Gazi Üniv. Müh. Mim. Fak. Der.
Cilt 29, No 4, 707-715, 2014

Journal of the Faculty of Engineering and Architecture of Gazi University
Vol 29, No 4, 707-715, 2014

YENİ BİR AĞ GÜVENLİĞİ YAKLAŞIMI: DİNAMİK ZEKİ GÜVENLİK DUVARI MİMARİSİ

O. Ayhan ERDEM*, Ramazan KOCAOĞLU*
* Bilgisayar Mühendisliği Bölümü, Teknoloji Fakültesi, Gazi Üniversitesi, Beşevler, Ankara
ayerdem@gazi.edu.tr, ramazankocaoglu@gazi.edu.tr

(Geliş/Received: 04.11.2013; Kabul/Accepted: 16.10.2014)

ÖZET

Bu makalede, geleneksel güvenlik duvarı mimarilerinden tamamen farklı, yeni bir güvenlik duvarı mimarisi geliştirilmiştir. Geliştirilen mimari DIFA (Dynamic Intelligent Firewall Architect) olarak adlandırılmıştır. DIFA kendi kendisini yönetebilme temeline dayanan bir güvenlik duvarı mimarisidir. DIFA, üzerinden geçen trafiğin analizini yaparak ve yerel alan ağını tarayarak erişim kurallarını kendisi oluşturmaktadır. Koruduğu ağda yapısal bir değişiklik oluştuğunda bunu tespit ederek gerekli yapılandırmaları kendisi yapabilmektedir. Ağ yöneticisine sadece kontrol amaçlı ihtiyaç duymaktadır. DIFA'nın verimliliği gerçek ağ ortamları kullanılarak test edilmiştir. Elde edilen sonuçlar, DIFA'nın kural oluşturma işlemini başarılı bir şekilde yapabildiğini göstermiştir.

Anahtar Kelimeler: DIFA, ağ güvenliği, güvenlik duvarı, erişim kontrol kuralları

A NEW APPROACH FOR NETWORK SECURITY: DYNAMIC INTELLIGENT FIREWALL ARCHITECTURE

ABSTRACT

A new firewall architecture which is different from traditional firewalls has been developed in this article. The new firewall approach has been called as DIFA (Dynamic Intelligent Firewall Architecture). DIFA is based on self-configuration. It scans local area network and analyses incoming internet traffic so that it can create access rules. DIFA is able to detect topology changes in LAN and able to revise access rules and configures itself. It requires a network administrator for approval of the process. Efficiency of DIFA has been tested using real and various network environments. Obtained results show that DIFA achieved successfully creation of access lists.

Keywords: DIFA, network security, firewall, access control list

1. GİRİŞ (INTRODUCTION)

Güvenlik duvarları ağ ortamında iletişimi sağlayan trafiğin istenildiği şekilde filtrelenmesine olanak tanıyan cihazlardır. Bir güvenlik duvarı, paketlerin içeriklerine bakarak uygun olmayan paketleri düşürür veya güvenliğini sağladığı yerel ağ içerisine girmesine izin verir. Belirtilen uygunluk kontrolünü sağlayan ise güvenlik duvarları üzerindeki erişim kurallarıdır. Bu kurallar ağ yöneticileri tarafından oluşturulmaktadır. Erişim kurallarının oluşturulması karmaşık ve zor bir süreçtir. Erişim kuralları ihtiyaca göre sürekli ağ yöneticileri tarafından düzenlemektedir. Düzenleme işlemlerin dikkatsizce yapılması, gereksiz olan TCP/IP portlarının erişime açılmasına neden olmaktadır. Bu durumda güvenlik açığı meydana gelmektedir.

Kural tabanlı çalışan güvenlik duvarlarının temel bileşeni kural bütünlüğüdür. Üzerlerindeki kurallara göre paketleri düşürme veya izin verme gibi işlemler gerçekleştirilir. Kural bütünlüğünü sağlamak için bazı modeller geliştirilmiştir [1]. Bununla birlikte, ağ üzerinde tarama ve izleme yaparak ağdaki bağlantı sorunlarının tespiti için yapılan araştırmalar mevcuttur. Ağın belli bir süre görüntülenmesi sonucunda elde edilen veriler sorunun çözümünde kullanılabilir [2]. Mevcut bir güvenlik duvarı, kendisine gelen bir paketin içerisindeki başlık bilgilerine bakarak paketin geçirilmesine ya da düşürülmesine karar vermektedir. Paketlerde başlık

bilgilerine bakılmadan akan trafiğin tanımlanmasının mümkün olduğu gösterilmiştir [3]. Bu sonuç, gelecek nesil güvenlik duvarı mimarilerinde göz önünde bulundurulması gereken bir durumdur. Farklı güvenlik duvarı mimarilerinin ortaya çıkmasına neden olabilir. Büyük bir ağın altyapısı farklı ve karmaşık cihazlardan oluşur. Bu altyapı model alınarak ağın büyütülmesi veya başka ortamlarda kullanılması oldukça zor ve zahmetli bir iştir. Aynı şekilde, bir güvenlik duvarındaki kurallar model alınarak farklı ağ ortamları için yapılandırılması zor ve sıkıntılı bir iştir [4]. Bu duruma kendiliğinden uyum sağlayabilen, farklı ağ ortamlarını otonom olarak birleştirilebilen tasarımlara ihtiyaç duyulmaktadır.

Günümüzdeki güvenlik duvarı mimarilerinin gözden geçirilmesi gerekmektedir. Ipv6 adresleme yapısının yaygınlaşacak olması bu gerekliliği hızlandıracaktır. Bununla beraber, mevcut güvenlik duvarlarının yönetimi özel bilgi gerektirmekte ve yönetim sırasında hata yapılma olasılığı oldukça fazladır. Geleneksel güvenlik duvarlarının aksine, günümüz teknolojisine uygun bir güvenlik duvarı kısmen kendi kendini yönetebilmeli, duruma göre yapısında değişikliğe gidebilmelidir. Durağan bir yapıdan çıkıp daha dinamik bir mimariye sahip olması gerekmektedir. Üzerinden geçen trafiğin yönüne ve trafiğin çeşidine göre kendisini yenileyebilmelidir. Güvenlik duvarını yöneten kişi bu süreçte sadece oluşumu kontrol eden kişi olarak görev almalıdır. Literatürde bir güvenlik duvarındaki kuralların yönetimi için birçok yaklaşım bulunmaktadır. Bu makalede, yeni bir güvenlik duvarı mimarisi yaklaşımı sunulmuştur ve DIFA (Dynamic Intelligent Firewall Architecture) olarak adlandırılmıştır. Sunulan yaklaşım, erişim kontrol listelerinin daha kolay yönetilebilmesi için geliştirilen ya da listeler içerisindeki kural çakışmalarını bulan bir araç değildir. Tamamen farklı bir bakış açısı ile tasarlanan, yeni bir güvenlik duvarı mimarisidir.

