DoxTracker: Doküman Kaçak Takip Sistemi

Bu tez Bilgi Güvenliği Mühendisliği'nde
Tezli Yüksek Lisans Programının bir koşulu olarak

Ulaş Kaya
tarafından

Fen Bilimleri Enstitüsü'ne
sunulmuştur.

Bu tezi okuduk, kapsam ve nitelik açısından Bilgi Güvenliği Mühendisliği alanında Yüksek Lisans derecesi için tümüyle uygun olduğu görüşüne vardık.

ONAYLAYANLAR:

Dr. Mehmet Kara
(Tez Danışmanı) .....

Prof. Dr. Tahsin Erkan Türe .....

Prof. Dr. Nevcihan Duru .....

Bu tez İstanbul Şehir Üniversitesi, Fen Bilimleri Enstitüsü tarafından belirlenen tüm koşullara uygundur.

ONAY TARİHİ: 24 Aralık 2015

MÜHÜR/İMZA:

Yazarlık Beyanı

Ben, Ulaş KAYA, başlığı, 'DoxTracker: Doküman Kaçak Takip Sistemi' olan tezin ve içinde sunulan bilgilerin şahsıma ait olduğunu beyan ederim. Ayrıca:

- Bu çalışmanın bütünü veya esası bu üniversitede Yüksek Lisans derecesi elde etmek üzere çalıştığım süre içinde gerçekleştirilmiştir.
- Daha önce bu tezin herhangi bir kısmı başka bir derece veya yeterlik almak üzere bu üniversiteye veya başka bir kuruma sunulduysa bu açık biçimde ifade edilmiştir.
- Başkalarının yayımlanmış çalışmalarına başvurduğum durumlarda bu çalışmalara açık biçimde atıfta bulundum.
- Başkalarının çalışmalarından alıntıladığımda kaynağı her zaman belirttim. Tezin bu alıntılar dışında kalan kısmı tümüyle benim kendi çalışmamdır.
- Esaslı yardım aldığım bütün kaynaklara teşekkür ettim.
- Tezde başkalarıyla birlikte gerçekleştirilen çalışmalar varsa onların katkısını ve kendi yaptıklarımı tam olarak açıkladım.

İmza: 

Tarih: 24.12.2015

"Basit bir insanın elinden geleni yapabilmesi, zeki bir insanın tembelliğinden çok daha değerlidir."

Baltasar Bracias

"Durmak ölüm, taklit uşaklıktır; çalışmak ve yetişmek ise hayat ve özgürlüktür."

L.Y. Rauch

DoxTracker: Doküman Kaçak Takip Sistemi

Ulaş Kaya

Öz

Bilgi sistemleri kullanım oranının her geçen gün arttığı günümüzde, dış kaynaklı saldırılara göre çok daha fazla zarar verici olabilen iç kaynaklı saldırılara ve iç tehdit unsurlarına karşı halen geliştirilmiş güçlü bir önlem bulunmamaktadır. Bu çalışma kapsamında; hali hazırda kullanılan güvenlik mekanizmalarını güçlendirebilecek, saldırı tespit ve önleme yetkinliğine katkıda bulunabilecek ve hassas/gizli dokümanların korunabilmesine imkan tanıyacak bir kontrol-takip sistemi önerilmiştir. Önerilen sistem temel olarak dokümanların açılıp açılmadığının tespit edilmesine dayanmaktadır. Bu sistem sayesinde; korunan bir dokümanın ne zaman, hangi IP adresinden ve coğrafi konumdan, hangi kullanıcı tarafından açıldığı web tabanlı takip merkezi üzerinden anlık izlenebilecektir.

Anahtar Sözcükler: Doküman takip sistemi, güvenlik önlemleri, iç tehdit, saldırı tespiti, tuzak doküman, tuzak sistem

Saygıdeğer aileme ve sevgili eşime . . .

Teşekkür

Öncelikle bu çalışmam süresince her türlü yardım ve fedakârlığı sağlayan; bilgi, tecrübe ve güler yüzü ile çalışmama ışık tutan danışmanım Sayın Dr. Mehmet Kara'ya teşekkür ederim.

Tezimin hazırlanması sırasında beni cesaretlendiren, ümit veren ve manevi destek sağlayan değerli iş arkadaşlarıma bilhassa, Bâkır Emre ve Ömer Erdem'e teşekkürü bir borç bilirim.

İçindekiler

Şekil Listesi

Kısaltmalar

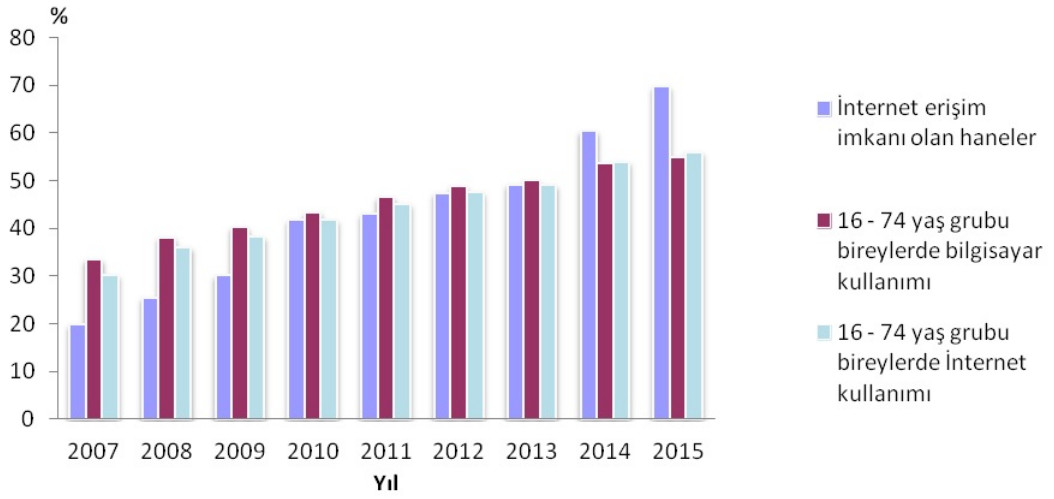
ABD	**A**merika **B**irleşik **D**evletleri
APT	**A**dvanced **P**ersistent **T**hreat
CC	**C**ommon **C**riteria
CIA	**C**entral **I**ntelligence **A**gency
DARPA	**D**efense **A**dvanced **R**esearch **P**rojects **A**gency
DDOS	**D**istrubuted **D**enial **O**f **S**ervice
DLP	**D**ata **L**oss **P**revention
DNS	**D**omain **N**ame **S**ystem
HTML	**H**yper **T**ext **M**arkup **L**anguage
HTTP	**H**yper **T**ext **T**ransfer **P**rotocol
IDS	**I**ntrusion **D**etection **S**ystem
IPS	**I**ntrusion **P**revention **S**ystem
ISO	**I**nternational **O**rganization for **S**tandardization
JPEG	**J**oint **P**hotographic **E**xperts **G**roup
KGB	**K**omitet **G**osudarstvennoy **B**bezopasnosti
MIT	**MIT L**icense
MVC	**M**odel **V**iew **C**ontroller
NSA	**N**ational **S**ecurity **A**gency
OOP	**O**bject **O**riented **P**rogramming
OOXML	**O**pen **XML**
TÜBİTAK	**TÜ**rkiye **Bİ**limsel **T**eknolojik **A**raştırma **K**urumu
TÜİK	**TÜ**rkiye **İ**statistik **K**urumu
PDF	**P**ortable **D**ocument **F**ormat
PHP	**H**yper **T**ext **P**reprocessor
PNG	**P**ortable **N**etwork **G**raphics

