmentindeki gelen ve giden veri paketlerinin hedef veya kaynak adresine bakılmaksızın tamamının görünmesi sağlanabilir. Bu ağ tabanlı SSS'ler için önemlidir fakat aynı şekilde paket dinleyiciler ağ trafiğini gözleyebilirler.

Yönlendirici

Yönlendirici, alt-ağların bağlanması için kullanılan aygıtlardır. 7 katmanlı OSI modelindeki iletim ve ağ katmanında çalışırlar. Daha basit olarak anlatılmak istenirse yönlendiriciler, paketlere hedeflerine doğru yönlenmeleri için yardım ederler. Aynı zamanda bir çoğu istenmeyen paketlerin filtrelenmesi için erişim kontrol listesine sahiptir. Yönlendiriciler tuttukları kayıtları SSS'lere iletebilirler.

Tarayıcı

Tarayıcılar ağı, sistemleri veya ağ zayıflıklarını bulmak için tarayan otomatik araçlardır.

İmza

SSS'ler için büyük öneme sahip olan saldırı imzaları SSS'in tetiklenmesini sağlarlar. Çok kısa olurlarsa SSS'in çok fazla tetiklenmesini sağlayabilirler çok uzun olurlarsa da SSS'i yavaşlatırlar.

Gizli arayüz

Gizli arayüzler SSS'lerin dış dünyaya görünmeden saldırıları ortaya çıkarmasını sağlarlar. Çoğu zaman silahsızlandırılmış alanın dışında, güvenlik duvarı korumasının ötesinde kullanılırlar [40].

5.3. Saldırı ve Saldırı Sezme

Saldırı, bilginin veya kaynakların gizliliğinin, bütünlüğünün ve ulaşılabilirliğinin tehdit edilmesidir [41].

Saldırı sezme, bir bilgisayar sisteminde veya ağında meydana gelen olayların gözlenmesi ve bu olayların bilgi gizliliğini, bütünlüğünü ve ulaşılabilirliğini tehdit edebilecek veya güvenlik mekanizmalarını bertaraf edebilecek nitelikteki saldırıların imzalarını taşıma durumlarının analiz edilmesidir [38].

Başka bir kaynakta da [42] saldırı sezme şu şekilde tanımlanmaktadır; Saldırı sezme, aktif olarak çalışan bir yazılım kullanarak, saldırı girişimlerinin

sezilmesi ve saldırı durumuyla ilgili olarak sistem yöneticisinin uyarılması işlemidir. Bazı durumlarda saldırganlara karşı veri alış verişinin engellenmesi gibi eylemler de uygulanabilir.

Saldırı sezmede sistem üzerinde gerçekleşen olaylar eşzamanlı veya eşzamanlılığa yakın bir şekilde gözetlenip incelenir [43].

5.4. SSS'lerin Kullanılma Nedenleri

SSS'ler organizasyonların, sistemlerini artan bağlanabilirliğin getirdiği tehditlerden korumalarını sağlar. Modern ağ güvenliği tehditleri karşısında güvenlik uzmanlarının cevaplamaları gereken soru SSS'leri kullanıp kullanmayacakları değil hangi SSS'i kullanacaklarıdır.

Geniş kitlelere yayınlanan teknikleri kullanan saldırganlar her sistemde olmasa bile ağa bağlı olan pek çok sistemde yetkisiz erişim hakkı kazanabilirler. Eğer bilinen sistem açıkları düzeltilmediyse bu sıklıkla olur. Bahsedilen açıklar şunlar olabilir;

- Pek çok eski sistemde işletim sistemi yamalanmamış yada güncellenmemiştir.
- Her ne kadar sistem yamaları kabul edilebilir ölçüde yapılmış olsa da sistem yöneticileri genellikle güncellemeleri yeterli ölçüde takip edecek zamanı bulamazlar. Çok sayıda konağın ve geniş bir yelpazeye yayılan farklı donanım ve yazılımların bulunduğu ortamlarda bu genel bir problemdir.
- Kullanıcılar ağ servislerini ve protokollerini saldırıya açık hale getirecek şekilde zorlayabilirler.
- Kullanıcılar ve ağ yöneticileri sistemleri konfigüre ederken veya kullanırken hata yapabilirler.
- Sistemlerin erişim kontrol mekanizmalarını organizasyonun bilgisayar kullanım poliçelerine yansıtırken ihtilaflar oluşabilir. Bu ihtilaflar kullanıcıların kimlik onaylama prosedürlerinden geçmeden bazı ayrıcalıklara sahip olmalarını sağlayabilirler.

İdeal dünyada, ticari yazılım üreticileri ürünlerindeki açıkları minimize ederler ve kullanıcılar rapor edilen bütün açıkları hızlı ve güvenilir bir şekilde düzeltirler. Fakat gerçek dünyada bu nadiren olur. Bu durumda SSS'ler bir sistemi koruma adına kusursuz bir yaklaşım olarak görülmektedir. Bir SSS düzeltilmemiş veya düzeltilemez bir sistem açığını suistimal ederek sistemi delmeye çalışan bir saldırganı yakalayabilir. Bunun yanı sıra sistem saldırıya uğradığında sistemi kurtaracak olan sistem yöneticisini haberdar etmek gibi bir görevi de yerine getirebilir.

Saldırganlar bir sisteme tipik olarak bilinen aşamalarla saldırırlar. Saldırının birinci aşaması genellikle sistemi veya ağı en iyi giriş noktasını bulmak için incelemektir. SSS bulunmayan sistemlerde saldırgan sistemi düşük bir yakalanma ve cezalandırılma riskiyle özgürce inceleyebilir. Bu özgür erişim, bahsedilen saldırganın sistemdeki yada ağdaki zayıf noktaları bulmasını sağlar.

SSS ile gözetlenen benzeri bir ağda saldırgan çok daha zor bir mücadelenin içindedir. Her ne kadar saldırgan ağı zayıf noktaları bulmak için tarasa da SSS bu incelemeleri görecek ve şüpheli olarak tanımlayacaktır. SSS saldırganın hedef sistemine erişimini engelleyebilir ve güvenlik personelini saldırgan hakkında uyarabilir.

SSS'lerin kullanım nedenlerinden birisi de var olan tehditlerin belgelendirilmesidir. Ağ güvenliği ile ilgili bütçe taslağı hazırlarken ağa yapılmış veya yapılmakta olan saldırlar ile ilgili kayıtların olması faydalıdır. Bu kayıtlar, ağa tahsis edilecek güvenlik önlemlerinin belirlenmesine yardımcı olacak olan, saldırıların sıklığı ve karakteristikleri ile ilgili bilgiler verirler.

SSS'ler organizasyonların ağları dışındaki veya içindeki tehditleri ayrıntılarıyla ortaya çıkararak, karakterize ederek ve tanımlayarak sistem yöneticilerinin organizasyon ağı için kullanılacak bilgisayar güvenliği kaynaklarının kullanımı ile ilgili doğru kararlar almalarında destekleyici rol oynarlar. SSS'lerin bu anlamda kullanımları önemlidir. Bazı insanların yaptığı gibi ağlara dışarıdan veya içeriden yapılan saldırıları göz ardı etmemek gerekmektedir. Ayrıca SSS'ler sistem yöneticilerine güvenlik stratejileri hakkında

karar verirken, gereksinimleri uygulamalı olarak gösterirler. Bu sayede sistem yöneticileri, tahmine dayalı veya gelenekselleşmiş stratejiler oluşturmak durumunda kalmazlar.

SSS'lerin olumlu özelliklerinden bir başkası da güvenlik tasarımı ve yönetimi için kalite kontrolü sağlamasıdır. SSS Ağda bir süre çalıştıktan sonra, sistem kullanımının yapısı ve problemleri açıkça ortaya çıkar. Tasarımdaki ve yönetimdeki eksiklikler SSS sayesinde görülerek bir sorun oluşmadan düzeltilebilir.

Gerçekleştirilen saldırılar hakkında kullanışlı bilgilerin sağlanması da yine SSS'ler ile mümkündür. Her ne kadar saldırıları önleyemeyebilseler de saldırılar ile ilgili detaylı ve güvenilir bilgi toplayabilirler. Bu bilgiler saldırının kontrol altına alınmasıyla ve verdiği zararların düzeltilmesi ile ilgili çalışmalarda destekleyici olurlar [38].

5.5. SSS'lerin Sınıflandırılmaları

SSS'lerin sınıflandırılması oldukça zor bir iştir. Zor olmasının asıl sebebi bir çoğunun birden fazla yaklaşımın üzerine kurulu olması ve uygulamalarının birden fazla metotla yapılabilmesidir. Sistemler, farklı teknikleri farklı bilgi işleme seviyesinde kullanabilmektedirler. Aynı zamanda farklı konfigürasyon parametreleri ile farklı çalışma kipinde olabilmektedirler. SSS'ler hakkındaki kaynaklarda [47,48] bulunan saldırı sezme sistemleri sınıflandırması Şekil 5.2 de gösterilmiştir.