2. İLGİLİ ÇALIŞMALAR (RELATED WORKS)

Günümüz güvenlik duvarları, üzerinden geçen trafiğin durumu ile ilgili karar verirken hâlihazırdaki mevcut kuralları sırası ile kontrol etmektedir. Bir güvenlik duvarı üzerinden akan trafik için ilk uyan kurala göre işlem yapmaktadır. Gouda ve Liu mevcut güvenlik duvarlarındaki filtreleme yapısından dolayı karşılaşılan 3 tane problem tanımlamışlardır. Bu problemlerinin çözümüne aday olabilecek yeni bir güvenlik duvarı tasarımı sunmuşlardır [5]. Chao ve Yang, güvenlik duvarlarının yönetimi için yazılan kuralların birbirleri ile olan ilişkilerini incelemişlerdir. Bu kuralların birbirlerinin çalışabilirliğine etki ettiklerini göstererek, oluşturdukları sorunları 3 ana başlık altında toplamışlardır. Bu sorunlara çözüm olabilecek bir sistem tasarlamışlardır [6]. Liu yaptığı çalışmada, güvenlik duvarlarındaki kural dizilimlerindeki yanlışlıkları bulmak için bir araç geliştirmiştir. Bu araç sayesinde, güvenlik duvarındaki bir kuralın yeterli olup olmadığını anlaşılabilmektedir [7].

Güvenlik duvarlarının işletim sistemi tarafında çalışan düşük seviyeli kural yazım dili farklılık göstermektedir. Bu düşük seviyeli dilleri anlamak ve öğrenmek zaman alıcı ve zor bir iştir. Pozo ve arkadaşları yaptıkları çalışmada, yüksek seviyeli kural yazma ortamı sunan, oluşturulan kuralları otomatik olarak üzerinde çalıştığı platforma göre düşük seviyeli dile çeviren bir araç (MBD) geliştirmişlerdir [8]. Platformdan bağımsız basit ve kolay bir erişim kontrol listesi yazma modeli otomatik olarak bir dizi işlemler sonrasında istenilen güvenlik duvarı platformuna uygun şekilde çevrilebilir. CONFIDDENT şu anda bu otomatik erişim kontrol listesi çevirme işlemi için önde gelen bazı platformları desteklemektedir [9]. Sereelaja ve Pai, karınca kolonisi optimizasyon yaklaşımını kullanarak geliştirdikleri yeni bir sistem (ACO-PF) ile filtreleme işlemini yapmışlardır [10]. Ağdaki tüm güvenlik ürünlerinin doğru ve etkili şekilde yönetimi zordur. Bu yüzden bir ağdaki tüm güvenlik ürünlerinin merkezi olarak yönetilmesi daha etkin bir çözüm olacaktır [11]. Güvenlik duvarının üzerindeki erişim kontrol listesi çok iyi bir şekilde düzenlenmelidir. Bu optimizasyonu sağlamak amacıyla bir araç geliştirilmiştir [12]. Büyük ağlarda yapılan değişiklikler sonucunda bir süre sonra yapı düzensiz ve karmaşık bir hal alır. Yapılandırma bozulmadan gereksiz olan ayarlamalar silinerek yapı daha sade hale getirilebilir [13].

Liao ve arkadaşları ENAVis isimli bir araç geliştirmişlerdir. Bu araç sayesinde kurumsal bir ağ içerisinde olup biten her şeyi görsel olarak ağ yöneticisine sunabilmişlerdir. Ağ içerisinde meydana gelen güvenlik problemlerinin kaynağını daha az zaman ve iş gücü harcayarak tespit etmişlerdir [14]. İnternet kullanımı ve güvenlik tehditlerinin artmasına paralel olarak, sonraki nesil güvenlik duvarlarından beklenen yeterlilikler artmaktadır. Bundan dolayı son zamanlarda bilgisayar bilimleri ve ağ güvenliği alanında yapılan çalışmaların çoğunluğu saldırı önleme sistemleri (IPS) üzerinedir. [15-18]. Bilgi teknolojileri altyapısının hızlıca büyümesi sonucu bulut bilişim, sanallaştırma teknolojisinin doğal bir uzantısı haline gelmiştir. Bulut bilişimin etkin şekilde kullanılması güvenliğin sağlanabilmesine bağlıdır. Bulut içerisindeki sanal sunucuların ve sanal ağların güvenliğinin sağlanabilmesi üzerinde çalışılan bir konudur [19]. Buluttaki kaynaklar ve servisler çeşitli saldırılara maruz kalabilir ve bu saldırılardan etkilenebilir. Bulut içerisinde kullanılabilen saldırı tespit ve önleme sistemlerinin verimliliği incelenmiştir. Geleneksel saldırı tespit ve önleme sistemlerinin bulut bilişimde kullanılması verimsizdir. Bulut bilişim yapısına uygun saldırı tespit ve önleme sistemleri için araştırmalar yapılmaktadır [20-21].

Tablo 1. Mevcut çalışmaların temel özelliklerinin karşılaştırılması (Comparison for attributes of existing study in literature)

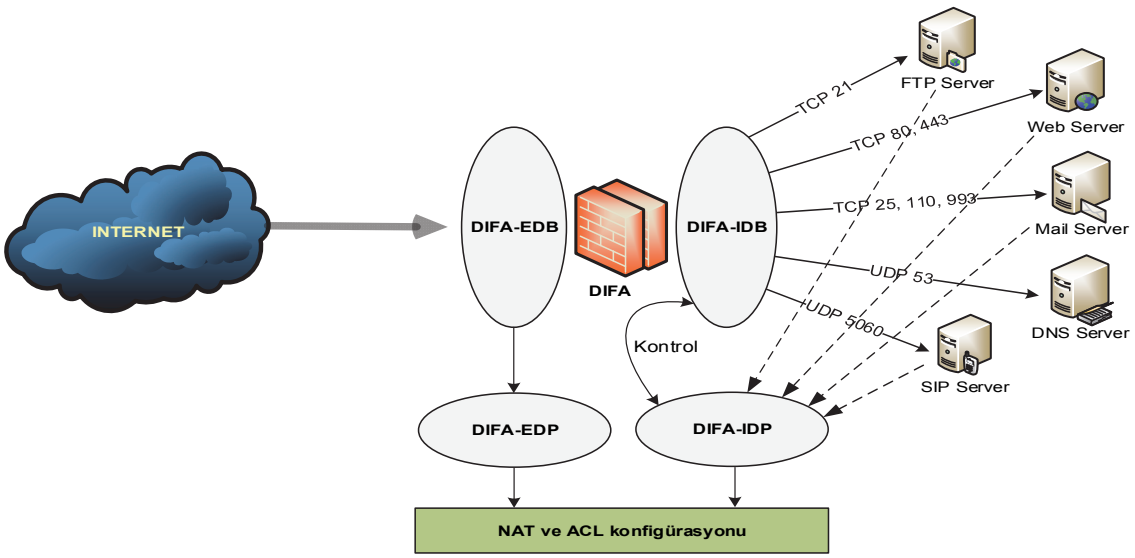
Değerlendirme Kriterleri	CONFIDDENT	MBD	ACO-PF	ENAVis	Three-tried Visual
Kural Uyumsuzluk tespiti	×	✓	✓	×	×
Kural Fazlalık tespiti	✓	×	✓	×	✓
Kural Çakışma tespiti	✓	✓	✓	✓	✓
Zone desteği	×	✓	×	✓	×
Yönetim biçimi	Görsel	Komut	Komut	Görsel	Görsel
Arayüz karışıklığı	Orta	Orta	Orta	Düşük	Yüksek
Platform bağımsızlığı	✓	✓	×	✓	×
Tasarım Modeli	model tabanlı	kural tabanlı	sezgisel tabanlı	kural tabanlı	kural tabanlı
Topoloji takibi	×	×	×	×	✓
VPN kural kontrolü	✓	×	✓	✓	×

Literatürde öne çıkan bazı çalışmaların erişim kontrol listesi oluşturma ve düzenleme biçimleri dikkate alınarak temel özelliklerinin karşılaştırılması Tablo 1'de gösterilmiştir.