PRC	**P**rivacy **R**ights **C**learinghouse
SMS	**S**hort **M**essage **S**ervice
SNMP	**S**imple **N**etwork **M**anagement **P**rotocol
VPN	**V**irtual **P**rivate **N**etwork
XML	**E**xtensible **M**arkup **L**anguage

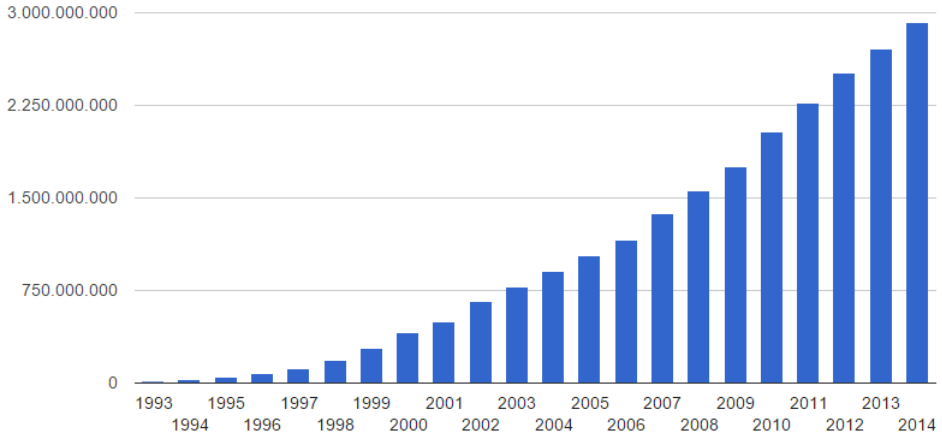
Bölüm 1

Giriş

Tüm zamanların en eğitimli bilgisayar korsanlarından biri olarak kabul edilen Robert Tappan Morris, master projesi olarak kendini kopyalayabilen, kendi kendine yayılabilen ve internet ortamını da kullanabilen dünyanın ilk bilgisayar solucanını kodlamıştır. Kendi soyadından yola çıkarak Morris Solucanı adını verdiği bu programı test etmek isteyen Morris, internete bağlı olan sistemlerin yüzde 10'unun ya tamamen bozulmasına ya da devre dışı kalmasına sebep olmuştur. Bulaştığı bilgisayarlara 200 ile 5000$ arasında zarar veren Morris solucanı interneti kullanan ve kullanmayı düşünen insanlar üzerinde olumsuz etkiler yaratmış ve insanların internete olan güvenini sarsmıştır [1]. Yaşanan bu kötü olaya rağmen internet üzerine yapılan çalışmalara yoğun bir şekilde devam edilmiştir. 20 yıllık bir emek ve kazanılan tecrübeler neticesinde de 90'lı yılların başında Minnesota Üniversitesi'nde sıradan insanların da rahatlıkla kullanabileceği bir internet arayüzü geliştirilmiştir [2]. Sonraları ise teknolojinin baş döndürücü hızdaki gelişimine paralel olarak bilgisayar ve akıllı telefonlar gibi elektronik cihazların internet ortamında kullanım oranı bir hayli artmıştır. Kullanım kolaylığı, zamandan tasarruf sağlaması, hızlı olması ve daha sayılabilecek bir çok avantajlarından ötürü her geçen gün internete bağlı bu tür cihazların kullanımı artmıştır. Bu kapsamda bakıldığında Türkiye nüfusunun yarısından fazlası, dünya nufüsunun ise yarısına yakını bu cihazları internet ortamında kullanır duruma gelmiştir [3]. TÜİK verilerine göre; Türkiye'de Nisan 2015 itibariyle hanelerin yüzde 70'ı internet erişim imkânına sahip iken 16-74 yaş grubu bireylerin yüzde 55'i internet kullanmaktadır [4].



ŞEKİL 1.1: TÜİK verilerine göre Türkiye'de internet ve bilgisayar kullanım oranları



ŞEKİL 1.2: Dünya genelinde internet kullanımının yıllara göre dağılımı

Teknolojinin gelişmesine paralel olarak daha yakın temas halinde olmaya başladığımız, güvensiz bir ortam olan internet üzerinde bu kadar çok işlemin güvenli bir şekilde gerçekleştirilebilmesinin sağlanması ise gelişen teknolojinin en büyük dezavantajlarından birisidir. PRC isimli derneğin 2005 yılından itibaren kayda geçirdiği bilişim güvenliği ihlalleri olaylarına göre, istisnasız her yıl büyük bir bilgi hırsızlığı olayı yaşandığı tespit edilmiştir. Bugüne kadar sadece bu derneğin raporladığı 816 milyon bilişim güvenliği ihlali, durumun vahametini gözler önüne sermektedir [5].

Bilginin güvenli olarak saklanması, işlenmesi ve iletilebilmesi için donanımsal veya yazılımsal olarak çok çeşitli teknolojiler kullanılmakta olsa da bu teknolojileri tasarlayanların ve kullananların insan olduğunun unutulmaması gerekir. Teknolojiler ile ilgili eksiklikler tespit edildiğinde gerekli düzenlemeler yapılarak bu eksiklikler giderilebilmekte ve aynı hatanın tekrarlanması önlenebilmektedir. Fakat insan, yapısı gereği öğrendiğini

unutabilen veya yapmaması gerektiğini bildiği bir davranışı yanlışlıkla ya da kasten yapabilen bir varlıktır. Bu yüzden, özellikle değişen ve gelişen dünya şartlarında, bilgiyi korumak adına gerekli olan farkındalık periyodik olarak güncellenmeli ve sürekli olarak güncel tutulmalıdır.

Bilgisayar sistemlerinde gizlilik, bütünlük ve sürekliliğin sağlanabilmesi için birçok kuruluş ve ülke tarafından bugüne kadar çok sayıda standartlar, çerçeveler (framework) ve ürünler geliştirilmiş ve hali hazırda da bu alanda çalışmalar devam etmektedir. ISO 27001, ISO 19790, Ortak Kriterler (Common Criteria) standartları, antivirüs yazılımları, güvenlik duvarları, VPN (Virtual Private Networks) yazılımları/donanımları, saldırı tespit ve önleme sistemleri ve içerik kontrolcüler bu çalışmalara örnek verilebilir. Geliştirilen bu teknik çözümlerin yanında bilişim sistemlerinin güvenli bir şekilde tasarlanması ve yönetilebilmesi amacıyla Bell-LaPadula ve Clark-Wilson modelleri gibi modeller de önerilmiştir [6, 7]. Genel olarak bakıldığında, yapılan bu çalışmaların odak noktası sisteme ve bilgiye izinsiz/yetkisiz erişimi engellemek olmuştur. Fakat tüm bunlara rağmen yani geliştirilen sistemlerin ve alınan önlemlerin çoğu zaman iç tehdit karşısında yetersiz kaldığı görülmektedir.