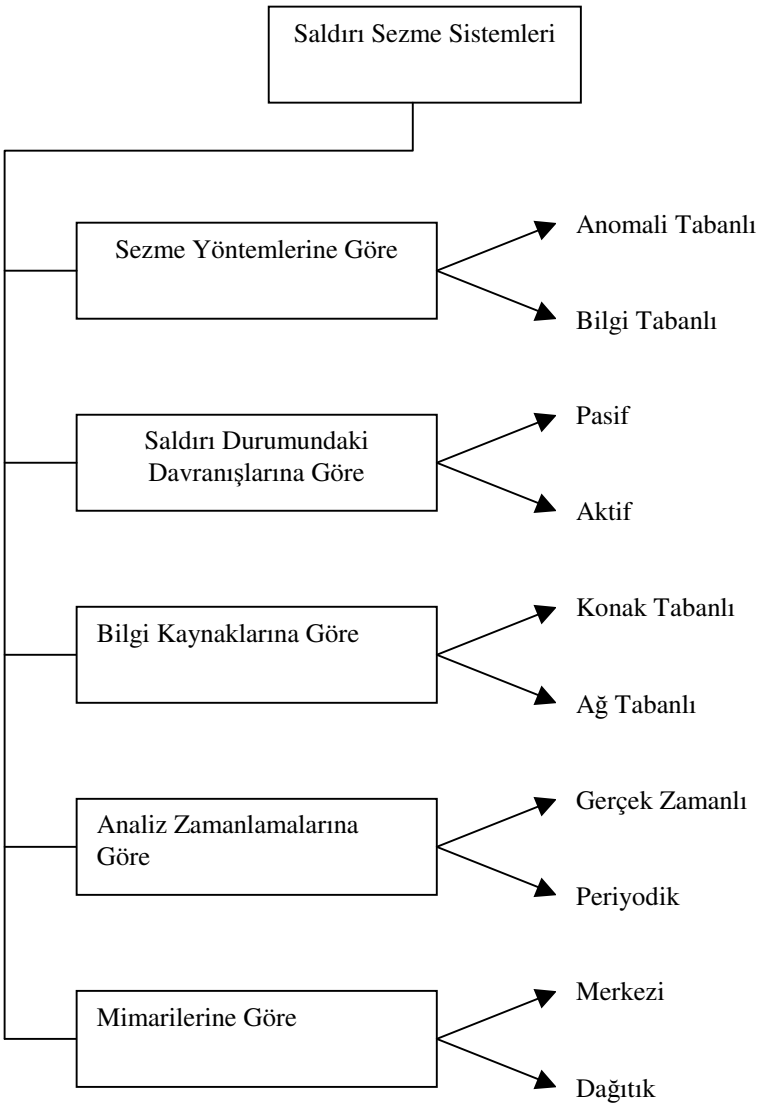
SSS'ler şu iki saldırı sezme yaklaşımından birini kullanabilirler: suistimal sezme ve anomali sezme.

Yakalanan saldırılara iki yolla tepki verebilirler. Tepki olarak sistem açıklarını kapatmak, servisleri durdurmak veya saldırgan ile ilgili kayıtlar tutmak gibi aktivitelerde bulunanlar aktif sistemler olarak adlandırılırlar. Eğer sadece bazı uyarılar üretiyorlarsa bunlara pasif sistemler denilir.

Bir başka karakteristiklerini ise kullandıkları bilgi kaynakları oluşturur. Konak tabanlı SSS'ler (vekil veya sensör olarak da adlandırılırlar) aktiviteler ile ilgili bilgileri sistemdeki belli bir konak üzerinden alarak analiz ederler. Ağ tabanlı SSS'ler ise ağ katmanında çalışırlar ve ağ trafiğini analiz ederler.

Bilgilerin analizi genellikle iki kipte yapılabilir. Saldırı sezme işlemi sürekli olarak çalışabilir. Buna gerçek zamanlı da denilir. Burada "gerçek zamanlı" terimi sadece SSS'in saldırılara "yeterince çabuk" tepki vermesi anlamındadır. Bunun yanı sıra saldırı sezme işlemi periyodik olarak da yapılabilir.

SSS'ler mimarilerine göre de karakterize edilebilirler. SSS'lerin gelişimi de bilgisayar sistemlerinin gelişimi ile aynı yolu takip etme eğilimindedir. Geleneksel SSS'ler merkezileştirilmişlerdir. Bu, bütün SSS işlevselliğini taşıyan tek bir modül olarak ya da birbiri ile iletişim halinde olan birkaç modül halinde oluşturulmuş olmaları anlamına gelir. Dağıtık SSS'ler ise sistem üzerine yayılmış ve kendi görevlerini taşıyan sistemlerden oluşurlar [46].



Şekil 5.2. Saldırı Sezme Sistemlerinin sınıflandırılması

5.5.1. Sezme yöntemine göre SSS'ler

Saldırı sezme için bütünleyici iki yaklaşım vardır. Birincisi, saldırıların bilinen saldırılar hakkında önceden toplanmış olan bilgiler ile karşılaştırılarak ortaya çıkarılmasına, ikincisi ise sistem kullanımının normal işleyişine göre sapmalarının gözlenmesine dayanır. Birinci eğilim genellikle suistimal sezme olarak adlandırılır. İkinci eğilime ise anomali sezme denilir [45].

5.5.1.1. Bilgi tabanlı SSS'ler

Bilgi tabanlı SSS'ler spesifik saldırılar veya sistem açıkları ile ilgili kayıtların tutulmasına ve saldırı sezme işleminin bu kayıtların kullanımı ile yapılmasına dayanır. SSS, bu açıklarla ilgili bilgileri ihtiva eder ve bunların suistimal edilme girişimlerini izler. Böyle bir girişim sezildiği zaman uyarı üretir. Diğer taraftan, net bir şekilde saldırı olarak tanımlanmayan bir faaliyet ise kabul edilebilir olarak görülür. Bu nedenle bilgi tabanlı SSS'lerin hassasiyet açısından başarılı oldukları düşünülür. Bununla birlikte eksiksiz çalışabilmeleri için, saldırılar hakkındaki bilgileri muntazam bir şekilde güncellenmelidir.

Bilgi tabanlı saldırı sezme yaklaşımının avantajı, çok düşük oranda yanlış alarm üretmesi ve saldırıların durum analizini detaylı bir şekilde yapabilmesidir. Bu güvenlik elemanının önleme ve kurtarma çalışmalarını kolaylaştırmaktadır.

Bilinen saldırılarla ve açıklarla ilgili bilgi toplayarak bu bilgileri güncel tutmanın zorluğu bilgi tabanlı saldırı sezme yönteminin dezavantajları arasındadır. Bilgi tabanlı SSS'lerin bakımı, her sistem açığının ve olası saldırının dikkatli analizini gerektirir. Bu oldukça zaman alıcı bir iştir. Bilgi tabanlı saldırı sezme yaklaşımı, genelleme sorunu ile de karşı karşıyadır. Saldırılar hakkındaki bilgiler işletim sistemine, versiyonuna, çalışma platformuna ve uygulamalara oldukça odaklanmış durumdadırlar. Bunun yanı sıra ağ içinden, ayrıcalıkların kötüye kullanılması ile yapılan saldırıların sezilmesi de oldukça zor olmaktadır. Çünkü böyle durumlarda hiçbir açığın suistimal edilmesi söz konusu değildir [45].

Bilgi tabanlı saldırı sezme için şu yaklaşımlar mevcuttur;

- Uzman sistemler
- İmza analizi
- Petri ağları
- Durum geçiş Analizi

Örnek olarak bu yaklaşımlardan uzman sistemler ve imza analizi incelenecektir.

Uzman sistemler

Uzman sistemler Bilgi Tabanlı Saldırı Sezme yöntemlerinde birincil olarak kullanılırlar. Uzman sistem saldırıyı tanımlayan bir kurallar dizisi içerir. Olaylar, uzman sistemin içinde bulunan anlamsal işaretlere dönüştürülerek gözden geçirilirler ve uzman sistemdeki ara yüz motoru bu kuralları ve olayları kullanarak karar oluşturur. Bu yöntem gözden geçirilen olaylara anlam verilmesinde olayların soyutlanma oranını artırır.

Uzman sistemler topladıkları bilgileri modellememek için kural tabanlı dilleri kullanırlar. Bu yaklaşım bilinen sistem açıkları ile ilgili kayıtların sistematik olarak taranmasını sağlar. Aynı zamanda bir organizasyonun güvenlik poliçesinin uygunluğunun doğrulanması için de kullanılırlar.

İmza analizi

İmza analizi yaklaşımında uzman sistemlere benzer bir yöntem kullanılır. Fakat eldeki bilgi farklı bir yolla işlenir. Bu yaklaşımda saldırıların oluşturacağı numune veri dizileri yani imzalar, kayıtların içerisinde doğrudan aranırlar.

Bu teknik oldukça etkili bir uygulama sağlar. Bu nedenle ticari saldırı sezme ürünlerinde yaygın olarak kabul görmüştür [45]. Bütün bilgi tabanlı saldırı sezme yaklaşımlarında olduğu gibi, bu yaklaşımın da en büyük dezavantajı yeni keşfedilen sistem açıklarının ortaya çıkaracağı saldırı imzalarını karşılayacak şekilde güncelleme yapmanın mecburi olmasıdır.

5.5.1.2. Anomali tabanlı SSS'ler

Anomali tabanlı SSS'ler bir saldırının, sistemin veya kullanıcıların normal veya beklenen davranışların dışına çıkmaları halinde keşfedilebileceğini varsayarlar.