3. GELİŞTİRİLEN GÜVENLİK DUVARI TASARIMI (DEVELOPED NETWORK SECURITY DESIGN)

Kurumsal ağlarda sadece intranet ortamına hizmet veren sunucular olduğu gibi, internet ortamında hizmete açılması gereken birçok sunucu da bulunmaktadır. Bu sunucuların internet üzerinden erişilebilir olması güvenlik duvarı üzerindeki NAT (Network Address Translation-Ağ Adresi Dönüştürme) ve ACL (Access Control List-Ağ Erişim Kontrol Listesi) yapılandırmasına bağlıdır. Sunucu sayısı ile bu yapılandırma arasında bir paralellik söz konusudur. Yani, sunucu sayısının artması gerekli olan NAT ve ACL sayısını artırır. Dolayısı ile yapılandırma daha karmaşık ve zor bir hal alır. Özellikle bir hata olduğunda bu hatanın tespit edilmesi için harcanan zaman ve bilgi ihtiyacı oldukça artmaktadır. DIFA bu gereksinimleri en aza indirmek için geliştirilmiş bir yaklaşımdır. Gerekli olan NAT ve ACL yapılandırması DIFA tarafından otomatik olarak belirlenir. Şekil 1'de DIFA'nın genel mimarisi gösterilmiştir. DIFA, yapısal olarak 4 bölümden oluşmaktadır. DIFA-EDB (Dynamic Intelligent Firewall Architect -External Data Buffer) ve DIFA-EDP (Dynamic Intelligent Firewall Architect - External Data Pool) olarak adlandırılan bölümler güvensiz olduğu varsayılan internet tarafı ile ilgili işlemlerden sorumlu iken, DIFA-IDB (Dynamic Intelligent Firewall Architect - Internal Data Buffer) ve DIFA-IDP (Dynamic Intelligent Firewall Architect -Internal Data Pool) olarak adlandırılan bölümler ise güvenli olduğu varsayılan LAN tarafı ile ilgili işlemlerden sorumludur.

DIFA durum denetlemeli (stateful) güvenlik duvarı sınıfı içerisine girmektedir. Bu yüzden, dinamik paket filitreleme olarak da adlandırılan stateful inspection özelliğine sahiptir. Statik paket filtreleme, bağlantı filtreleme gibi özellikleri desteklemektedir. NAT desteği sunmaktadır ve NAT'ın çalışması için gerekli olan kuralları oluşturabilmektedir. DIFA, mevcut bulunan güvenlik duvarı ile uyumlu olarak çalışabilir ve onların üzerindeki kuralları yönetme işlemini



Şekil 1. DIFA'nın genel mimarisi (General Architect of DIFA)

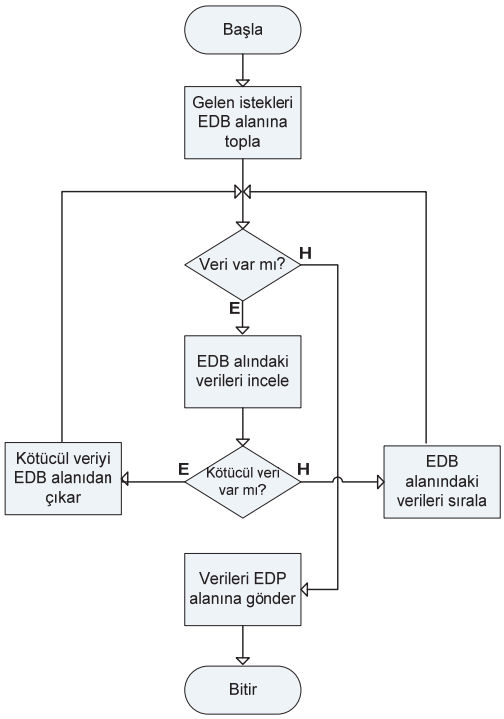
gerçekleştirebilir. DIFA mevcut hali ile uygulama katmanında filtreleme yapabilme özelliğine sahip değildir. İletim katmanına kadar paketleri inceleyip karar verebilmektedir.

3.1. DIFA-EDB (Dynamic Intelligent Firewall Architect – External Data Buffer)

DIFA-EDB, internetten mevcut güvenlik duvarına doğru gelen trafiğin analizini yapmak için toplanan bir bellek alanı olarak ifade edilebilir. İnternetten gelen trafiğin ilk olarak karşılandığı kısımdır. Mevcut güvenlik duvarına gelen tüm trafiğin "mirroring" işlemi uygulanarak DIFA-EDB alanına toplanması sağlanır. DIFA-EDB içerisinde bulunan paketler kaynak IP adresi, hedef IP adresi, hedef port numarası gibi alanlara göre incelenir. İnceleme sonucunda anormal bir paket yapısı bulunursa bu paketler düşürülür. Geriye kalan paketler hedef port numarasına göre sınıflandırılmaktadır.

3.2. DIFA-EDP (Dynamic Intelligent Firewall Architect – External Data Pool)

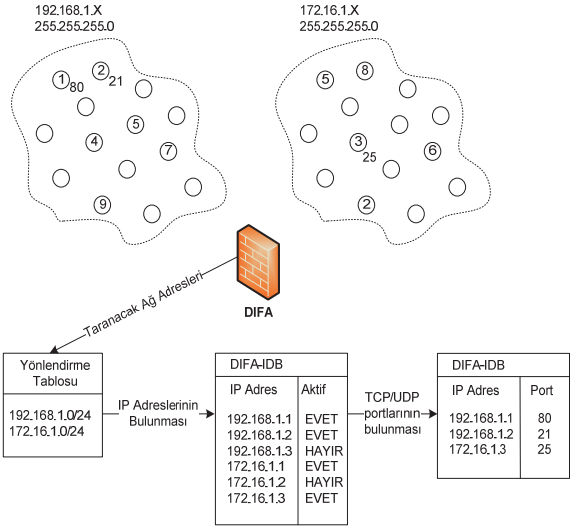
DIFA-EDP, port numarasına göre sınıflandırılmış ve sıralanmış şekilde verilerin bulunduğu bir alan olarak tanımlanmaktadır. DIFA-EDB alanında bulunan veriler sıralandıktan sonra bu alana gönderilir. DIFA-EDB ve DIFA-EDP arasındaki ilişkinin akış şeması Şekil 2'de gösterilmiştir. Bu mimariye göre, bir güvenlik duvarına internet gibi bir dış ortamdan gelen trafik ilk olarak DIFA-EDB alanında toplanır. Bu alandaki veriler bitene kadar tcp/udp port numarasına göre sınıflandırılır.