Amerikan Bilgisayar Güvenliği Enstitüsü'nün (US Computer Security Institute) raporuna göre iç kaynaklı tehditlerin sebep oldukları güvenlik olaylarının sayısı virüs ve worm kaynaklı güvenlik olaylarının sayısını geçmiştir [8]. 2010 yılında yayınlanan bir rapora (E-crime Watch Survey) göre ise elektronik ortamda işlenen suçlar arasında iç kaynaklı suçlar ikinci sırayı almıştır [9].

İç kaynaklı tehditlerin sebep oldukları olaylara verilebilecek en güzel örnek, eski CIA ve NSA çalışanı olan Edward Joseph Snowden'in gizli NSA belgelerini wikileaks isimli internet sitesi üzerinden ifşa etmesidir. Bu ifşa operasyonu ile ABD hükümetinin itibarı zedelenmiş ve NSA tarafından yürütülen bir çok gizli illegal çalışma da gün yüzüne çıkartılmıştır [10]. Diğer çarpıcı bir örnek ise dünyanın en büyük on bankasından biri olan Barclays Bank çalışanlarından birinin 27000 müşteriye ait tüm bilgileri çalması ve bankada bulunan bilgileri de silmiş olmasıdır. Bu çapta bir bilgi kaybı sonucunda Barclays bankasının milyon dolarlar kaybettiği iddia edilmektedir [11]. McLaren/Mercedes'in Ferrari'ye ait bilgileri ele geçirmesi, Amerikan savunma endüstrisinde normal bir çalışan olarak görülen fakat içeriden bilgi sızdıran Çinli Chi Mak, işten kovulunca işverene ait gizli ticari sırları çalarak rakip bir Güney Kore firmasına satan Michael Mitchell örneğinde

olduğu gibi çok sayıda iç tehdit unsurlarının sebep oldukları olaylardan bahsedilebilir [12]. İç tehdit kaynaklı yaşanan son olaylardan biri de 'Game of War: Fire Age' üst düzey bir yöneticisinin karıştığı şirket içi doküman hırsızlığıdır [13]. Bu olay günümüz üstün teknolojilerinin iç tehdit karşısında yetersiz kaldığını ve DoxTracker gibi bir yaklaşıma ihtiyaç olduğunu bir kez daha göstermiştir.

Bu kapsamda değinilmesi gereken bir diğer önemli konu ise bulut bilişimdir. Çünkü popülerliği ve kullanım oranı her geçen gün artan bulut bilişim ortamında da en büyük tehlike unsuru iç tehdit olarak gösterilmiştir [14]. Bulut ortamında yaşanan veri çalma saldırılarına verilebilecek en iyi örnek Twitter olayıdır. Fransız bir bilgisayar korsanı olan ve Hacker Croll olarak tanınan François Cousteix, Twitter'ın ürün yönetim müdürü olan Jason Goldman'ın yahoo mail hesabını 'şifremi unuttum' seçeneğini kullanarak ele geçirmiştir. Daha sonra Goldman'ın maillerini kurcalarken Twitter'ın admin/yönetici kullanıcısına ait erişim şifresini bulmuştur. Bunun üzerine bu şifreyi kullanarak Twitter yönetim paneline erişim sağlamıştır. Bu hamlesi ile Hacker Croll, yönetici haklarına sahip bir kullanıcıyı yani Jason Goldman'ı taklit eden kötü niyetli bir iç tehdit unsuru konumuna gelmiştir. Nitekim bu olay başta Twitter ve müşterilerini ciddi zararlara uğratacak şekilde Twitter'ın kurumsal bilgilerinin; Ashton Kutcher, Lily Rose Allen ve Barack Obama gibi ünlü isimlerin de itibarını sarsacak gizli bilgilerin sızdırılması ile neticelenmiştir [15]. Bu olayda da görüldüğü üzere iç kaynaklı bir saldırı, çok kolay bir şekilde çok büyük ve hesapta olmayan zararlar verebilmektedir.

Çoğu zaman dış kaynaklı saldırılara göre çok daha fazla zarar verici olabilen iç kaynaklı saldırılara ve tehdit unsurlarına karşın halen farkındalık ve güvenlik eğitimleri dışında güçlü bir önlem/mekanizma bulunmamaktadır. İç kaynaklı saldırılar hala daha saldırı tespit edildikten sonra çeşitli adli analiz yöntemleri ile ortaya çıkarılmaktadır. Dolayısı ile bu tarz saldırılar ne gerçek zamanlı tespit edilebiliyor, ne önlenebiliyor ne de caydırılabiliyor. Bu güne kadar iç kaynaklı tehditlerin tespit edilebilmesi için geliştirilen bir çok sistem; kullanıcı davranışları ve alışkanlıklarını izleyerek bu kayıtlara göre kullanıcı profilleri oluşturmakta, oluşturulan bu kullanıcı profillerini baz alarak da kötüye kullanım tespiti yapmaya çalışmaktadır. Bu tarz bir sistemin öğrenme yeteneği ve gerçek kullanıcıyı tanımlama aşaması pratikte her zaman doğru bir şekilde gerçekleşmediğinden veya bazen bu sürecin manipüle edilebilmesinden dolayı genelde başarısızlıkla sonuçlanan bir yöntem olmaktadır. Bundan dolayı da bu tarz bir sistemin tek başına kullanılmasından pek verim alınamamaktadır. Onun için gerekli güvenlik önlemlerinin yanında destek

güç olarak sürekli monitörleme yapılabilmesine imkan tanıyan bir gözetleme/tespit sistemi de olmalıdır.

Bu çalışma kapsamında hedeflenen, var olan güvenlik mekanizmalarına destek olabilecek, saldırı tespit ve önleme yetkinliğine katkıda bulunabilecek, gizli belge içeriklerinin korunabilmesini sağlayacak bir belge takip sisteminin kurulmasıdır. Bu doğrultuda farklı bir yaklaşım/yöntem önerilmiştir. Önerilen yöntem temel olarak dokümanların açılıp açılmadığının tespit edilmesine dayanmaktadır. Bu yöntem sayesinde; açılan bir dokümanın ne kadar süre açık kaldığı, hangi ip adresinden ve coğrafi konumdan erişim sağlandığı, doküman bütünlüğüne etki edecek bir aktivitenin gerçekleşip gerçekleşmediği gibi bir çok bilgi web tabanlı olan monitörleme merkezi üzerinden takip edilebilecektir.