Öncelikle beklenen veya normal sayılan davranış modelini oluşturmak için gerekli olan verileri çeşitli yollardan edinip bir davranış modeli oluştururlar. Daha sonra, var olan aktiviteleri bu modelle karşılaştırırlar. Eğer gerçekleşmekte olan aktivitelerde normalden sapma durumu gözlenirse uyarı üretirler. Diğer bir deyişle eğer bir aktivite önceden öğrenilenlere göre farklılık gösterirse saldırı olarak değerlendirilir. Tablo 5.1 de kullanıcılardan beklenilen normal davranışlara, Tablo 5.2'de de aynı kullanıcıların anormal sayılabilecek davranışlarına örnekler verilmiştir.

Tablo 5.1. Kullanıcılar ve normal sayılabilecek davranışları

Kullanıcı	**Normal davranış**
Sistem Yöneticisi	Yönetici şifresiyle giriş yapar, sistemin şifrelerinin tutulduğu veritabanına erişir, kullanıcı erişim yetkilerini düzenler, sistem konfigürasyon ve izleme araçlarını çalıştırır.
Patron	Sisteme zaman zaman giriş yapar, haftada bir e-mail okur ve cevaplar.
Sekreter	Şirketin çalışma saatleri içinde sisteme lokal olarak giriş yapar, kelime işlem yazılımını kullanır, sıklıkla e-mail okur ve gönderir.
Yönetici	Sabahın erken saatlerinden gece geç saatlere kadar sistemde kalır, yönetim araçlarını ve İnternet tarayıcısını kullanır, sıklıkla e-mail okur ve gönderir.
Programcı	Sabah geç saatlerden gece geç saatlere kadar sistemde kalır, yazılım geliştirme araçlarını çalıştırır.

Tablo 5.2. Kullanıcılar ve anomali içeren davranışları

Kullanıcı	**Anomali içeren kullanıcı davranışı**
Sistem Yöneticisi	Bir programcı gibi yazılım geliştirme araçlarına ve bazı projelere ait olan yazılım kodlarına erişir.
Patron	Gece yarısında sisteme girer ve yoğun biçimde e-mail gönderir, gizli bilgilere ulaşır.
Sekreter	Sisteme uzak bir konaktan giriş yapar, yönetici gibi davranır.
Yönetici	Sistem yöneticisi olmaya çalışır, kullanıcıların ağ üzerindeki yetkilerinde değişiklikler yapar.
Programcı	Bir sekreter gibi personel veritabanına erişir.

Anomali tabanlı SSS'lerin avantajı, henüz bilinmeyen sistem açıklarına yapılan suistimal girişimlerini de bulabilmeleridir. Ayrıca işletim sistemine özel mekanizmalara daha az bağlıdırlar. Bunların yanı sıra, gerçekte hiçbir sistem açığını suistimal etmeyen "yetkinin kötüye kullanımı" tipindeki saldırıları da yakalayabilmektedirler.

Yanlış uyarı (yanlış pozitif) üretme seviyelerinin oldukça yüksek olması, anomali tabanlı SSS'lerin en büyük dezavantajlarıdır. Ayrıca sistemlerin veya ağların davranışları zamanla değişebilir. SSS'lerin buna uyum sağlaması için periyodik olarak yeniden eğitilmeleri, beraberinde SSS'lerin bir süreliğine de olsa etkin olamamalarını veya daha fazla yanlış uyarı üretmelerini getirecektir. Bu tip SSS'lerde hassasiyet oluşturmak zordur [45].

Anomali tabanlı saldırı sezme için şu yaklaşımlar mevcuttur;

- İstatistiklerin kullanımı
- Uzman sistemler
- Yapay sinir Ağları
- Kullanıcı niyetinin tanımlanması

- Sistem bağışıklığı

Örnek olması amacıyla anomali sezme tabanlı SSS'lerde istatistiklerin ve yapay sinir ağları'nın kullanımı incelenecektir.

İstatistiklerin kullanımı

Anomali tabanlı SSS'lerde kullanılan en yaygın araç istatistiklerdir [45]. Bu yaklaşımda kullanıcıların veya ağın davranışları birçok değişkenle zamana bağlı olarak örneklenir. Bunlara her oturumun açılması ve kapanması arasında geçen süre, kaynakların kullanım süreleri ve kullanılan disk-işlemci zamanı-hafıza miktarları örnek olarak verilebilir. Burada kullanılan zaman örneklemesi birkaç dakikadan birkaç haftaya kadar uzanan oldukça geniş bir aralıkta değişebilir. Daha sonra ağ, kullanıcılar veya uygulamalar, bu örnekler ile istatistiksel anlamda karşılaştırılarak olağan dışı durumların görülmesi halinde uyarı üretilir.

Yapay sinir ağlarının kullanımı

Yapay sinir ağları, giriş çıkış vektörleri arasındaki ilişkiyi öğrenip bu ilişkiyi mantıksal bir yolla genelleyerek farklı giriş-çıkış vektörlerine uygulayabilen sistemlerdir. Teorik olarak yapay sinir ağları anomali sezme tabanlı SSS'lerde saldırıların izlerini öğrenecek şekilde kullanılabilirler. Bununla birlikte uyarı durumunda, uyarının oluşma nedeni ile ilgili bilgi edinmek için kullanışlı değildirler. Çünkü yapay sinir ağları, saldırı ile ilgili bir muhakeme veya açıklama sunmazlar.

Bu nedenle saldırı sezme için yapay sinir ağlarının kullanımı, büyük ölçüde ağdaki kullanıcıların ve programların davranışlarını öğrenmeleri amacını taşır. Saldırı sezmede yapay sinir ağlarının kullanılması yaklaşımı, değişkenler arasındaki doğrusal olmayan ilişkileri açıklama konusunda daha başarılı olmaları ve eğitilme/yeniden eğitilme işlemlerini otomatik olarak yapabilmeleri yönleri ile istatistiksel yaklaşıma göre daha avantajlıdırlar [45].

5.5.2. Saldırı durumundaki davranışlarına göre SSS'ler

SSS'ler saldırı durumundaki davranışlarına göre pasif SSS'ler ve aktif SSS'ler olmak üzere ikiye ayrılırlar [47].

5.5.2.1. Pasif SSS'ler

SSS, bir saldırının sezilmesi durumunda uyarı üretiyor, fakat atağa karşı bir tedbir almıyor ise pasif olarak adlandırılır. Çoğu SSS pasiftir.

5.5.2.2. Aktif SSS'ler

Bazı SSS'ler bilgisayarlarda veya ağda güvenlik sorunu ile karşılaşmaları halinde aktif olarak önlem alabilme kapasitesine de sahiptirler. Bu tip SSS'ler dosya sistemindeki yetkilendirmelerin değiştirilmesi gibi sistem açıklarını kapatma amaçlı komutlar üretebilir veya sistemi tekrar değişimden önceki haline döndürebilirler. Bunun gibi, uyarı üretmenin yanı sıra sistem güvenliğini sağlayıcı aktivitelerde bulunan SSS'ler aktif SSS'ler olarak adlandırılırlar [45].

5.5.3. Bilgi kaynaklarına göre SSS'ler

İlk saldırı sezme araçlarının dizayn edildiği zamanlarda güvenliğinin sağlanması hedeflenen yapı, bütün kullanıcıların lokal olarak merkezi bir bilgisayarı kullandığı bir sistem olarak düşünülmüştür. Dışarı ile seyrek olarak etkileşime giren bu yapı, SSS'lerin görevlerini oldukça basit kılmaktaydı. Saldırı sezme aracı, sistem kayıtları lokal olarak merkezi bilgisayardan veya ayrı bir makine üzerinden alarak, güvenilirliğinden şüphe edilen durumlar için uyarı üretmekteydi.

Bilgisayar sistemleri, merkezi yapıdan uzaklaşıp dağıtık iş istasyonlarının oluşturduğu bir yapı almaya başlayınca, bu yapıya uygun olarak çalışan çeşitli SSS prototipleri geliştirildi. Bu alandaki ilk araştırmaların sonucunda konak tabanlı SSS'ler geliştirilmişlerdir.

İnternet kullanımının yaygınlaşması, SSS'lerin ağın kendisine yapılan saldırılar üzerine odaklanmasını sağlamıştır. Üçüncü bölümde örnekleri verilen bazı ağ saldırılarının, sistem kayıtlarını inceleyerek tam anlamıyla sezilmesi

mümkün değildir. Bu nedenle ağı izleyerek, ağ üzerindeki paketleri gerçek zamanlı olarak yakalayıp bu saldırıları arayan özel araçlar geliştirilmiştir. Bu şekilde çalışan sistemler bilgi kaynaklarına göre ağ tabanlı SSS olarak adlandırılmışlardır [45].

Bu anlamda bilgi kaynaklarına göre iki farklı SSS tipi oraya çıkmıştır denilebilir;

- Konak tabanlı SSS'ler
- Ağ tabanlı SSS'ler

Bazı kaynaklarda [48], hem ağ hem de konak tabanlı SSS'lerin birlikte kullanıldığı melez yapıların bahsi geçmiştir. Ancak burada melez yapıdaki SSS'ler üzerinde ayrıca durulmayacaktır.

5.5.3.1. Konak tabanlı SSS'ler

Konak tabanlı SSS'ler bulundukları konaktaki kullanıcı ve sistem aktivitelerini izlemek ve olası saldırıları ortaya çıkarmak için dizayn edilmişlerdir.

Ağ tabanlı SSS'lerden farklı olarak, ağ üzerindeki konaklar arasında iletilen bütün verileri değil, sadece bulundukları konağı ilgilendiren verileri incelerler [49].