Şekil 2. DIFA-EDB ve DIFA-EDP'nin akış şeması (Flow Control of DIFA-EDB and DIFA-EDP)

3.3. DIFA-IDB (Dynamic Intelligent Firewall Architect – Internal Data Buffer)

Yerel alan ağında bulunan sunuculardan hangilerinin internet ortamında hizmet vereceğinin DIFA tarafından bulunması gerekmektedir. Yani, dışarıdan içeriye hangi portların hangi sunuculara doğru yönlendirileceği bilinmelidir. Bunu yapabilmek için sunucuların IP adresleri DIFA tarafından öğrenilmelidir. DIFA, kendisine bağlı ağ adreslerini yönlendirme tablosuna bakarak öğrenir. Öğrendiği ağ adreslerini tarayarak sunucuların IP adreslerini ve hangi porttan hizmet verdiklerini tespit etmektedir. Yerel alan ağındaki tüm IP blokları taranması sonucu elde edilen veriler DIFA-IDB alanında saklanır. Özet olarak DIFA-IDB, güvenlik duvarının koruduğu ağı tarayarak elde ettiği verilerin toplandığı alan olarak tanımlanabilir. Şekil 3'de DIFA-IDB yapısının işleyişi gösterilmiştir.



Şekil 3. DIFA-IDB ve DIFA-IDP'nin oluşturulması (Creating process of DIFA-IDB and DIFA-IDP)

3.4. DIFA-IDP (Dynamic Intelligent Firewall Architect – Internal Data Pool)

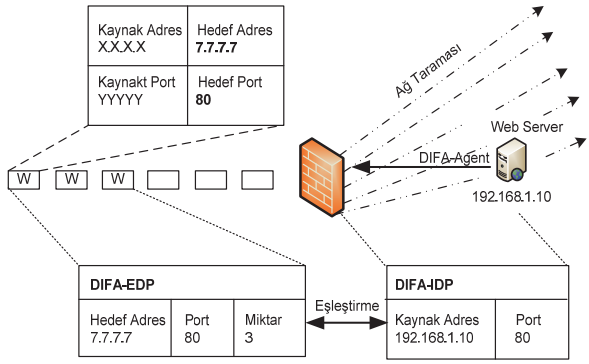
DIFA-IDB alanında LAN'da bulunan tüm cihazların IP adresi, kullanımda olan tcp/udp port bilgileri bulunmaktadır. Fakat bu bilgiler içinde NAT ve ACL yazılmasına gerek olmayan bilgiler çoğunluktadır. Hatta gereksiz yere kural yazılması güvenlik açığına neden olur. Bu yüzden DIFA-IDB alanında bulunan her kayıt için kural oluşturulmamalıdır. Örneğin bir web sunucusu TCP-80 portu üzerinden hizmet vermektedir. Bu nedenle sadece bu port için kural yazılması gerekmektedir. Ancak herhangi bir işletim sistemi varsayılan olarak birçok port üzerinden sürekli dinleme durumundadır. Örneğin, her Windows işletim sisteminde tcp 135 (rpc) portu varsayılan olarak dinleme konumunda bulunmaktadır.

İnternet ortamından erişime açılması istenilen sunuculara DIFA-Agent olarak adlandırılan ufak bir

aracı yazılım yüklenmektedir. DIFA-Agent yüklü olduğu sunucu üzerinde çalışan servisleri içeren bilgileri DIFA'ya göndermektedir. DIFA bu bilgilere bakarak ilgili sunucunun web sunucusu ya da mail sunucusu olduğunu anlamaktadır. Böylece internete açılacak tüm sunucuların hangi hizmeti verdiği DIFA tarafından bilinmektedir. DIFA sürekli olarak DIFA-Agent'lardan gelecek bilgileri bekleme konumundadır. DIFA-Agent ile DIFA arasında bir paylaşımlı anahtar kullanılarak bilgilerin güvenli şekilde iletilmesi sağlanmaktadır. DIFA-Agent'lardan gelen bilgiler DIFA-IDP alanında toplanmaktadır. Ayrıca, DIFA-IDP alanındaki verilerin DIFA-IDB alanında da olup olmadığı kontrol edilmektedir. Bu sayede yerel ağda gerçekten böyle bir cihazın olup olmadığı garanti altına alınmış olunur. Yerel alan ağı içerisindeki sunuculara yerel ağ üzerinden erişim kurallarının oluşturulması sırasında DIFA-IDP kullanılmaktadır.

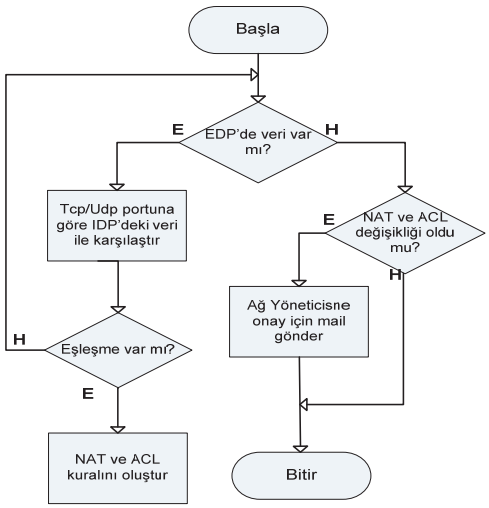
3.5. DIFA-EDP ve DIFA-IDP Entegrasyonu
(Integration between DIFA-EDP and DIFA-IDP)

Kuralların oluşturulabilmesi için, DIFA-EDP alanı içerisindeki veriler ile DIFA-IDP alanı içerisindeki kayıtlar eşleştirilir. Eşleştirme sonucu dışarıdan hangi porta gelen isteğin içeride hangi sunucuya yönlendirileceği belirlenmiş olur. Eşleştirilmiş kayıtlar güvenlik duvarına uygulanmadan önce kontrol amaçlı ağ yöneticisine mail aracılığı ile gönderilir ve ağ yöneticisinin onay beklenir. Ağ yöneticisinin onay vermesi sonucunda dışardan içeriye yönlendirme işlemi DIFA tarafından uygulanır. Bu yapının işleyişi Şekil 4'de gösterilmiştir.