2.bölümde, bu konu ile ilgili daha önce yapılmış olan benzer çalışmalar incelenmiş ve detaylı analizleri yapılmıştır. 3.bölümde, konunun daha iyi anlaşılabilmesi amacıyla, çalışma kapsamında bahsi geçen veya kullanılan kavramlar hakkında genel bir bilgi verilmiştir. 4.bölümde geliştirilen sistem, sistemin yetenekleri, altyapısı, kullanılan teknolojiler, kullanımı, uygulanması ve çalışma mantığı detaylı olarak anlatılmıştır. 5.bölümde ise önerilen sistemin gerçek hayatta nerelerde ve nasıl kullanılabileceği farklı senaryolar üzerinden gösterilmiştir. Son bölüm olan 6. bölümde ise çalışmanın hedefleri değerlendirilmiş, sonuçlar yorumlanmış ve gelecekte yapılabilecek çalışmalara yönelik önerilerde bulunulmuştur.

Bölüm 2

İlgili Çalışmalar

Dokümanların açılıp açılmadığı ve nerede kim tarafından açıldığı hakkında bilgi edinebilmek amacıyla siber dünyada yapılan ilk çalışma Clifford Stoll tarafından gerçekleştirilmiştir. Stoll'un "The Cuckoo's Egg" ismiyle romanlaştırdığı bu çalışmasının amacı; Lawrence Berkeley Ulusal Laboratuvarı'na ait bilgisayar sistemlerine sızan Alman asıllı bilgisayar korsanlarının aktivitelerini tespit etmek ve asıl amaçlarının ne olduğunu saptamaktı [16]. Stoll, bu amacını gerçekleştirebilmek için bilgisayar korsanlarının dikkatini çekebilecek içerik ve isimde tuzak dokümanlar hazırlamıştır. Bu dokümanların en büyük özelliği ise açıldıkları zaman bir alarm üretmeleri yani bir geri bildirimde bulunmalarıydı. Böylelikle dokümanların açılıp açılmadığını takip edebilecekti. Stoll, geliştirdiği bu takip sistemi sayesinde Alman asıllı bilgisayar korsanlarının çaldıkları bilgi ve belgeleri Rus gizli servisi olan KGB'ye sattıklarını saptamıştır [16].

Geliştirdiği tuzak sistem(honeypot) ile iç tehdit tespiti üzerine çalışma yapan Spitzer, "bilgisayar olmayan bir tuzak sistem" olarak tanımladığı ve kredi kartı bilgileri gibi içerisinde sahte bilgiler barındıran tuzak doküman(honeytoken) kavramını ortaya atmıştır [17, 18]. Spitzer'in honeytoken sistemi ile iç tehdit tespitinin nasıl yapılabileceği üzerine başlattığı tartışmanın ilk çıktısı Yuill ve arkadaşları tarafından geliştirilen honeyfile sistemi olmuştur [19]. Gelişmiş bir dosya sunucusu olarak tanımladıkları honeyfile sistemi sayesinde kullanıcının sahip olduğu herhangi bir doküman, içerisine yerleştirilen bilgisayar kodu ile takip edilebilen bir tuzak dokümana dönüşebiliyordu. Böylece bir dokümanın açılıp açılmadığı, hangi konumda açıldığı gibi bilgiler merkezi bir sunucu üzerinden takip edilebiliyordu. Yuill ve arkadaşları ilk tuzak doküman kavramını ortaya

atan Cliff Stoll'un konseptini temel almışlar ve bu konsepti genişleterek bir adım daha ileriye taşımışlardır. Geliştirdikleri dosya sunucusu üzerinde tanımlı olan kullanıcılar istedikleri zaman istedikleri bir dokümanı tuzak dokümana dönüştürebiliyorlardı [19].

Spitzer'in tohumunu attığı, Yuill ve arkadaşlarının da geliştirdiği tuzak doküman kavramı; başında Salvatore J. Stolfo'nun bulunduğu Colombia Universitesi IDS laboratuvarının da dikkatini çekmiştir. Stolfo, başta Colombia Universitesi doktora programına kayıtlı öğrenciler olmak üzere post-doktora programı kapsamında kabul ettiği çalışma arkadaşları ile tuzak doküman kullanılarak, özellikle iç kaynaklı tehditlerin tespit edilmesi amacıyla uzun yıllara yayılan detaylı ve kapsamlı çalışmalar yapmıştırlar. Hatta yaptıkları çalışmalar ABD Savunma Bakanlığı İleri Araştırma Projeleri Ajansı (DARPA) ve ABD İç Güvenlik Bakanlığı gibi önemli kurumlar tarafından da desteklenmiştir. Bu kapsamda ilk olarak Malek Ben Salem ile birlikte iç tedit tespit sistemleri üzerine genel bir araştırma/inceleme yapmışlardır [20]. Yaptıkları bu kapsamlı çalışmalar neticesinde iç tehdit kaynaklı saldırı çeşitlerini tanımlamışlardır. Bunlar:

- Verilerin izinsiz dışarı çıkarılması/sızdırılması ve çoğaltılması/kopyalanması,
- Verilerin izinsiz değiştirilmesi ve veri bütünlüğünün bozulması,
- Kritik varlıkların silinmesi veya tahrib edilmesi,
- Ağ trafiğinin dinlenmesi,
- Diğer kullanıcıların taklit edilmesi,
- Sosyal mühendislik saldırıları

Aynı çalışma kapsamında bu tarz saldırıların genelde arkada bir takım izler bıraktıklarını ve bu izlerden yola çıkılarak adli analiz yöntemleri ile gerçekleştirilen saldırıların tespit edilebildiğini vurgulamışlardır. Fakat bu durumun da saldırıyı gerçekleştiren saldırganın yeteneklerine bağlı olarak değişim gösterdiğinin altını çizmiştirler. Çünkü iyi bir saldırganın genelde arkasında bıraktığı izleri temizlemeye çalıştığını belirtmişlerdir. Bundan dolayı da hem sunucu hem de ağ tabanlı sensörlerin kullanılarak kullanıcı profillerinin çıkartılması ve bunlarla beraber sistemin o an ki durumunun canlı olarak izlenebilmesini sağlayan bir monitörleme sisteminin olması gerektiğini savunmuşlardır.

Malek Ben Salem ile iç kaynaklı saldırılar üzerine yaptıkları araştırmalardan elde ettikleri veriler neticesinde Stolfo ve arkadaşları, iç kaynaklı saldırganlara karşı tuzak doküman kullanılması ve bu dokümanlara erişim kayıtlarının anlık monitörlenmesi gerektiğini düşünerek bir sistem geliştirmeye yönelmişlerdir. Brian M. Bowen'in de dahil olduğu bu çalışma kapsamında saldırgan profilleri sınıflandırılmış ve bu sınıflandırma derecelerine göre bir tuzak dokümanın sahip olması gereken özellikler tespit edilmeye çalışmışlardır [21]. Yaptıkları çalışmalar sonrası bir tuzak dokümanın sahip olması gereken özellikleri şöyle sıralamışlardır:

- *İnanılırlık*: Bir tuzak dokümanın, incelendiği zaman gerçek ve güvenilir olmasını ifade etmektedir. Bir tuzak dokümanın sahip olması gereken temel özelliklerin biridir. Olası bir sızma durumunda bilgisayar korsanı bu tuzak doküman ile gerçek dokümanlar arasındaki farkı anlayamamalıdır. Örneğin tuzak bir vergi belgesi, gerçeği ile tamamen aynı alanlara sahip olmalıdır. İçerisinde bilgiler ele geçirilmesi durumunda zarar veremeyecek gerçek bilgiler ile doldurulmalıdır. Burada akla gelen sorulardan biri; tuzak doküman isminin mi yoksa tuzak doküman içeriğinin mi inanılırlığı üzerinde durulmalıdır? Bilgisayar korsanı dokümanı açtığı zaman dokümanın açıldığını bildiren alarm mekanizması zaten tetikleneceği için tuzağa düşürmek için öncelikle tuzak doküman isminin inandırıcı olması gerekmektedir.