Konak tabanlı SSS'lerin en büyük avantajları bir saldırının başarıya ulaşıp ulaşmadığını kesin olarak belirleyebilmeleridir. Bir ağ tabanlı SSS saldırı durumunda uyarı üretebilir ancak her zaman saldırının başarıya ulaşıp ulaşmaması ile ilgili kesin bir yargıya varamaz. Bunun yanı sıra konak tabanlı SSS'ler tezin ikinci bölümünde anlatılmış olan fragmentasyon saldırıları için endişelenmek zorunda değildirler. Çünkü konağın kendi yığını bu sorunun üstesinden gelir. Son olarak konak tabanlı SSS'lerin bir avantajı da şudur; ağ trafiğinin şifrelenmiş olması durumunda konak tabanlı SSS'ler gözledikleri sistem üzerindeki veri trafiğini deşifrelenmiş olarak görebilmektedirler.

Konak tabanlı SSS'ler iki önemli dezavantaja sahiptirler. Bu dezavantajlar şunlardır;

- Ağın noksan tasviri
- Çoklu işletim sistemi desteğinin gerekmesi

Konak tabanlı SSS, sadece konak seviyesindeki bilgileri incelediği için ağın tamamında olan olayların tasvirini yapmakta zorluk çekmektedir. Diğer bir zorluk da konak tabanlı SSS'in ağdaki her sistemde çalışmasının gerekmesidir. Bu, SSS'in ağdaki farklı her işletim sistemini desteklemesini gerektirmektedir.

5.5.3.2. Ağ tabanlı SSS'ler

Ağ tabanlı SSS'lerde konak seviyesindeki saldırı sezmenin yerine ağ üzerinde hareket eden veri paketleri incelenir. Sistem bu trafiği inceleyerek saldırgan bir aktivitenin işaretlerini arar ve bulması durumunda saldırı uyarısı üretir. Çoğu zaman ağ tabanlı SSS'ler, saldırıların başarılı olup olmadıklarına dair bir karara varamazlar. Sadece var olan saldırgan aktivitelere dikkat çekerler.

Ağ tabanlı SSSler, bütün ağ üzerindeki saldırıları kolaylıkla görme ve koordine etme ayrıcalığına sahiptirler. Ayrıca, sadece bir sistemin üzerinde çalışan SSS'in bütün ağı gözleyebilmesi, ağda kullanılan bütün işletim sistemlerine destek verme zorunluluğunu ortadan kaldırmaktadır.

Ağ üzerindeki paketlerin şifreli olarak akması Ağ Tabanlı SSS'in bu paketleri büyük ölçüde görmemesini sağlar. Parçalanmış paketlerin birleştirilmesi de ağ tabanlı SSS için çözülmesi gereken bir problemdir. Muhtemelen ağ tabanlı SSS için en büyük dezavantaj, ağın hem bant genişliği hem de fiziksel yayılma anlamında büyümesidir. Böylesi bir büyüme durumunda ağ tabanlı SSS'in bütün paketleri yakalayacak bir noktaya yerleştirilmesini zorlaşmaktadır. Bu durumda ağ üzerinde daha fazla sensör kullanmak zorunda kalınacaktır [50].

5.5.4. Analiz zamanlamalarına göre SSS'ler

Bölüm 4.5 de değinildiği gibi SSS'ler analiz zamanlamalarına göre gerçek zamanlı SSS'ler ve periyodik SSS'ler olmak üzere iki kategoriye ayrılırlar.

5.5.4.1. Gerçek zamanlı SSS'ler

Bulundukları sistem veya ağ üzerindeki paketleri neredeyse gerçek zamanlı olarak izleyen SSS'lerdir. Ağ üzerinde gidip gelmekte olan paketleri yakalayarak incelerler.

En önemli avantajları sürmekte olan saldırılara karşı hemen tepki verebilmeleridir. Ayrıca periyodik SSS'lerle sezilemeyecek olan, bilgisayar ağlarının yapılarından kaynaklanan sistem açıklarından faydalanılarak yapılan servis dışı bırakma türündeki saldırıları da sezebilmektedirler.

Gerçek zamanlı SSS'lerde kurulması gereken bir denge vardır. Çünkü hızlı bir saldırı sezme sağlayabilmek için basit olma zorunluluğu var iken, etkili bir saldırı sezme için karmaşık veri işleme prosedürlerini de gerçekleştirmeleri gerekmektedir. Aynı zamanda gerçek zamanlı SSS'ler büyük miktarda belleğe ihtiyaç duyarlar. Yeterli belleğin olmaması durumunda bazı paketleri atlayabilirler [51].

5.5.4.2. Periyodik SSS'ler

Periyodik SSS'ler belirli aralıklarla (periyodik olarak) bulundukları sistemin veya ağın anlık durumu ile ilgili verileri alarak bu veriler üzerinde saldırı sezme işlemlerini gerçekleştirirler. Bu anlık durum verilerinde yazılım zayıflıkları ve konfigürasyon hataları gibi açıkları ararlar. Bu kontroller, uygulamaların en son yamalarla güncellenip güncellenmediğini, zayıf şifrelerin kullanılmakta olup olmadığını, kullanıcıların klasörlerindeki dosyaların içeriklerini ve açık ağ servislerinin konfigürasyonunun doğru yapılıp yapılmadığını da içerir. Fakat bu kontrollerin sonuçları sadece yapıldığı an için geçerlidir [45].

Periyodik SSS'ler anlık durum değerlendirmesini günlük denilen olay kayıtlarına bakarak yaparlar. Bu kayıtların bir tek dosyada tutulmasından

kaçınılmalıdır. Çünkü saldırganlar bu kayıtlara ulaşarak istenmeyen değişiklikler yapabilirler. Günlük dosyalarının kopyalarının ağ üzerindeki birden fazla sisteme dağıtılması çok daha iyidir. Ancak bu, sistem ve ağ üzerine ek bir yük getirecektir.

Olaya fonksiyonellik açısından bakılırsa var olan her olayın günlüklere kaydedilmesi sistem ve ağ kaynaklarına oldukça büyük bir yük getirecektir. Bu anlamda daha çok kayıt tutabilmek için günlük sıkıştırması yapılabilir. Özel durumların kayıt dışında bırakılmasını sağlamak da bir çözüm gibi görünse de bazı saldırıların gözden kaçmasını sağlayabilir. Kayıt depolama periyodu ile ilgili uygun bir ayarlama yapmak kaynakların daha iyi kullanılmasını sağlayacaktır.

Periyodik SSS'lerin avantajları şunlardır;

- Saldırı sonrasında saldırı ile ilgili analiz yapılabilir.
- Tekrarlanan saldırılar belirlenir.
- Başarıya ulaşan saldırganlar hakkında tanımlayıcı bilgiler edinilir.
- Sahip olunan sistemdeki açıklar belirlenir.
- Anomali sezme tabanlı SSS'ler için gerekli olan veriler temin edilir.
- Günlük kayıtlarının tutuluyor olması saldırganlar üzerinde caydırıcı etki yaratır.
- Büyük kapasitede bilgi tutulabilecek olması yapay zeka ve veri madenciliği gibi mekanizmaların kullanılmasına imkan sağlar

Günlükler üzerinde çalışan periyodik SSS'ler, üçüncü bölümde anlatılan servis dışı bırakma tipindeki saldırılara maruz kalabilirler. Bu saldırılar kayıt mekanizmasını doldurarak sistemde boş alanın kalmamasını sağlarlar [51].

5.5.5. Mimarilerine göre SSS'ler

Mimarilerine göre SSS'ler aşağıdaki gibi iki sınıfa ayrılabilirler;

- Merkezi SSS'ler
- Dağıtık SSS'ler

5.5.5.1. Merkezi saldırı sezme sistemleri

Mimari anlamda merkezi olarak yapılandırılmış SSS'lerde veri analizi, gözetlenen konak sayısına bağlı olarak belirli noktalarda toplanmıştır. Bu SSS'lerde merkezi noktalarda toplananlar, veri toplama bileşenleri değil analiz bileşenleridir. Merkezi SSS'lerde de veri toplama bileşenleri ağ üzerine dağıtılmış olabilir [52].

Merkezileştirilmiş yaklaşımın getirdiği dezavantaj saldırılara karşı açık olmasıdır. Eğer bir saldırgan, merkezi SSS'i devre dışı bırakabilirse, koruma tamamen olmasa da büyük ölçüde ortadan kalkar [54].

5.5.5.2. Dağıtık SSS'ler

Dağıtık saldırı sezmede, saldırı sezme modülleri ağ üzerinde farklı noktalara dağıtılmışlardır. Söz konusu olan modüller, merkezi bir kontrolör ile birlikte çalışır ve saldırı sezme için gerekli olan analizleri yaparlar. Bu, farklı alt ağlar ve farklı konaklardaki saldırıları sezmek için güçlü bir yapı sağlar. Ancak, böyle bir yapının merkezi kontrolörlüğü için bir adanmış sunucu gerekir. Ayrıca merkezileştirilmede saldırılar karşısında zayıflık sağlayabilir.

5.6. SSS'lerin Verimliliğini Belirleyen Unsurlar

Bir SSS'in verimliliğini şu üç unsur belirler [45];

- Hatasızlık
- Performans
- Etkinlik

Hatasızlık

Saldırı olmayan bir durumun saldırı olarak değerlendirilmesi veya anomali niteliği taşımayan bir durum hakkında anomali olduğu kararının alınması saldırı sezme sisteminin yapabileceği hatalardır. Bu hatalar yanlış pozitif olarak da adlandırılırlar. Saldırı olan bir durum hakkında saldırı olmadığı kararına varılması

da SSS'lerin yapabileceği hatalar arasındadır. Bu tipteki hatalara da yanlış negatif denilir.