Şekil 4. DIFA-EDP ve DIFA-IDP'nin oluşturulması (Working of DIFA-EDP and DIFA-IDP)

Şekil 4'de internet tarafından gelen tcp 80 nolu port istekleri *"w"* olarak ifade edilmiştir. Her paketin doğası gereği kaynak IP adres, hedef IP adres, kaynak port ve hedef port kısımları bulunmaktadır. DIFA, gelen paketlerin hepsini DIFA-EDB alanına iletir. Paketleri hedef IP ve port numarasına göre sınıflandırarak DIFA-EDP alanını oluşturur. Şekil 4'de görüldüğü üzere DIFA-EDP içerisinde 7.7.7.7 hedef IP adresine ve 80 numaralı hedef portuna sahip 3 adet paket geldiği bilgisi bulunmaktadır. Bu paketlerin toplam sayısının belirlenen bir eşik değerine ulaşması gerekmektedir. DIFA ayrıca yerel alan ağını tarayarak 192.168.1.10 IP adresine sahip 80 numaralı port üzerinden hizmet veren bir sunucunun olduğunu keşfetmiştir. Ayrıca, DIFA-agent aracılığı ile sunucunun üzerinde apache ya da IIS gibi bir web servisinin çalıştığı bilgisine sahiptir. Bu bilgi ise DIFA-IDP alanı içerisinde tutulmaktadır. Sonuç olarak DIFA elde ettiği bu bilgilere göre, 7.7.7.7 IP adresine gelecek olan 80 nolu port isteklerinin 192.168.1.10 IP adresine sahip olan sunucuya gönderileceğini tahmin etmektedir. Bu gereksinimi sağlayacak yapılandırmayı kendisi otomatik olarak hazırlar ve onay için ağ yöneticisine mail gönderir. Ağ yöneticisinin onay vermesi ile hazırlanan kurallar hâlihazırdaki güvenlik duvarına ssh ile bağlantı kurularak uygulanır. Şekil 5'de bu sistemin akış şeması gösterilmektedir.



Şekil 5. DIFA'nın Akış Şeması (Flow Control of DIFA)

İnternet tarafından gelen her bir paket için; W_s paketin kaynak adresi, W_d paketin hedef adresi, W_{sp} paketin kaynak port numarasi, W_{dp} paketin hedef port numarasını göstermektedir. LAN'daki her bir sunucu ise L ile ve bu sunucuların hizmet verdiği port ise L_{hp} ile gösterilmiştir. LAN'daki n tane L sunucusu için aşağıdaki model kullanılarak kural oluşturma işlemi gerçekleştirilir.

```
begin
for (L=L1, L=Ln,
   while (Wdp == Lhp) then
      C=+1
      if C >= Eşik Degeri  then
         Ws == Any & Wd == Ls & Wsp == Wsp &
         Wdp == Lhp & Wsp == Wsp & permit
         Ws == Any & Wd == Any & Wsp == Any &
         Wdp == Lhp & Wsp == Any & deny
      end if
   end
end
```

Tablo 2. DIFA'nın tasarım kriterleri açısından karşılaştırılması (Comparison for design criteria of DIFA)

ARAÇLAR	Kural Oluşturma Biçimi	Dahili Kural Çakışması Kontrolü	Harici Kural Çakışması Kontrolü	Platform Bağımsızlığı	Merkezi Yönetim	NAT kuralı oluşturma
DIFA	Otomatik	✓	×	✓	×	✓
CONFIDDENT	El ile	✓	×	✓	✓	×
ENAVis	El ile	✓	✓	✓	×	×
MBD	El ile	✓	×	✓	✓	×
ACO-PF	El ile	✓	✓	×	✓	×

Geliştirilen bu yaklaşım sürekli olarak kendisini yenileyebilecek şekilde tasarlanmıştır. Erişim kuralları oluşturulduktan sonra, DIFA sürekli dinleme ve kontrol etme işlemine devam eder. Her 24 saatin sonunda DIFA-EDP ve DIFA-IDP alanlarında toplanan veriler incelenir. Üzerinde mevcut bulunan kurallardan farklı bir kurala neden olacak bir eşleşme bulunursa, ağ yöneticisine DIFA tarafından haber verilir ve uygulamak için onay istenir. Eğer DIFA-IDP içerisinde bulunan verilerden aktif olmayan bir kayıt ile karşılaşırsa, 12 saat boyunca bu kayıt karantinaya alınır. Her saatte bir olmak üzere aktif olup olmadığını kontrol edilir. Eğer 12 saat sonunda ilgili sunucuya hizmet verdiği portlardan erişilmiyor ise, erişim kurallarından ilgili sunucuya ait kuralların silinmesi için gerekli olan düzenlemeler yapılır. Ağ yöneticisine erişim kuralının silineceğine dair mail gönderir ve onay beklenir. Ağ yöneticisi tarafından onay verilmesi ile erişim kuralları düzenlenmiş olunur. Tasarım kriterleri dikkkate alınarak DIFA'nın literatürde öne çıkan erişim kontrol listesi oluşturma yaklaşımları ile karşılaştırılması Tablo 2'de gösterilmiştir.

Razzag ve arkadaşları güvenlik duvarları ile ilgili bir analiz çalışması yapmışlardır. Farklı güvenlik duvarlarını bir takım guvenlik kriterlerini dikkate alarak incelemişlerdir [22]. Bu çalışmadaki güvenlik kriterleri temel alınarak DIFA üzerinde bir güvenlik analizi yapılmıştır. Bu analizler sonucunda elde edilen veriler Tablo 3'de sunulmuştur.

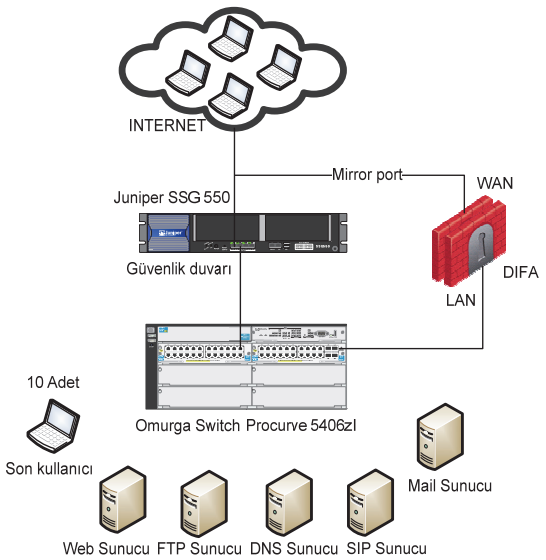
Tablo 3. Geliştirilen güvenlik duvarının güvenlik analizi (The security analysis of the improved firewall)

Saldırı Türleri	LAN	WAN	DMZ
SYN taşırma saldırısı	✓	✓	✓
Durum tablosu taşırma	✓	✓	✓
ARP zehirlemesi	✓	×	✓
IP yanıltma saldırısı	✓	✓	✓
Ortadaki adam saldırısı	✓	×	✓
DNS yanıltma saldırısı	×	✓	✓
Oturum koruma	✓	✓	✓
Trafik İzleme	✓	✓	×
Trafik Engelleme	✓	✓	✓

Yakın gelecekte tüm iletişim altyapısı için IPv6 adresleme yapısının kullanılacak olması, ağ cihazlarının IPv6 ile uyumlu çalışmasını zorunlu kılmaktadır. DIFA'nın çalışma mantığı adresleme yapısına bağlı olarak değişmemektedir. DIFA, IPv6 adreslemenin kullanıldığı ortamlarda aynı çalışma prensibi ile IPv4'de olduğu gibi kuralları oluşturabilir. Ayrıca IPv6 adreslemede NAT'ın kullanılmayacak olması DIFA için olumlu bir durumdur. Çünkü DIFA, NAT işlemini gerçekleştirmek için gerekli olan kuralları oluşturmak zorunda kalmayacaktır.