- *Cezbedicilik/Caziplik*: İdeal bir tuzak doküman inandırıcı olmasının yanında birde saldırganların dikkatini çekecek şekilde yapılandırılmalıdır. Bu özellik temelde saldırganın motivasyonuna bağlıdır. Şöyle ki; eğer saldırgan finansal motivasyona sahip ise onu cezbedecek dokümanlar parasal verilerle alakalı olan dokümanlardır, eğer motivasyonu rakip firmaya bilgi sızdırmaksa o zaman da şirketin gizli bilgileri ile alakalı dokümanlar saldırganı cezbedir.

- *Dikkat çekicilik/Barizlik*: Dikkat çekicilik bir önceki maddede bahsedilen cezbedicilik ile çok yakın bir ilişkiye sahiptir. İkiside bir saldırganın dokümanı açma olasılığına neredeyse aynı oranda etki etmektedir. Cezbedicilik, dokümanı açması için saldırganı merak ettirmeyi amaçlarken; dikkat çekicilik, saldırganın tuzak dokümana en kolay şekilde ulaşabilmesini amaçlamaktadır. Örneğin, sisteme yetkisiz erişim sağlayan bir bilgisayar korsanının dikkatini çekmek ve onu tuzağa düşürmek için tuzak dokümanlar Masaüstü, İndirilenler veya Belgelerim dizinlerine yerleştirilebilir.

- *Tespit edilebilirlik*: Yukarıda bahsi geçen tuzak doküman özelliklerinin hepsi tuzak doküman ve saldırgan arasındaki ilişki temelli iken bu özellik tuzak doküman ve doküman sahibi arasındaki ilişki ile ilgilidir. Bu özellik ile ideal bir tuzak dokümanın, açıldığının tespit edilebilmesi için açıldığı zaman merkezi sunucuya alarm göndermesi gerektiğini savunmaktadır.

- *Değişkenlik*: Bu özellikte, sisteme tespit edilmeden sızmayı başarmış bir bilgisayar korsanının, bir kaç tuzak dokümanı inceledikten sonra geri kalan diğer dokümanların hangilerinin tuzak hangilerinin de gerçek doküman olduğunu ayırt edememesi gerektiği belirtilmektedir. Yani bu özellik, oluşturulan tuzak dokümanların sabit bir standardı olmaması gerektiğini ve daha önce oluşturulmuş olan tuzak dokümanların incelenerek ileride oluşturulacak olan tuzak dokümanların gerçek dokümanlardan ayırt edilememesi gerektiğini belirtmektedir.

- *Normal akışı engellememe*: Bu özellikte, tuzak dokümanların meşru kullanıcıların normal aktivitelerini engellememesi gerektiği belirtilmektedir.

- *Ayırt edilebilirlik*: Bu özellik bir nevi inanılır olma özelliğinin zıttı gibi düşünülebilir. Bu özellikte, tuzak dokümanın potansiyel saldırganlara karşı gerçek bir dokümanmış gibi görünmesi ve ayırt edilememesi gerektiğini, gerçek kullanıcılar tarafından da bariz bir şekilde ayırt edilebiliyor olması gerektiğini belirtmektedir.

Stolfo ve arkadaşları, yukarıda açıklanan bu özelliklere göre PDF ve Word dosya formatında otomatik olarak tuzak doküman üretebilen, web tabanlı "Decoy Document Distribution" ismini verdikleri bir servis de geliştirmişlerdir. Otomatik olarak üretilen bu tuzak dokümanların içeriği ise banka hesap bilgileri, kredi kartı bigileri ve email hesap bilgileri gibi sahte bilgiler ile doldurularak tuzak dokümanların ayırt edilebilirliğini zorlaştırmışlardır. Tuzak dokümanların içerisine doldurdukları bu sahte bilgilerin kullanılıp kullanılmadığını da sunucu bazlı kullandıkları sensörler aracılığı ile gözlemlemişlerdir [21].

Günümüzde bir çok kötücül yazılımın, antivirüs yazılımlarını ve sunucu tabanlı sensörleri kolay bir şekilde atlatabiliyor duruma gelmiş olması, anlık monitörleme teknolojilerinin kullanılması gerektiğinin en büyük göstergesidir. Bunun farkına varan ilk iki çalışmada ayrı ayrı yer alan Bowen ve Salem bu sefer monitörleme teknolojilerinin iç tehditlere

karşı hafifletici etkileri üzerine bir çalışma yapmak üzere Stolfo önderliğinde bir araya gelmişlerdir [22].

Başta da belirtildiği üzere bu alanda detaylı çalışmalar yapmış olan Stolfo ve arkadaşları, bir tuzak dokümanın sahip olması gereken özelliklerin tespiti üzerine şimdiye kadarki yaptıkları çalışmaların doğruluğunu ve tanımladıkları tuzak doküman özelliklerinin etki değerlerini gösterebilmek amacıyla çeşitli deneyler yapmışlardır [23]. Örneğin; tuzak doküman sayısının ve tuzak dokümanların dosya sunucusunda bulundukları konumlarının, meşru kullanıcıların normal aktivitelerine olan etkisi üzerine bir deney yapmışlardır. Bu ve benzeri deneyler sonucunda elde ettikleri verilere göre de bir kaç öneride bulunmuşlardır. Örneğin; eğer amaçlanan kritik bilgilerin sızdırılmasına yönelik olan iç tehditlere karşı koymak ise inanılırlık çok önemli bir özellik olmaktadır. Fakat amaçlanan yetkili bir kullanıcının haklarının ele geçirilmesi sonucu taklit edilmesi ile gerçekleştirilen iç kaynaklı saldırıları önlemek ise inanılırlık çok da önemli olmamaktadır. Çünkü içerideki yetkili bir kişi tuzak dokümanın içeriğinin sahte bir veri olup olmadığını ayırt edebilir ve bu sahte bilgileri kullanmaz. Fakat dış kaynaklı olan ve taklit yöntemi ile saldırı gerçekleştiren bir başkası bu ayrımı kolay kolay yapamaz ve sahte bilgileri kullanabilir. Bundan dolayı bu amaç doğrultusunda bir tuzak doküman kullanılmak istenildiğinde sahip olması gereken özelliklerinin öncelik sırası şöyle olmalıdır:

1. Tespit edilebilirlik
2. Barizlik
3. Çekicilik
4. Normal akışı etkilememe ve Türevlenebilirlik
5. İnanılırlık

Bu güne kadar dokümanların korunumu genellikle IDS, IPS, güvenlik duvarı ve VPN gibi çözümler kullanarak, dokümanların bulundukları dosya konumlarına olan erişim izinlerinin kontrol edilmesi ile sağlanmıştır. Fakat bahsi geçen teknolojik çözümler ne kadar güçlü olsa da özellikle hızla değişen ve gelişen ileri seviye siber tehditler (APT) karşısında yetersiz kalmaktadır. Tuzak dokümanlar aracılığıyla kurulan alarm mekanizmasının etkin bir şekilde kullanılması ile gerçek dokümanların korunmasına katkı sağlamak amacıyla yukarıda bahsi geçen çalışmalar gerçekleştirilmiştir. Yapılan bu çalışmalar neticesinde

geliştirilen sistemlerin, tuzak dokümanların açılıp açılmadığına dair herhangi bir alarm bilgisi alınamadığında pek bir işlevselliği kalmamaktadır. Peki bahsi geçen güvenlik çözümlerinin yetersiz kaldığı ve tuzak dokümanlardan da alarm bilgisinin alınamadığı yani mevcut güvenlik sistemleri ile herhangi bir tehdit tespitinde bulunulamaz iken gerçek dokümanların korunumu nasıl sağlanacaktır? Bu noktada yukarıda bahsi geçen çalışmalarda, sadece tuzak dokümanların oluşturulması ve bunların açılması ile tetiklenen alarmların takibi ile saldırı tespitinde bulunulmaya çalışılmıştır. Fakat bahsedildiği gibi bu alarm mekanizması çalışmadığında çalınma ihtimali olan gerçek dokümanların içeriğini korumaya yönelik uygulama seviyesinde gerçekleştirilmiş olan bir çalışma bulunmamaktadır. Bundan dolayı, bu çalışmada iç tehdit minimize edilirken orijinal doküman içeriğinin korunabilmesine imkan tanıyan bir doküman takip sistemi önerilmiştir.

Bölüm 3

İlgili Temel Bilgiler

Bu bölümde, gerçekleştirilen çalışma kapsamında kullanılan kavramlar ayrıntılı olarak açıklanmıştır.

3.1 İç Tehdit (Insider Threat)

Bir toplumda yaşam tarzı ve yaptıkları çalışmalar ile diğer insanlara faydalı olmayı amaçlayan insanlar bulunduğu gibi bunun tam tersi bir motivasyon ile çalışma yapan insanlar da bulunmaktadır. Bu bakış açısını bilgi güvenliği/bilişim teknolojileri alanında ele aldığımızda da durum pek değişmemektedir. Örneğin; bir tarafta insanların, şirketlerin ve kurum/kuruluşların işlerini kolaylaştırmak amacı ile yazılım geliştirilirken, diğer tarafta zarar vermek amacı ile bilgisayar programı yazılmaktadır. Bu tarz kötü niyetli çalışmalar ve saldırılar genelde karşı tarafa kayıp ve hasar vermeye yönelik olmaktadır. Kötü niyetli gerçekleştirilen bu saldırılar iki ana başlık altında incelenebilir. Bunlar; içeriden bir kullanıcının direkt veya dolaylı yoldan dahil olduğu iç kaynaklı (iç tehdit) saldırılar ve sistem açıklıklarının sömürülmesi ile gerçekleştirilen dış kaynaklı (dış tehdit) saldırılardır.

Bilgi güvenliği alanında çok önemli bir yere sahip olan iç tehdit, organizasyon bünyesinde çalışan veya daha önceden çalışmış olan kişilerin oluşturduğu tehditler bütünüdür. İç tehditler kendi arasında iki sınıfa (traitors, masqueraders) ayrılmaktadır. Bunlardan ilki; organizasyon içinde çalışan, bilgi kaynaklarına ve sistemlere erişim hakkı olan fakat

bunları gizlilik, bütünlük ve kullanılabilirlik ilkelerine aykırı olacak şekilde kötüye kullanılmasıdır (traitors). Bu tarz bir saldırıya verilebilecek en iyi örnek veritabanı yöneticisinin, eriştiği verileri çıkar amacıyla başka bir firmaya satmasıdır. Bu şekilde yaşanmış bir örnek ise Greek Vodafone olayıdır [24]. İkinci sınıf ise bir kişinin yetki sahibi başka bir kişiye ait erişim bilgilerini elde ederek, normal şartlarda erişememesi gereken bilgilere erişerek kötü niyetli bir aktivite gerçekleştirmesidir (masqueraders). Bu tarz bir saldırıya verilebilecek en iyi örnek ise veritabanı yöneticisi olmayan ve normalde veritabanına erişim hakkı bulunmayan birisinin, veritabanına erişim hakkı bulunan bir kullanıcının bilgilerini bir şekilde elde ederek, kayıtlı verileri elde etmesi ve bunları çıkarları doğrultusunda kullanmasıdır.

3.2 Dış Tehdit (External Threat)

Kurum dışı ağlardan(internet veya diğer ağlar), özellikle yetkisiz erişim elde etmek amacıyla çeşitli açıklıkların ve zafiyetlerin istismar edilmesi ile gerçekleştirilen saldırılar dış tehdit olarak adlandırılmaktadır. Bu tip saldırılarla yetkisiz erişim elde etme girişimleri dışında yüksek miktarda ağ trafiği üretilerek (DDOS) kurumun sağladığı hizmetlerin kısmen/tamamen durdurulması amaçlanabilir. Özellikle aktivist grupların(Anonymous, Honker Union, LulzSec, vb.) gerçekleştirdiği saldırılar dış tehdide verilebilecek örnekler arasında gösterilebilir.

3.3 Tuzak Sistem (Honeypot)

Üst seviye bir güvenlik altyapısı kurulmuş olsa dahi amacına ulaşmak için elinden geleni yapacak olan saldırganı önlemek çoğu zaman mümkün olamamaktadır. Bu durum düşünüldüğü zaman klasik güvenlik önlemlerine destek olabilecek, saldırı yüzeyini ve saldırılara karşı koyma derinliğini arttıracak alternatif çözümlerin başında ilk olarak akla gelen tuzak sistemlerdir. Tuzak sistemler, istenilen/bilinen servisleri ve açıklıkları/zafiyetleri simüle edebilen dolayısı ile saldırganların dikkatini çeken/cezbeden sahte makinelerdir/sistemlerdir. Tuzak sistemler genelde bir ağın parçasıymış gibi görünen, saldırmak için sebep olabilecek bilgi veya değer taşıyan fakat gerçek bilgisayar ağından izole edilmiş, üzerinde gerçekleşen her hareketin özellikle kaydedildiği ve izlenebildiği

bir bilgisayar/sunucu olabilir. Tuzak sistem üzerinde gerçekleşen her hareket kayıt altına alındığı için, saldırganların nasıl ve hangi yöntemle sızdıkları, sızdıktan sonra neler yaptıkları gibi daha bir çok bilgi toplanarak bir yandan saldırgan sahte sistemler ile oyalanırken diğer yandan da saldırgan profili çıkartılabilir. Dolayısı ile tuzak sistemlerin kullanımı ile yetkisiz ve kötü amaçlı erişimler tespit edilebilir. Bu tarz erişimlerin tespit edilmesi ve kısa sürede engellenmesi ile de meydana gelebilecek olan zararlar minimize edilebilir.