Performans

Bir SSS'in performansını birim zamanda inceleyebildiği sistem kaydı miktarı belirler. Eğer bir SSS'in performansı düşük ise gerçek zamanlı saldırı sezme mümkün olmaz.

Etkinlik

Etkin olmayan bir SSS, bazı saldırı durumlarını belirleyemeyebilir. Bu unsurun iyileştirilmesi diğer iki unsura göre daha zordur, çünkü saldırılar ve suistimaller hakkında küresel anlamda bütün bilgilere sahip olmak mümkün değildir.

Bunlara ek olarak iki unsurun daha SSS'in verimliliğini belirlediği söylenebilir. Bu unsurlar şunlardır;

- Hata toleransı
- Vakitlilik

Hata toleransı

Bir SSS'in verimli çalışabilmesi için öncelikle kendisinin saldırılara karşı direnişli olması gerekir. Özelikle de hizmet dışı bırakma saldırılarına karşı koyabilmelidir. Bu özellikle önemlidir, çünkü çoğu SSS açıkları bulunan ticari işletim sistemlerinin ve donanımların üzerine kurulmuştur.

Vakitlilik

SSS, var olan saldırıyı mümkün olduğu kadar çabuk işlemelidir. Güvenlik elemanını sistem çok fazla zarar görmeden ve saldırganın kayıtlar üzerinde değişiklik yapmasına zaman vermeden uyarmalıdır.

5.7. SSS'lere Karşı Yapılabilecek Saldırılar

SSS'ler yapılarından dolayı bazı saldırılara açık olabilmektedirler. Bu bölümde SSS'lerin işlevlerini yerine getirememelerini sağlayabilecek olan yaygın üç saldırı yöntemi örnek olarak incelenecektir. İncelenecek olan saldırılar şunlardır;

- Hizmet dışı bırakma
- Araya yerleştirme
- Kaçamak yapma

5.7.1. Hizmet dışı bırakma

Bölüm 3'de de değinildiği gibi, hizmet dışı bırakma saldırıları sistem kaynaklarının hazır bulunabilirliğini hedef alır. Bu saldırılar, sistemin gereksiz işlemlerle aşırı miktarda yüklenmesini ve görevini yerine getirememesini sağlarlar. Hizmet dışı bırakma saldırılarına ağa yoğun bir şekilde paket gönderilmesi örnek olarak gösterilebilir. Ayrıca, yazılım veya donanım hataları suistimal edilerek de hizmet dışı bırakma saldırıları yapılabilir.

SSS'ler sistemlere gelip giden paketleri izlemek zorunda olduklarından dolayı, ağ üzerinde yoğun paket trafiği oluşturan bir hizmet dışı bırakma saldırısından oldukça etkilenebilirler. Aşırı miktardaki paket trafiğini karşılayacak işlem hızına sahip olmayan SSS'ler hizmet veremez duruma gelebilirler. Hele de sistem konfigürasyonunda değişiklik yapamayan pasif SSS'ler bu tür saldırılara karşı savunmasızdırlar.

Maalesef hizmet dışı bırakma saldırılarına karşı savunma yapmak son derece zordur. Kaynakların kısıtlılığı sorunu kolayca çözülebilir değildir. Saldırılar SSS'in kendi kendini çökertmesini sağlayabilirler.

5.7.2. Araya yerleştirme

Bir SSS, son sistemin (konağın) reddettiği veri paketini kabul edebilir. Bu gibi durumlarda SSS yanlışlıkla, son sistem almasa da, son sistemin veri paketini alıp işleme koyduğunu varsayar. Saldırganlar SSS'in geçerli sayacağı ancak son

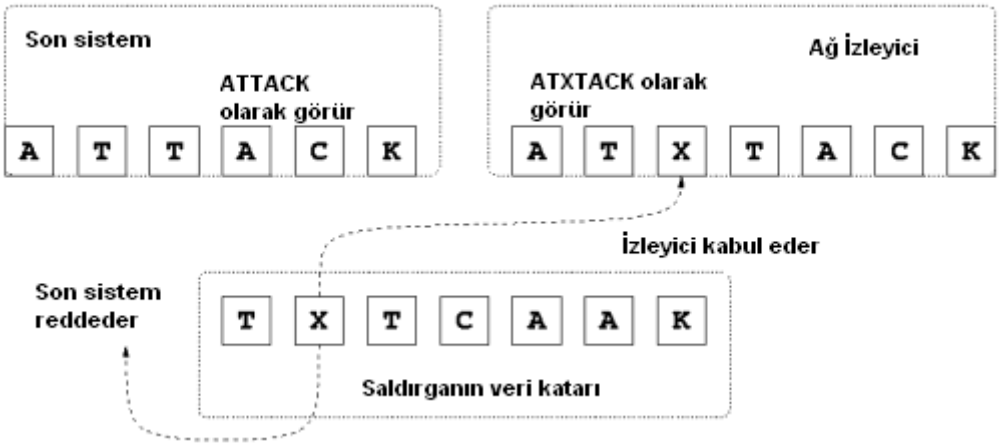
sistemin reddedeceği paketler göndererek bu durumu suistimal edebilirler. Saldırganlar bunu yaparak SSS'e ağdaki diğer bilgisayarları ilgilendirmeyen bilgi girişi yapabilirler.

Bu saldırılara "araya yerleştirme" saldırıları denmektedir ve testler sırasında SSS'lerin en zayıf olduğu saldırılar arasındadırlar. Saldırganlar araya yerleştirme saldırılarını imza analizini aşarak SSS'i etkisiz kılmak için kullanabilirler.

Araya yerleştirme saldırılarının imza analizini nasıl aştığını anlamak için gerçek SSS'lerde imza analizinin nasıl kullanıldığını bilmek önemlidir. İmza analizinin en önemli kısmında bir karşılaştırma algoritması ile verilerin içinde belli karakter dizileri aranır. Örneğin bir SSS, phf saldırısını yakalamak için içinde HTTP GET istemi olmayan "GET /cgi-bin/phf?" benzeri bir "phf" katarı arar.

SSS "phf" katarını HTTP istemi içinde basit bir katar araması ile bulabilir. Fakat, saldırgan aynı istemi sunucudan defalarca yaptığı zaman problemin çözülmesi oldukça zorlaşır. SSS, katarı örneğin "GET /cgi-bin/pleasedontdetectttthisforme?" olarak görmeye zorlanabilir. Saldırgan araya yerleştirme saldırısını "leasedontdetectt", "is" ve "orme" katarlarını orijinal katara eklemek için kullanabilir.

Şekil 4.3. aynı saldırının basit bir örneğini göstermektedir. Saldırgan, bir harfinin (x harfi) yanlızca SSS tarafından kabul edileceği karakter paketleri ile SSS'i yanıltabilir. Sonuç olarak SSS ile son sistem iki farklı veri katarı oluştururlar. Genellikle araya yerleştirme saldırıları SSS'in paketleri incelemede son sisteme göre daha az titizlik gösterdiği zaman gerçekleşir. Şu aşikardır ki bu sorun, SSS'in paket incelemedeki titizliği arttırılarak en aza indirgenebilir.



Şekil 5.3. Araya yerleştirme saldırısı

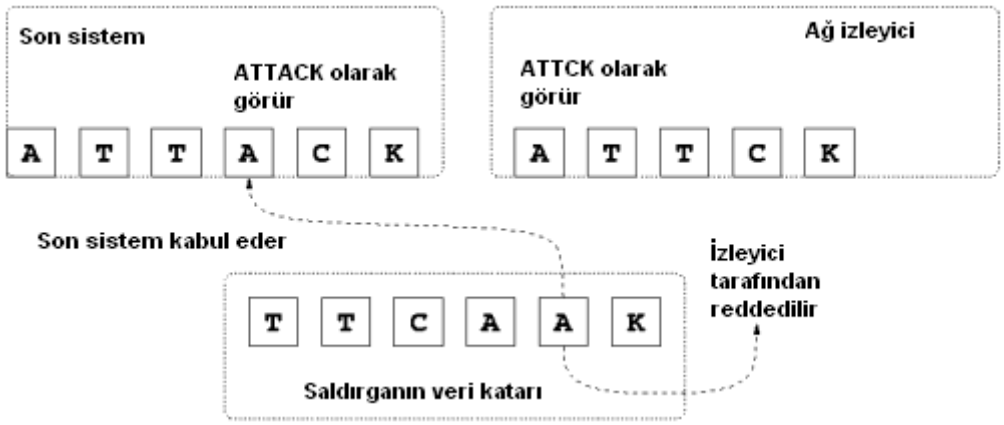
5.7.3. Kaçamak yapma

Bir SSS'in son sistemin reddettiği bilgiyi kabul etmesi gibi bir son sistem de bir SSS'in reddettiği bilgiyi kabul edebilir. Paketi kabul etme konusunda SSS'in son sisteme göre daha titiz olması durumu suistimal edilerek paketlerin SSS'i geçmesi sağlanabilir. Bu paketler SSS'in kontrolünden "kaçmaktadırlar".

Bu tip saldırılara kaçamak yapma adı verilir. SSS'lerin aşılması için sıklıkla kullanılan bir yöntemdir. Bu yöntemle bir oturum tamamen SSS'in kontrolü dışında gerçekleştirilebilir.