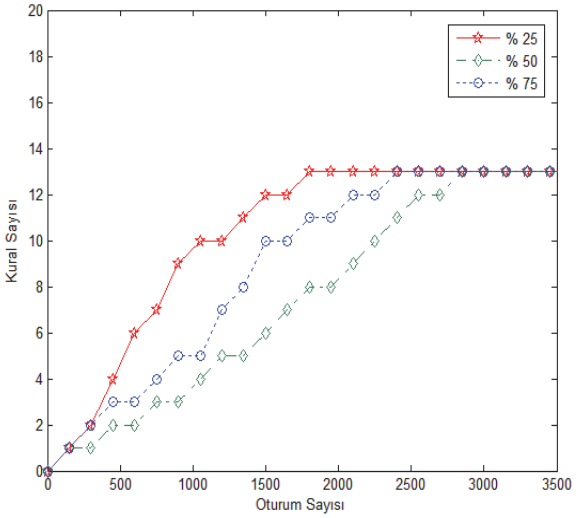
4. DENEYSEL SONUÇLAR (EXPERIMENTAL RESULTS)

DIFA'nın verimliliğini test edebilmek için bir laboratuvar ortamı kurulmuştur. Bu ortam içerisinde bir adet Juniper SSG 550 güvenlik duvarı, bir adet Procurve 5406zl omurga anahtar, bir adet Procurve 2510 kenar anahtar, web, mail, ftp, dns, sip sunucu ve 14 adet bilgisayar kullanılmıştır. 4 adet bilgisayar internet ortamını temsil etmektedir ve 2510 kenar anahtara bağlıdır. Bu bilgisayarlar ile sürekli olarak LAN tarafındaki sunuculara istek gönderilmektedir. Ayrıca, LAN'da bulunan son kullanıcıları temsilen 10 adet bilgisayar kullanılmıştır. DIFA'nın internet ortamını temsil eden portu 2510 kenar anahtara bağlıdır. Bu portta "mirroring" uygulanarak Juniper güvenlik duvarına gelen tüm isteklerin DIFA'ya ulaşması sağlanmıştır. DIFA'nın diğer portu LAN tarafındaki sunucuların bağlı olduğu 5406zl omurga anahtara bağlanmıştır. Test için oluşturulan laboratuvar ortamı Şekil 6'da gösterilmiştir.



Şekil 6. Test için oluşturulan laboratuvar ortamı (Laboratory environment established for testing)

Laboratuvar ortamı bu hali ile gerçek bir ağ ortamını temsil etmede yeterli değildir. Çünkü gerçek dünyada, internet güvenli bir ağ ortamı olarak kabul edilmez. Bu yüzden, test yapılacak laboratuvar ortamını belli oranlarda güvensiz hale getirmek daha gerçekçi sonuçlar alınması açısından önemlidir. Ayrıca internet ortamına benzer bir yapı sağlanarak gerçek bir ağ ortamı elde edilmiş olunur. Gerçek bir ağ ortamını oluşturabilmek için, 4 adet bilgisayar normal ve saldırgan kullanıcı olacak şekilde düzenlenerek internet ortamına benzer heterojen bir yapı sağlanmıştır. Test ortamında saldırgan kullanıcı sayısı oranının toplam kullanıcıya sayısına oranı, güvensizlik oranı olarak tanımlanmıştır. Örneğin, 1 adet saldırgan ve 3 adet normal kullanıcının olduğu ortamın güvensizlik oranı %25'dir. Normal kullanıcılardan LAN'daki sunucuların hizmet verdikleri portlara doğru normal istekler gönderilirken, saldırgan bilgisayardan sürekli olarak port taraması yapılıp aldatıcı paketler oluşturularak LAN'daki sunuculara sahte bağlantı isteği gönderilmektedir. LAN'daki 5 adet sunucunun internet ortamına erişime açılması gereken tcp/udp port sayısı 13 tanedir. Bu portlar tcp 21, 80, 443, 25, 587, 110, 993, 143, 995, 53 ve udp 53, 5060, 5061'dir.



Şekil 7. Farklı güvensizlik ortamlarında DIFA'nın kural oluşturma durumu (Creating insecurity situation in the different rules of DIFA)

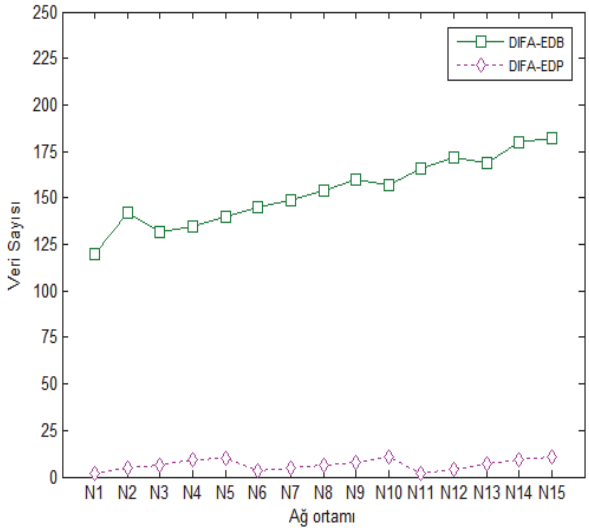
Laboratuvar ortamında DIFA'nın çalışma durumu Şekil 7'de gösterilmiştir. İnternet ortamında bulunan 4 adet bilgisayar istenilen güvensizlik oranına göre, hping3 ve nmap araçları kullanılarak saldırgan bilgisayarlar haline getirilmiştir. Elde edilen sonuçlar, güvensizlik oranının %25 olduğu ortamda DIFA'nın gerekli olan NAT ve ACL kurallarını oluşturması daha sayıda az oturum isteği ile gerçekleşirken, güvensizlik oranı arttıkça daha fazla sayıda oturum isteği ile gerçekleştiğini göstermiştir. Bunun sebebi ise, saldırgan bilgisayarların farklı tcp/udp portlarından birçok oturum isteği göndermesidir. Bu yüzden saldırgan bilgisayar sayısının artması, DIFA'ya gelen oturum sayısını arttırmıştır. Ancak, DIFA'nın oluşturduğu kural sayısında bir artış olmamıştır. Bu sonuç DIFA'nın başarılı şekilde sadece ihtiyaç duyulan kuralları oluşturduğunu göstermiştir.

Oluşturulan laboratuvar ortamında, daha detaylı analizler yapabilmek için DIFA'nın koruyacağı ağlar gruplandırılmıştır. Gruplandırma işlemi ağda hâlihazırda aktif bulunan cihaz sayısı ve internet ortamına hizmet veren tcp/udp port sayısı dikkate alınarak yapılmıştır. Her farklı ağ ortamında DIFA'nın verimliliği gözlemlenmiştir. Tablo 4'de test amaçlı sınıflandırılan ağ ortamları gösterilmektedir. Örneğin, toplam aktif cihaz sayısı 1 ile 5 arasında ve hizmet verdiği tcp/udp port sayısı 10 olan ağ ortamı N_5 olarak adlandırılırmıştır.