3.4 Tuzak Doküman (Honeyfile)

Tuzak doküman, dosya sunucusunda yer alan ve açıldığı zaman takip sunucusuna bir alarm gönderen dokümandır. Gönderilen bu alarmlar sayesinde dokümanın nerede açıldığı, hangi sayfalarının okunduğu, kaç dakika açık kaldığı gibi bilgiler elde edilebilmektedir. Elde edilen bu bilgilerin yorumlanması ile tespit edilemeyen yetkisiz erişimler tespit edilebilir ve saldırı yüzeyi arttığı, savunma derinliği arttığı için de saldırıların etkisi azaltılabilir.

3.5 Saldırı Tespit Sistemleri (IDS, IPS)

Hayatımıza akademik amaçlı bir araştırma ağı olarak giren internet, günümüzde önemli toplumsal dönüşümlere altyapı sağlar duruma gelmiştir. İnternetin bu kadar kapsamlı ve etkili kullanılabileceği öngörülemediğinden internet ortamındaki güvenlik uzun süre ikinci planda kalmış ve bu konuda yeteri kadar çalışma yapılmamıştır. Fakat internet kullanım oranının artması, internete bağlı kurum sayısının artması, internet ortamında yapılabilen işlerin çeşitliliğinin artması gibi daha sayılabilecek bir çok nedenden ötürü güvenlik konusu ister istemez ciddi bir problem haline gelmiştir.

Genel olarak yapılan saldırıların büyük bir çoğunluğu kullanılan sistemlerin zaafları ve/veya açıklıklarından faydalanılarak gerçekleştirilmektedir. Bu tür saldırıları engellemenin iki yolu vardır. Birincisi tamamen güvenli bir sistem ve ortam oluşturmak, ikincisi ise en kısa zamanda saldırıların tespit edilip gerekli önlemlerin alınmasının sağlanmasıdır. İlk yöntemle saldırıların tamamının önüne geçilmesi mümkün olamıyor. Onun için bir sistemin güvenliği, sistem güvenlik sorumluları tarafından, rutin kontrolleri yapılmak kaydı

ile saldırı gelene kadar bekleme pozisyonunda kalarak, saldırı geldiğinde de olabildiğince hızlı bir şekilde saldırıyı tespit edip gerekli önlemi alabilmeyi mümkün kılacak şekilde tasarlanmalıdır. İşte bu aşamada da devreye saldırı tespit sistemleri girmektedir.

En genel anlamıyla, saldırı tespiti işini yapmak için geliştirilen sistemlere "saldırı tespit sistemleri" denilmektedir. 1980 yılında James Anderson'ın yaptığı tanımdan [25] günümüze kadar yapılan araştırmalar ve çalışmalar neticesinde saldırı tespit sistemleri için farklı tanımlar yapılmıştır. Bu tanımlar yanlış olmamakla birlikte sadece günümüzdeki saldırı tespit sistemleri tanımının yanında biraz eksik kalmaktadır. Örneğin yapılan tanımlardan bazıları şöyledir:

- Bilgisayar sistemlerine yapılan atakları ve kötüye kullanımları belirlemek için tasarlanmış sistemlerdir,
- Tercihen gerçek zamanlı olarak, bilgisayar sistemlerinin yetkisiz ve kötüye kullanımı ve suistimalini tespit etmek için kullanılırlar,
- Kullanım alanı ve türüne bağlı olarak saldırıyı engelleyebilen veya saldırıyı durdurma girişiminde bulunmayan, olası güvenlik ihlali durumlarında sistem güvenlik çalışanlarına uyarı mesajı veren sistemlerdir,
- Bilgisayar sistemlerinin kaynaklarına veya verilerine yetkisiz erişimleri tespit edebilen sistemlerdir,
- Bilgisayar ortamındaki "hırsız alarm"larıdır.

Günümüzde kullanılan tanımı ise tüm bu yapılan tanımları kapsamaktadır. Saldırı tespit sistemleri, bilginin elektronik ortamlarda taşınırken, işlenirken veya depolanırken başına gelebilecek tehlike ve tehditlerin ortadan kaldırılması veya bunlara karşı tedbir alınması amacıyla, bilgiye yetkisiz erişim ve bilginin kötüye kullanılması gibi internet veya yerel ağdan gelebilecek çeşitli paket ve verilerden oluşan girişimleri tespit edebilme, bu tespitleri sms, e-posta veya Simple Network Management Protocol (SNMP) mesajları ile sistem güvenliğinden sorumlu kişilere iletebilme ve gerektiğinde paketi/erişimi düşürebilme özelliğine sahip yazılımsal ve/veya donanımsal güvenlik araçları olarak tanımlanabilir.

Saldırı Tespit Sistemleri, internet dünyasının gelişim sürecinde özellikle tüm dünyada kullanılan web trafiğinin artması ve de web sayfalarının popüler hale gelmesi ile birlikte

kişisel ya da tüzel sayfalara yapılan saldırılar sonucu ihtiyaç duyulan en önemli konulardan biri haline gelmiştir. Bununla birlikte kurum ya da kuruluşların sahip oldukları ve tüm dünyaya açık tuttukları Mail, DNS ve Web gibi sunucularının benzeri saldırılara maruz kalabilecekleri ihtimali yine saldırı tespit sistemlerini internet güvenliği alanının vazgeçilmez bir parçası haline getirmiştir. Yine kurumların sahip oldukları çalışanların kendi kurumlarındaki kritik değer taşıyan yapılara/verilere saldırabilme/zarar verme ihtimalleri düşünülünce iç ağın ya da tek tek kritik sunucuların kontrol altında tutulma gerekliliği de saldırı tespit sistemlerinin kullanımını kaçınılmaz kılmıştır [26].

3.6 Bulut Bilişim (Cloud Computing)

Bulut bilişim; yazılım, donanım ve bakım maliyetlerini arttırmadan birçok farklı teknolojiyi mümkün olan en iyi performans değerlerinde kullanabilmek olarak tanımlanabilir. Bulut bilişim; daha az sermaye yatırımı gerektiren, bakım ve destek maliyetlerinin %90 oranında azalmasını sağlayan, isteğe/ihtiyaca göre ölçeklenebilen güçlü ve kesintisiz bir altyapı sunan, kullanımı ve kurulumu kolay olan, temeli internet protokollerine dayanan bir servistir/hizmettir. Daha basit hali ile kullanıcının yerel konumunda herhangi bir donanım veya yazılım gereksinimine ihtiyaç duymadan tüm işlemlerinini internet üzerinden bağlantı kurabildiği uzak bir sunucu üzerinde gerçekleştirebilmesine imkan tanıyan bir hizmettir. Buradaki uzak sunucu olarak ifade edilen kavram buluttur. Dolayısı ile bulut bilişim aslında daha önceden de var olan ve kullanılan fakat yeni özellikler kazandırılmış olan bir teknolojidir.