Kaçamak yapma saldırıları imza analizini etkisiz kılmak için araya girme saldırısındaki yönteme benzer bir yöntem kullanırlar. Saldırganlar bu yöntemde de son sistem ile SSS'in farklı bilgileri görmesini sağlarlar. Ancak, kaçamak yapma saldırısında, araya girme saldırısından farklı olarak son sistem SSS'den daha çok bilgi görür.

Bölüm 4.7.2 deki araya girme saldırısında saldırgan bir HTTP istemi göndermektedir fakat istem verisindeki fazladan bilgi onu zararsız göstermektedir. Kaçamak yapma saldırısında ise saldırgan, aynı veri paketlerini yineleyerek SSS'in yanlışlıkla reddetmesini ve son sistemin kabul ettiği paketlerden haberdar olmamasını sağlamaktadır. Şekil 4.4. bu durumu göstermektedir [54].



Şekil 5.4. Kaçamak yapma saldırısı.

5.8. SSS'lerin Geleceği

SSS'ler son yıllarda güvenlik endüstrisinin gelişimine paralel bir şekilde hızla gelişmişlerdir. Bu araçlar, pek çok organizasyon için güvenlik duvarları kadar önemli araçlar haline gelmişlerdir. Fakat zamanla bazı şeyler değişmiştir. Ağlar ve saldırganlar hızla evrimleşmiş, buna bağlı olarak güvenlik araçları da gelişmiştir. Günümüz SSS'leri pek çok problemle karşı karşıyadır. Ancak, ağ güvenliği anlamında gelecekte yaşanacak olan mücadelelerde kullanılacak olan en önemli silahlar SSS'ler olacaktır.

5.8.1. SSS'lerin geçmişi ve bugünü

Tezin önceki bölümlerinde de bahsedildiği gibi SSS'ler makineleri veya ağları, olası saldırı girişimlerini, anomali durumlarını ve genel suistimalleri yakalamak amacı ile gözleyen araçlardır. Pek çok kişi için SSS terimi Snort, Shadow gibi ağ tabanlı araçları çağrıştırır. Şu ilginçtir ki, SSS'lerin erken dönemdeki formları yirmi yıl kadar öncesine uzanır. Bu SSS'ler, Swach gibi modern kayıt analizi programlarına benzer bir şekilde güvenlik olaylarını, sistem kayıtlarını veya kullanıcı hesap dosyalarını inceleyerek keşfetmekteydiler. Bu programlar evrimleştiler ve gerçek zamanlı sistem kayıtlarını incelemek ve ayrıntılı sistem kontrolleri yapmak gibi işlevler kazanarak konak tabanlı SSS adını aldılar.

Daha yakın zamanlarda ağ tabanlı SSS'lerin gelişimine tanık olundu. Bu uygulamalar ağ izleyicileri gibi ağ paketlerini yakalayarak bunların saldırı niteliği taşıyıp taşımadıklarını inceleyebilmekteydiler. Bugün ağ tabanlı SSS'ler en iyi savunma araçları olarak hizmet vermektedirler.

5.8.2. Modern SSS'lerdeki problemler

Bölünmüş yapıdaki ağ ortamları hızla artmaktadır. Aynı şekilde ağ üzerindeki trafik de günden güne çoğalmaktadır. Bu gibi ortamlarda SSS'ler bazı ağ paketlerini kaçırabilirler. Şu aşikardır ki her SSS limitli miktarda trafikle baş edebilecek hıza sahiptir. Paketlerin kaçırılması durumu da SSS'in paket analiz hızının ağ trafiğinden daha yavaş kaldığı zamanlarda oluşmaktadır. Zaten ticari SSS'lerin analiz hızlarındaki performansları en önemli satış noktalarıdır. Bu ürünlerden bazıları gigabit seviyelerinde hıza sahiptirler. Fakat daha hızlı trafik beraberinde daha fazla yanlış pozitif getirir. SSS kullanma deneyimine sahip birisi bilir ki incelenilen uyarıların, kayıtların ve işaretlenmiş trafiğin büyük bir kısmı zararsız olaylardır. Bu sorunun nasıl halledilebileceği sorusunun cevabı olarak SSS'lerin titizlikle ayarlanması gerekmektedir denilebilir. Ancak yinede daha fazla trafik daha çok yanlış alarm üretilmesine sebep olur. SSS'ler gereğinden fazla özelleştirilirlerse de bazı güvenlik sorunlarını gözden kaçırabilirler.

SSS'lerin performansları ile ilgili meseleler genel sorunlardır. Pek çok ağ yöneticisi yanlış pozitiflerin fazlalığından şikayet etmektedir. Ancak yanlış pozitifler gibi SSS'lerin hızları da satıcılar ve açık kaynak geliştiricileri tarafından zamanla arttırılmaktadırlar.

5.8.3. Yakın gelecekte SSS'ler

Yakın gelecekte SSS'ler ile ilgili olarak, parçalanmış ağların ve artan veri trafiğinin oluşturduğu problemler üzerinde durulacaktır. Şüphesiz ki SSS üreticileri ve gelişen donanım teknolojisi artan veri trafiği sorununun üstesinden gelebilecektir. Yüksek performanslı yazılımlar ve donanımlar yüksek fiyatlı olacak, ancak organizasyonlar bu fiyatları karşılayabileceklerdir. Parçalı yapıdaki ağlarla ilgili olarak da şimdiden bazı çözümler üretilmeye başlanmış durumdadır.

Örneğin Hoghwash adındaki Snort tabanlı bir SSS iki ağ parçası arasında konumlandırılarak iki taraftaki verileri de görebilecek yapıdadır.

5.8.4. Uzun vadede SSS'lerin gelişimi

SSS'lerin geleceği veri kolerasyonunda yatmaktadır. Yarının SSS'leri, veri girdisini farklı birçok kaynaktan sağlayacaktır. Konak ve ağ tabanlı SSS'ler ortadan kalkacak, bunların yerine dağıtık ve spesifik görevlere sahip bir çok unsura sahip olan SSS'ler kullanılacaktır.

Bu senaryoda konak tabanlı SSS'ler önemli rol oynayacaklar. Şifreli paketlerin kullanılması, saldırı sezmenin konak üzerinde yapılmasını zorunlu kılmaktadır. Buna başka bir çözüm bulmak zordur. Bu yapı ayrıca bir avantaja daha sahiptir. Her saldırı imzası bulunduğu sisteme özgüdür. Örneğin Windows tabanlı bir sistem ile Unix tabanlı bir sistem üzerine yapılabilecek saldırılar farklı imzalara sahip olacaklardır. Konak tabanlı SSS'ler ile bu farklılığın oluşturduğu sorunun da üstesinden gelinecektir. Bu SSS'ler bundan başka, sistem üzerinde çalışan uygulamaların bilinmesini ve kesin kural dizilerinin tanımlanmasını da sağlayacaklardır. Ağ üzerindeki tüm trafiği yakalayan sensörlerden farklı olarak sadece bulundukları konaktaki trafiği izlerler. Gelecekte bu konaklar, merkezileştirilmiş bir gözlem istasyonuna otomatik olarak rapor göndererek sistem yöneticilerinin işlerini kolaylaştıracaklardır.

Bu yeni SSS modelinin altında yönetim istasyonu olarak bilinen bir kavram yatar. Sistem yöneticisi herhangi birinin bilgisayarındaki herhangi bir dosyadaki değişiklikten kolayca ve çabucak haberdar olabilir. Bu sistemde aynı zamanda, herhangi bir konaktaki konak tabanlı SSS yöneticiye güncel trafikle ilgili özet raporları ve grafikleri gönderebilmektedir. Konsolun analizi bu verilerin saldırı olup olmadığını ortaya çıkarabilir. Saldırı durumunda da yönetici konağın konfigürasyonunu gözden geçirip konfigürasyonda kolaylıkla değişiklikler yapabilecektir. Aynı değişiklik, ağ üzerindeki diğer özdeş konaklara da uygulanarak onların da korunması sağlanabilecektir.

Yukarıda anlatılan model uygulanabilirse SSS'lerin geleceği ümit verici görünmektedir. Konak tabanlı sistemler davranış tabanlı çalışmaya ihtiyaç duyarlar. Kötü niyetli hareketleri veya sıra dışı olayları bu şekilde yakalarlar. Ağ veya sistem üzerindeki kötü niyetli hareketler numune karşılaştırma yöntemi ile de bulunabilirler. Var olan SSS'ler bu şekilde çalışmaktadırlar. İhtiyaç olan şey sadece bu sistemi biraz daha geliştirmektir. Anormal durumları yakalamak kolaydır fakat anlamak zordur.

Bu sorunun çözümü istatistiksel analizin ve yapay zekanın altında yatar. Geleceğin SSS yapısında yönetim konsolu bir çok makineden anormal olay uyarıları alacak, bu veriler üzerindeki korelasyona ve ilişkilere yoğunlaşacaktır. Bunun için ender görülen olayların rapor edilmesine ihtiyaç olacaktır. Bunun yanı sıra yönetim konsolu ağ yapısı üzerinde bulunan güvenlik duvarları, ağ geçitleri, farklı SSS'ler ve yönlendiriciler gibi bazı parçalarla da iletişim içinde bulunacaktır. Gelecekte bir SSS protokolüne veya SSS rapor formatına da ihtiyaç olacaktır.SSS'lerin geleceği konusunda olasılıklar sonsuzdur.