Tablo 4. Test için kullanılan gerçek ağ ortamları (The real network environments used for testing)

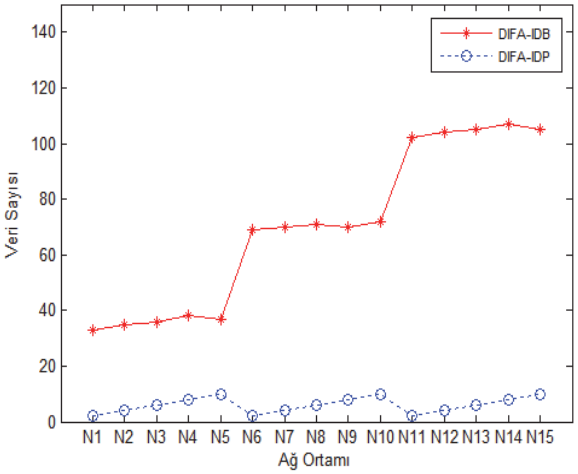
	Hizmet verilen tcp/udp port sayısı				
Cihaz sayısı	2	4	6	8	10
1-5	N_1	N_2	N_3	N_4	N_5
6-10	N_6	N_7	N_8	N_9	N_{10}
11-15	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}

Her bir ağ ortamı için sonuçlar elde edilmiştir. Bu sonuçlara göre DIFA'nın verimliliği ölçülmüştür. Şekil 8'de 15 farklı ağ ortamı için DIFA-IDB ve DIFA-IDP alanlarının analizi sonucu elde edilen grafik gösterilmektedir. DIFA, DIFA-IDB alanındaki verileri koruduğu ağ ortamındaki ip bloklarını tarayarak elde etmektedir. N_1 ağ ortamı için DIFA-IDB alanında toplamda yaklaşık 35 kadar açık port sayısı bulunurken, N_{15} ağ ortamı için bu değer 105'e kadar çıkmaktadır. Yerel alan ağından internet ortamına hizmet verilen tcp/udp port sayısının DIFA-IDP alanına doğrudan etkisi olduğu için, DIFA-IDP'deki veri sayısı, hizmet verilen tcp/udp port sayısına göre değişmektedir.



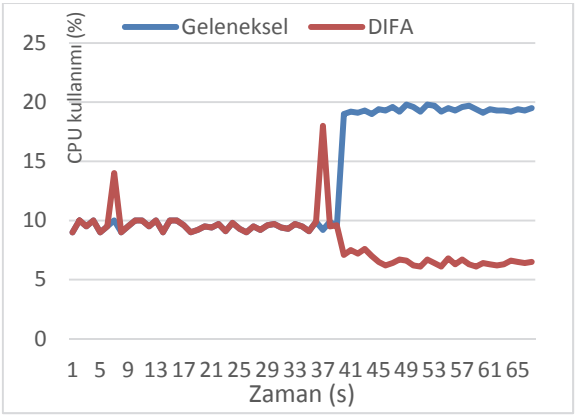
Şekil 8. DIFA-IDB ve DIFA-IDP alanındaki veri sayısı (The number of data in the field of DIFA-IDB and DIFA-IDP)

Şekil 9'da 15 farklı ağ ortamı için DIFA-EDB ve DIFA-EDP alanlarının analizi sonucu elde edilen grafik gösterilmektedir. DIFA-EDP; DIFA-EDB'deki verilerin filtrelenmesi sonucu oluşturulmaktadır. N_{15} ağ ortamı için DIFA-EDB'de yaklaşık 182 tane veri bulunmaktadır. Bu verilerden uygun olanları DIFA-EDP alanını oluşturmaktadır. Bu yüzden DIFA-EDP'de yaklaşık 12 tane veri bulunmaktadır. DIFA-EDP alanındaki veri sayısı ile DIFA tarafından oluşturulan kural sayısı arasında bir paralellik mevcuttur. İnternet ortamına açılacak port sayısı arttıkça, DIFA-EDB alanındaki veri sayısı ve DIFA tarafından oluşturulan kural sayısında artış olmaktadır. Ağ ortamındaki istemci sayısının internet ortamından yerel alan ağına doğru oluşturulacak erişim kuralı sayısı üzerine bir etkisi olmamaktadır. Gerçek durumda da olması gereken, internet ortamına hizmet verilen port sayısının artması sonucu güvenlik duvarları üzerindeki kural sayısının artmasıdır. İstemci sayısının artması sadece DIFA-IDB alanını etkileyecek bir durumdur.

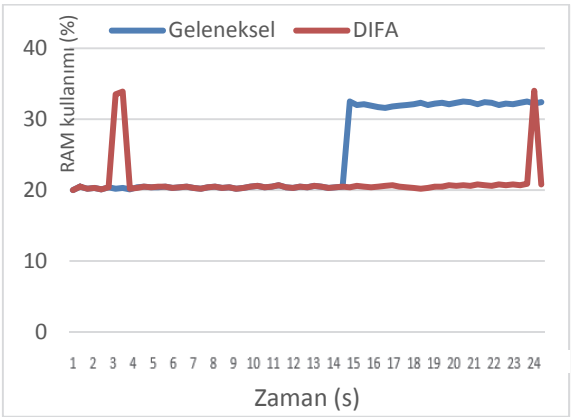


Şekil 9. DIFA-EDB ve DIFA-EDP alanındaki veri sayısı (The number of data in the field of DIFA-EDB and DIFA-EDP)

DIFA'nın zaman performansını ölçmek için laboratuvar ortamındaki web, ftp ve dns sunucuları DIFA'nın değişiklik kontrolü yapmasına 3 saat kala kapatılmıştır. DIFA'nın çalışması ve kural sayılarındaki değişim 24 saatlik periyod içerisinde incelenmiştir. Aynı donanımsal özellikler kullanılarak aynı laboratuvar ortamında geleneksel güvenlik duvarı için de benzer bir test gerçekleştirilmiştir. 24 saatlik periyotda DIFA ve gelenksel güvenlik duvarı yaklaşımının CPU ve RAM kullanım oranları sırasıyla Şekil 10 ve 11'de gösterilmiştir. DIFA 3'üncü saatde EDP ve IDP alanları ile ilgili işlem yaptığından CPU ve RAM kullanım oranında bir sıçrayış olmuştur. Bunun dışında 15'inci saate kadar DIFA ve geleneksel yöntem için benzer CPU ve RAM tüketim oranı gözlemlenmiştir. 15'inci saatten sonra DIFA'nın CPU kullanımında bir düşüş olurken, geleneksel yöntemde bir artış olmuştur. Benzer şekilde, DIFA'nın RAM kullanımında bir değişim olmazken, geleneksel yöntemde bir artış meydana gelmiştir. Bunun sebebi ise, 15'inci saatten sonra DIFA'nın web, ftb ve dns sunucuların kapanmış olduğunu anlaması ve ilgili kuralları silmesidir. Geleneksel yaklaşımda ise ilgili kurallar hala kural tablosunda durmaktadır. DIFA, WAN'dan bu sunuculara doğru gelen paketleri kural kontrolü sonucu hiç bir işlem yapmadan direk düşürürken, geleneksel güvenlik duvarı yaklaşımı ise kural kontrolü sonucu paketlerin geçişine izin vererek hedeflerine iletebilmek için işlem yapmaya çalışmaktadır. Bundan dolayı RAM ve CPU kullanımda artış meydana gelmiştir.