6. BİR SSS YAZILIMININ UYGULANMASI VE DEĞERLENDİRİLMESİ

Bu bölüm, RealSecure adlı SSS yazılımının Eskişehir Fatih Sultan Mehmet İlköğretim Okulu'nun bilgisayar laboratuarına uygulanmasını ve değerlendirilmesini içermektedir. Bu bölümde öncelikle SSS yazılımının kurulacağı laboratuar ortamı tanıtılacak sonra SSS yazılımının kurulumu ile ilgili bilgiler verilecektir. Daha sonra yazılım, bölüm 5.6'da anlatılan verimlilik unsurlarını taşıması yönünden değerlendirilmek üzere bazı testlere tabi tutulacak ve uygulamaların sonuçları değerlendirilecektir.

6.1. Uygulama Ortamı

SSS yazılımının uygulanacağı ağ 10 Mbps hıza sahiptir. Bir adet NT 4.0 işletim sistemi, 19 adet Windows 98 işletim sistemi ve bir adet Linux işletim sistemi çalıştıran bilgisayar vardır.

Windows 98 ve Linux çalıştıran sistemlerin donanımları aşağıdaki gibidir;

- Pentium II 400 Mhz Mikroişlemci
- 32 MB RAM
- 4 GB Sabit Disk

NT İşletim Sistemini çalıştıran sistem ise şu donanıma sahiptir;

- Pentium III 600 Mhz Mikroişlemci
- 64 MB Ram
- 8 GB Sabit Disk

Bu yapı, üç farklı işletim sistemini bulundurması nedeni ile oldukça sağlıklı bir değerlendirme ortamı oluşmasını sağlamıştır.

6.2. SSS Yazılımının Kurulumu

Yazılımın kurulumu kolay ve hızlıca tamamlandı. Ağ trafiğini izleme özelliği sayesinde ağdaki diğer bilgisayarlara kurulmasına gerek olmadı. Bu nedenle sadece Windows NT işletim sistemini kullanan bilgisayara kuruldu.

Yazılım çeşitli varyasyonlarda önceden tanımlanmış bir güvenlik poliçesi sundu. Bu poliçelerden en yüksek düzeyde koruma sağlamaya yönelik olanı seçildi.

Yazılım yüksek, orta ve düşük olmak üzere üç farklı uyarı seviyesi sunmaktadır. Kullanıcı bu uyarıların ne şekilde iletileceğini kendisi belirleyebilmektedir.

6.3. Değerlendirme Uygulamaları

Bölüm 5.6'da da değinildiği gibi bir SSS yazılımının verimliliğini belirleyen en önemli unsurlar etkinlik, hata toleransı, performans ve hatasızlıktır. Bu nedenle yazılımın değerlendirmesi için kullanılacak uygulamalar bu unsurlar göz önünde bulundurularak seçilmiş ve ayrı başlıklar altında anlatılmıştır.

6.3.1. Etkinlik değerlendirmesi

Etkinlik değerlendirmesinin yapılması için dışarıdan ve içeriden yapılabilecek olan ve sıklıkla karşılaşılan saldırı örnekleri seçilerek bu saldırılar ağa uygulanmıştır. Yazılım bu saldırılar ile ilgili uyarı üretmesi halinde başarılı sayılmıştır. Yapılan uygulamalar ve yazılımın verdiği tepkiler aşağıdadır.

- Windows NT sistemine yönetici parolasını tahmin etme amaçlı olarak 5 girişim yapıldı. Yazılım başarılı oldu.
- Windows NT işletim sisteminin sistem saati ağ üzerindeki başka bir bilgisayar kullanılarak değiştirildi. Yazılım başarılı oldu.
- Kullanıcı güvenlik hesapları ile ilgili bir dosyaya normal kullanıcı olarak erişmeye çalışıldı. Yazılım başarısız oldu.
- Windows NT sisteminin 1'den 2048'e kadar olan portları tarandı. Yazılım bir uyarı üreterek port taraması yapıldığını belirtti. Başarılı oldu.
- Yarı gizli port tarama işlemi yapıldı. Yazılım başarılı oldu.
- Ağın haritalanmasına çalışıldı. Yazılım ağ haritalama saldırısının yapıldığını belirten bir uyarı üretti. Başarılı oldu.
- SMTP Portundan erişii sağlanan bir bilgisayara SMTP komutları kullanarak mail bırakılmaya çalışıldı. Yazılım başarısız oldu.

- NT İşletim Sisteminin ilk versiyonlarında bulunan bir sistem açığının suistimali yapılarak Sistem Yöneticisi yetkisinin alınması gerçekleştirilmeye çalışıldı. Yazılım başarısız oldu.
- Windows NT ağlarında bulunabilen ve Nuke olarak adlandırılan bir hatanın suistimal edilmesi ile yapılan saldırı gerçekleştirildi. Yazılım başarılı oldu.
- Windows 98 sistemi üzerinden Windows NT işletim sisteminin şifrelerinin bir yardımcı program ile alınması denendi. Yazılım başarılı oldu.

Böylece Real Secure SSS yazılımı yapılan on etkinlik testinden sekizinde başarılı sayıldı. Güvenlik tehditlerinin belli ağırlık ve önceliklerinin olmaması nedeni ile yapılan testin sonucu, ancak başarılı olunan test sayısının toplam test sayısına oranı ile ifade edilebilecektir. Bu anlamda eldeki verilerle yazılımın etkinlik açısından %80 oranında başarılı olduğu yorumu yapılabilir.

6.3.2. Hata toleransı değerlendirmesi

Yazılımın çalışır durumunu koruyabilmesinin değerlendirilmesi ile ilgili olan bu test, yazılımı çalıştıran bilgisayara bir servis dışı bırakma saldırısı düzenlenerek yapılmıştır. Bölüm 3'te de bahsedildiği gibi servis dışı bırakma saldırısı sistem kaynaklarını tüketecek kadar çok yüklemeye, böylece de görevini yapamayacak hale getirmeye zorlamakla yapılır. Bunu sağlamak için öncelikle sistem servis dışı bırakma saldırısından etkilenebilecek bir hale getirilmiş sonra da bu sistem açıkları suistimal edilerek dağıtık bir servis dışı bırakma saldırısıyla sistemin aşırı yüklenmesi sağlanmıştır. Bu şartlar altında ağa SSS yazılımının normalde yakalayabileceği saldırılar düzenlenmiştir.

Aşırı yüklenen bilgisayar üzerinde çalışan yazılım bazı saldırıları fark edebildiği gibi, yakalayıp analiz edemediği paketler için de durumu belirten uyarılar üretmiştir. Servis dışı bırakma saldırısı durdurulduğunda herhangi bir takılma veya kilitlenme yaşanmamış, program çalışmasına devam etmiştir. Böyle bir ortamda dahi çalışmasına devam etmesi ve kaçırdığı paketler için uyarı

üreterek gözden kaçırılabilecek noktaların olabileceğine ilişkin bilgi vermesi, yazılımın hata toleransına sahip olduğunun göstergesidir denilebilir.

6.3.3. Performans değerlendirmesi

Performans testi için ağ çok yoğun veri trafiğine maruz bırakılmıştır ve yazılımın yoğun bir iş yüküne girmesi sağlanmıştır. Bu şartlarda kullandığı işlemci zamanı miktarına bakılmıştır. 10Mbit/s hızındaki ağ neredeyse tamamen doldurulmasına rağmen işlemcinin %70 seviyelerinde kullanıldığı görülmüştür. Yaklaşık 1200 tekil imzaya sahip olan bir yazılım olduğu ve üzerinde çalıştığı donanım göz önünde bulundurulursa bunun başarılı bir sonuç olduğu kanısına varılacaktır.

6.3.4. Hatasızlık değerlendirmesi

Real Secure yazılımı çeşitli filtreler kullanarak hatalı rapor üretimini azaltabilmektedir. Yazılımın kullandığı filtreler kullanıcı tarafından konfigüre edilebilmektedirler. Bu nedenle hatasızlık oranı büyük ölçüde kullanıcıya bağlıdır denilebilir. Hatasızlık ile ilgili karara ulaşmak için zamana ihtiyaç vardır. Yazılımın uygulamada kaldığı süre içinde oluşturduğu yanlış pozitif ve yanlış negatiflere bakılmalıdır. Biraz da uygulandığı ağın yapısı ve kullanım amacı izin vermediğinden, yazılım kullanıldığı üç hafta boyunca herhangi bir yanlış pozitif üretmemiştir. Ürettiği yanlış negatifler ise etkinlik değerlendirilmesinde ortaya çıkan iki durumla sınırlıdır.

Bu bilgilerle yazılımın hatasızlık konusundaki başarısıyla ilgili bir karara varmak zordur. Zaten filtre ayarlarının değiştirilebilmesine imkan tanımasıyla hatasızlık konusundaki başarısını büyük ölçüde kullanıcıya bırakmaktadır.

7. SONUÇ

Bilgi güvenliği kavramının önemi, bilgi ve iletişim teknolojilerinin gelişmesine ve yaygınlaşmasına paralel olarak hızla artmaktadır. Özellikle hayatın her alanında kullanılan İnternet, güçlü bir iletişim ortamı sağlamakla beraber bilgi güvenliğinin de tehlikede olması sonucunu doğurmuştur. Bu sonuç, kişi veya kurumları bilgi güvenliğini sağlamak için daha fazla kaynak ayırmaya zorlamıştır.

Alt yapı eksiklikleri, kullanılan yazılımların/donanımların hataları veya kullanıcıların eğitimsizliği gibi sebeplerden ötürü İnternet suistimal edilebilecek pek çok açığa sahiptir. Bu açıklar İnternet ve bilgisayar teknolojilerine hakim olan kişiler tarafından suistimal edilerek zarar verme amaçlı olarak kullanılabilmekte ve bilgi güvenliğinin tehdit altında kalmasına neden olabilmektedir. Bu kişiler tarafından pek çok saldırı yöntemi bulunarak zarar verme amaçlı olarak kullanılmıştır ve kullanılmaya devam edilmektedir.

Bilgi güvenliğini tehdit eden unsurların ortaya çıkması, bu tehditlere karşı önlemlerin oluşturulmasını sağlamıştır. Bu anlamda veri iletişiminin istendik bir şekilde kısıtlanmasını sağlayan güvenlik duvarları, iletilen verilerin bütünlüğünü ve gizliliğini sağlayan şifreleme algoritmaları, zararlı programların bulunmasını ve yok edilmesini sağlayan anti-virüs programları gibi teknolojilerin geliştirilmesinin yanı sıra İnternet altyapısındaki güvenlik eksikliklerini ortadan kaldırmayı hedefleyen yeni bir İnternet protokolü geliştirilmiştir.

Güvenliğin tehdit edilmesine karşı alınan önlemler ve geliştirilen teknolojiler çoğu zaman kalıcı bir çözüm sağlamamışlardır. Saldırganlar çoğu zaman oluşturulan güvenlik uygulamalarını aşacak yeni yöntemler bulmuşlardır.

Her geçen gün güvenlik teknolojileri anlamında yeni bir ürün ve yaklaşım ortaya çıkmaktadır. Bunlardan en çok umut vaat edeni saldırı sezme sistemleri olarak görülmektedir. Bu yazılım veya donanımlar sistem yöneticisinin sisteme yapılan saldırılardan haberdar edilmesini sağlamayı amaçlamaktadırlar.

Günümüzdeki SSS'lerin bir temsilcisi olan ve bu tezin 6. bölümünde incelenen Real Secure adlı yazılım, var olan SSS'lerin bazı eksikleri halen

taşımakla beraber bulundukları ağa güvenlik anlamında büyük yararlar sağladıklarını göstermiştir. Zaten SSS'ler çalışma yapılarından dolayı bilgi güvenliğini sağlamaya yönelik olarak büyük bir potansiyele sahip oldukları kanısını bırakmaktadırlar. Gelecekte bilgisayar donanımının gelişmesi, yapay sinir ağları, genetik algoritmalar ve yapay zeka gibi alanlardaki ilerlemelerin SSS'lere uygulanması ile çok daha başarılı sonuçlara ulaşılabilirler. İstatistik bilimi ise şimdi olduğu gibi gelecekte de SSS'ler için büyük öneme sahip olacaktır. Mimari olarak geleceğin saldırı sezme sistemlerinde bir analiz ve yönetim merkezi bulunacağı ve buna bağlı ajan yazılımların ağ üzerine dağıtılmış bir şekilde güvenlikle ilgili veri toplayıp bu verilerin raporlaştırılarak merkeze iletilmesini sağlayacağı düşünülmektedir. Merkezi birim ise bu raporların karşılaştırmalı analizinden ve bu raporlarla ilgili bir karara ulaşılmasından sorumlu olacaktır.

SSS'ler için, hem güvenlik duvarlarının hem de antivirüs yazılımlarının avantajlarına sahip olmaları ile geleceğin en önemli güvenlik teknolojilerinden birisi olabileceği söylenilebilir.

KAYNAKLAR

[1] DEREGÖZÜ, R., *Bilgisayar ağlarında güvenlik sorunu "Firewall" kullanarak ağ güvenliğini sağlama*, İstanbul Teknik Üniversitesi, İstanbul, (1999).

[2] DAĞ, B., *Ağ güvenliği ve güvenlik duvarları*, Kocaeli Üniversitesi, Kocaeli, (2001).

[3] KARAAHMETOĞLU, O., *İnternet güvenliği kavramları ve teknolojileri*, İstanbul Teknik Üniversitesi, İstanbul, (2001).

[4] CHAMPMAN D. B. ve ZWICKY E. D. *Building internet firewalls*, Oreilly Media, USA, (1995).

[5] http://www.packetwatch.net/documents/papers/snifferdetection.pdf, (12.03.2005).

[6] http://www.webopedia.com/TERM/I/IP_spoofing.html, (12.03.2005).

[7] http://www.linuxfocus.org/Turkce/March2003/article282.shtml, (18.04.2005).

[8] http://www.webopedia.com/TERM/I/IP_spoofing.html, (20.04.2005).

[9] http://www.cert.org/advisories/CA-1995-01.html, (12.04.2005).

[10] http://www.all.net/journal/netsec/1995-09.html, (07.04.2005).

[11] http://www.linuxfocus.org/Turkce/March2003/article282.shtml#282lfindex13, (07.04.2005).

[12] http://www.surasoft.com/articles/ddosa.php, (08.04.2005).

[13] http://www.olympos.org/index.php/article/articleview/128/1/2/?PrintableVersion=enabled, (16.04.2005).

[14] http://www.itmweb.com/essay534.htm#Source%20Routing%20Attack, (18.04.2005).

[15] http://www.cs.columbia.edu/~smb/papers/ipext.pdf, (18.04.2005).

[16] http://support.microsoft.com/kb/129972, (19.04.2005).

[17] http://www4.gantep.edu.tr/~op20476/virus.HTM, (21.04.2005).

[18] SAKA, Y., *Bilgisayar ağ güvenliği ve şifreleme*, Muğla Üniversitesi, Muğla, (2000).

[19] http://www.extension.iastate.edu/Publications/PM1789J.pdf, (21.04.2005).

[20] http://www.securitydocs.com/library/2742, (21.04.2005).

[21] http://www.cs.wright.edu/~pmateti/Courses/499/Viruses, (23.04.2005).

[22] BANGER, G., *Bilgisayar virüsleri*, Bilim Teknik Yayınevi, Eskişehir, (1991).

[23] http://its.atilim.edu.tr/guvenlik.htm, (24.04.2005)

[24] BAHTİYAR, Z, *Virüsler ve Güvenlik, Pusula Yayıncılık*, İstanbul, (2003).

[25] http://www.net-pa.com.tr/sss.html#web, (26.04.2005).

[26] CHESWICK W. R. ve ark., Firewalls And Internet Security, USA, (2000).

[27] http://www.pcstats.com/articleview.cfm?articleid=1450&page=4, (01.05.2005).

[28] http://www.pcstats.com/articleview.cfm?articleid=1450&page=5, (01.05.2005).

[29] http://www.cs.itu.edu.tr/~orencik/VekilSunucularveGuvenlikDuvarlari.doc, (02.05.2005).

[30] http://www.tatamcgrawhill.com/digital_solutions/kahate/Chapter%209.pdf, (02.05.2005).

[31] http://www.ce.itu.edu.tr/lisansustu/dersler/blg510/2003/sunumlar/AtesDuvariveVekilSunucular.doc, (05.05.2005).

[32] http://ab.org.tr/ab03/sunum/90.doc, (07.05.2005).

[33] http://ietfreport.isoc.org/idref/draft-alten-snmpv2-sec-encap, (04.05.2005).

[34] http://sertifika.bilten.tubitak.gov.tr/net/teknik/kriptografi.jsp, (09.05.2005).

[35] http://neworder.box.sk/newsread.php?newsid=9257, (10.05.2005).

[36] DIFFIE W. ve HELLMAN M. E., *New directions in cryptography*, IEEE Computer Society, USA, (1976).

[37] http://edergi.linux.org.tr/?~p=dergi&!m=1&action=show&which=76, (10.05.2005).

[38] http://csrc.nist.gov/publications/nistpubs/800-31/sp800-31.pdf, (11.05.2005).

[39] http://perso.rd.francetelecom.fr/debar/papers/DebDacWes99.pdf, (13.05.2005).

[40] http://www.securityfocus.com/infocus/1214, (10.05.2005).

[41] http://www.netsec.org.sa/int_det.htm#What%20is%20an%20Intrusion?, (11.05.2005).

[42] www.primode.com/glossary.html, (12.05.2005).

[43] www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids7/unix_cfg/glos s.htm, (16.05.2005).

[44] http://csrc.nist.gov/publications/nistbul/itl99-11.txt, (18.05.2005).

[45] http://perso.rd.francetelecom.fr/debar/papers/DebDacWes99.pdf, (20.05.2005).

[46] www.forum-intrusion.com/archive/ Intrusion%20Detection%20Techniques%20and%20Approaches.htm, (24.05.2005).

[47] www.rennes.enst-bretagne.fr/ ~ybouzida/TechnicalReports/TechReport2002.pdf, (23.05.2005).

[48] http://www.sec-1.com/intrusion_detection_systems.html, (23.05.2005).

[49] http://www.securityfocus.com/infocus/1514, (24.05.2005).

[50] http://www.ciscopress.com/articles/article.asp?p=25334&seqNum=3&rl=1, (26.05.2005).

[51] http://www.windowsecurity.com/articles/IDS-Part2-Classification-methods-techniques, (02.06.2005).

[52] http://www.cse.buffalo.edu/~sbraynov/seminar%202004/papers/zamboni-agents.pdf, (02.06.2005).

[53] http://www.cs.nps.navy.mil/people/faculty/rowe/ingramthesis.htm, (04.06.2005).

[54] www.windowsecurity.com, (08.06.2005).