Şekil 10. DIFA ve geleneksel güvenlik duvarı yaklaşımının CPU kullanım oranı (CPU usage of DIFA and traditional firewall)



Şekil 11. DIFA ve geleneksel güvenlik duvarı yaklaşımının RAM kullanım oranı (RAM usage of DIFA and traditional firewall)

5. SONUÇLAR (CONCLUSIONS)

Bu makalede, günümüz güvenlik duvarlarının çalışma mimarileri gözden geçirilmiştir. Mevcut güvenlik duvarlarının çalışma mimarilerine farklı bir bakış açısı ile yaklaşılmıştır. DIFA olarak adlandırılan yeni bir güvenlik duvarı mimarisi geliştirilmiştir. DIFA'nın verimliliği gerçek dünya ortamına benzer bir laboratuvar ortamı kurularak test edilmiştir. İnternet tarafı saldırgan bilgisayarlar kullanılarak güvensiz hale getirilmiştir. Bundan dolayı laboratuvar ortamından elde edilen sonuçların gerçek ortamlardan elde edilecek sonuçlara benzer olması beklenmektedir. DIFA mimarisi ile güvenlik

duvarlarının kendi kendisini yönetebilmesi amaçlanmıştır. Güvenlik duvarları üzerindeki erişim kontrol kurallarının oluşturulması zor ve karmaşık bir iştir. Bu yüzden, ağ yöneticilerinin güvenlik duvarları üzerinde sadece kontrol etme işlemini yapan taraf olması hedeflenmiştir. DIFA ile ağ yöneticilerinin güvenlik duvarı yönetimine ayıracakları zaman ve bilgi gereksinimi azalacaktır. Ayrıca, DIFA erişim için açılması gerekli olan TCP/IP portlarını kendisi bulup, kendisi açıp kapatacağı için daha güvenli bir yapı sunmaktadır. İnsan hatasından kaynaklanan, bir güvenlik açığının oluşma ihtimali en aza indirilmiştir. DIFA yaklaşımı diğer ağ cihazları ile entegre edilirse daha etkin şekilde kullanılabilir.

İnternetin hızla büyüyen yapısı göz önüne alındığında, DIFA mimarisine benzer dinamik güvenlik ürünlerinin geliştirileceği tahmin edilebilir. İlerleyen zamanlarda bilişim alanında çalışanların sadece sistemin istenildiği gibi çalışmasını kontrol eden taraf olacağı öngörülmektedir. DIFA mimarisi sadece güvenlik alanında bir yaklaşım olarak düşünülmemelidir. Bir bilgi sisteminin oluşturulması ve yönetilmesi sırasında ihtiyaç duyulan her alanda uygulanabilecek ürünler ve sistemler DIFA yaklaşımı ile geliştirilebilir.

KAYNAKLAR (REFERENCES)

1. Davy, S., Jennings, B., Strassner, J., "The Policy continuum-Policy authoring and conflict analysis", **Computer Communications,** Cilt 31, No 13, 2981-2995, 2008.
2. Lee, S., Kim, H.S., "End-user perspectives of Internet connectivity problems", **Computer Networks,** Cilt 56, No 6, 1710-1722, 2012.
3. Alshammari, R., Zincir-Heywood, A.N., "Can encrypted traffic be identified without port numbers, IP addresses and payload inspection?", **Computer Networks,** Cilt 55, No 6, 1326-1350, 2011.
4. Botta, A., Dainotti, A., Pescape, A., "A tool for the generation of realistic network workload for emerging networking scenarios", **Computer Networks,** Cilt 56, No 15, 3531-3547, 2012.
5. Gouda, M.G., Liu, A.X., "Structured firewall design", **Computer Networks,** Cilt 51, No 4, 1106-1120, 2007.
6. Chao, C.S., Yang, S.J., "A novel three-tiered visualization approach for firewall rule validation", **Journal of Visual Languages and Computing,** Cilt 22, No 6, 401-414, 2011.
7. Liu, A.X., "Firewall policy verification and troubleshooting", **Computer Networks,** Cilt 53, No 16, 2800-2809, 2009.
8. Pozo, S., Ceballos, R., Gasca, R.M., "Model-Based Development of firewall rule sets: Diagnosing model inconsistencies", **Information and Software Technology,** Cilt 51, No 5, 894-915, 2009.
9. Pozo, S., Gasca, R.M., Reina-Quintero A.M, Varela-Vaca A.J, "CONFIDDENT: A model-driven consistent and non-redundant layer-3 firewall ACL design, development and maintenance framework", **The Journal of Systems and Software,** Cilt 85, No 2, 425-457, 2012.
10. Sreelaja, N.K., Pai, G.A.V., "Ant Colony Optimization based approach for efficient packet filtering in firewall", **Applied Soft Computing,** Cilt 10, No 4, 1222-1236, 2010.
11. Kim, S., Kim, S., Geuk, L., "Structure design and test of enterprise security management system with advanced internal security", **Future Generation Computer Systems,** Cilt 25, No 3, 358-363, 2009.
12. Abdulmohsin, I.M.A., "Techniques and algorithms for access control list optimization", **Computers and Electrical Engineering,** Cilt 35, No 4, 556-566, 2009.
13. Lee, S., Wong, T., Kim, H.S., "Improving manageability through reorganization of routing-policy configurations", **Computer Networks,** Cilt 56, No 14, 3192-3205, 2012.
14. Liao, Q., Blaich, A., VanBruggen, D., Striegel, A., "Managing networks through context: Graph visualization and exploration", **Computer Networks,** Cilt 54, No 16, 2809-2824, 2010.
15. Liao, H., Lin, C.R., Lin, Y., Tung, K., "Intrusion detection system: A comprehensive review", **Journal of Network and Computer Applications,** Cilt 36, No 1, 16-24, 2013.
16. Njogu, H.W., Jiawei, L., Kiere, J.N., Hanyurwimfura D., "A comprehensive vulnerability based alert management approach for large networks", **Future Generation Computer Systems,** Cilt 29, No 1, 27-45, 2013.
17. Zhang, S., Li, J., Chen, X., Fan, L., "Build network attack graph for alert causal correlation", **Computers&Security,** Cilt 27, No 5-6, 188-196, 2008.
18. Morin, B., Me, L., Debar, H., Ducasse, M., "A logic-based model to support alert correlation in intrusion detection", **Information Fusion,** Cilt 10, No 4, 285-299, 2009.
19. Li, J., Li, B., Wo, T., Hu, C., Huia, J., Lui, L., Lam, K.P., "CyberGuarder: A virtualization security assurance architecture for green cloud computing", **Future Generation Computer Systems,** Cilt 28, No 2, 379-390, 2012.
20. Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., Rajarajan, M., "A survey of intrusion detection techniques in Cloud", **Journal of Network and Computer Applications,** Cilt 36, No 1, 42-57, 2013.
21. Patel, A., Taghavi, M., Bakhtiyari, K., Junior, J.C., "An intrusion detection and prevention system in cloud computing: A systematic review", **Journal of Network and Computer Applications,** Cilt 36, No 1, 25-41, 2013.
22. Razzag, A., Hur, A., Shahbaz, S., Masood, M., Ahmad, H.F., "Critical Analysis on Web Application Firewall Solutions", **IEEE Eleventh International Symposium on Autonomous Decentralized Systems,** Mexico City, Mexico, 1-6, 6-8 Mart 2013.