Bilgisayar kuramcıları tarafından internetin geleceğinin bulut bilişimden geçtiği iddia edilmektedir. Buna iddiaya göre gelecekte, bilgisayar hard disklerinin yerine çevrim içi bulutların kullanılacağı öngörüsü hakimdir [27]. Bu bakış açısına sahip olan Google, Amazon, Microsoft ve IBM gibi bir çok büyük firma da bulut bilişim alanında devasa yatırımlar yapmaktadır.

Bölüm 4

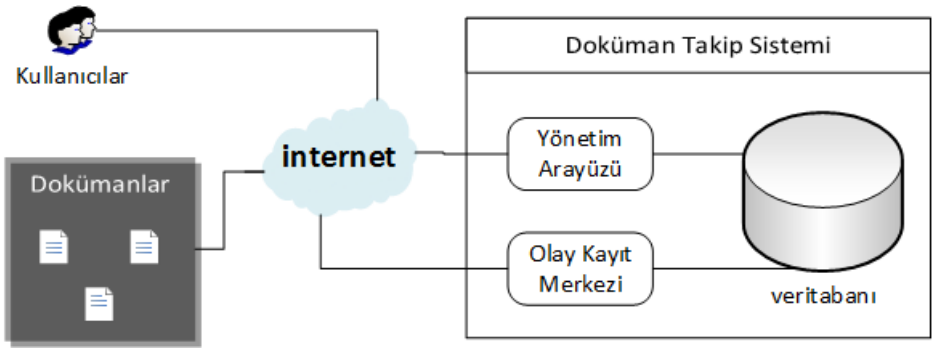
Önerilen Sistem - DoxTracker

Bu bölümde DoxTracker sisteminin genel yapısı, bileşenleri, altyapısı, geliştirme sırasında kullanılan teknolojiler ve genel kullanım akış şeması gösterilerek önerilen takip mekanizmasının dokümanlara nasıl uygulandığı hakkında detaylı bilgi verilmiştir.

4.1 Genel Yapı

Çalışma kapsamında önerilen sistem, temel amaçları karşılayacak şekilde bir prototip olarak gerçeklenmiş ve testleri başarılı bir şekilde yapılmıştır. Bu bölümde de geliştirilen prototibin temel bileşenleri tanıtılarak genel yapı açıklanmıştır.

DoxTracker kullanılarak herhangi bir doküman tuzak dokümana dönüştürülebilir veya şifrelenerek içeriği koruma altına alınabilir. Yeni doküman oluşturma, doküman silme, doküman olay geçmişini görüntüleme gibi tüm işlemler Yönetim Arayüzü aracılığı ile basit bir şekilde gerçekleştirilebilmektedir. Dokümanlardan gelen alarmların sistem tarafında karşılanması, kontrol edilmesi ve veritabanına kaydedilmesi gibi işlemler ise Olay Kayıt Merkezi tarafından gerçekleştirilmektedir. Bir doküman açıldığı zaman Olay Kayıt Merkezine, dokümanın açıldığına dair bir alarm bilgisi gönderilir. Bu alarm ile açılan dokümanın kim tarafından ve nereden açıldığı gibi bilgiler Olay Kayıt Merkezi tarafından veritabanına aktarılır. Kullanım arayüzü oldukça basit olan sistemin genel görünümü Şekil 4.1'dedir.



ŞEKİL 4.1: DoxTracker genel yapısı

- *Kullanıcılar*: Sistemde kayıtlı olan kullanıcıları tanımlamaktadır. Bu kullanıcılar internet erişimi olan herhangi bir cihaz aracılığı ile yönetim arayüzüne bağlanarak dokümanlarını görüntüleyebilir, düzenleyebilir ve takip işlemleri gözlemleyebilir.

- *Dokümanlar*: DoxTracker tarafından takip edilmekte veya korunmakta olan dokümanları tanımlamaktadır.

- *Yönetim Arayüzü*: Dokümanların listelenmesi, yeni doküman takip sürecinin başlatılması, harita üzerinde dokümanların açıldıkları konumların görüntülenmesi gibi bir çok faklı işlevin kolay bir şekilde yönetilebilmesi için tasarlanmış web tabanlı kullanıcı dostu bir arayüzdür.

- *Olay Kayıt Merkezi*: Kullanıcıların dahil olmadığı ve herhangi bir şekilde erişemediği bir kısımdır. DoxTracker tarafından takip edilmekte olan dokümanlar açıldıkları zaman merkezi sunucu ile haberleşmektedir. Olay Kayıt Merkezi, DoxTracker sunucusunda bulunan ve sadece dokümanlardan gelen talepleri karşılayarak gerekli işlemleri gerçekleştiren yapıdır.

- *Veritabanı*: Bir dokümanın ne zaman kim tarafından oluşturulduğu, ne zaman açıldığı, hangi coğrafi konumdan açıldığı, kim tarafından açıldığı gibi daha bir çok verinin bulunduğu depolama birimidir.

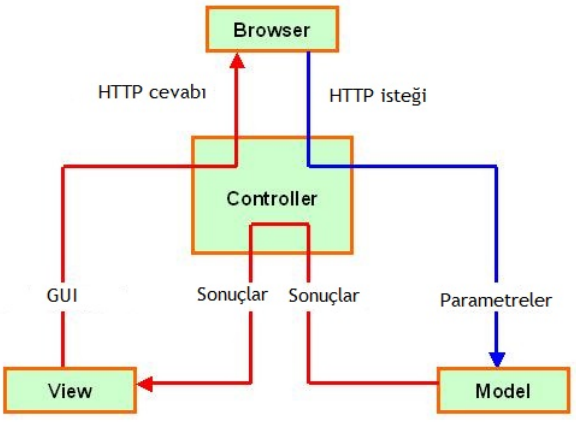
4.2 Kullanılan Teknolojiler ve Altyapı

Bu bölümde DoxTracker sisteminin geliştirilmesi sırasında yaralanılan teknolojiler açıklanmıştır.

- *Ubuntu*: Sistemin üzerinde çalışacağı sunucu olarak Ubuntu seçilmiştir. Ubuntu, Güney Afrika'lı girişimci Mark Shuttleworth ve Canonical Ltd. sponsorluğunda geliştirilen ve dünyanın en yaygın kullanılan Linux tabanlı ücretsiz bir işletim sistemidir. Ubuntu, herkesin yayınlamakta, kopyalamakta ve kodlarını değiştirip geliştirebilmekte özgür olduğu yazılımlardan oluşmaktadır. Özellikle düzenli güncellemeler ile desteklenmesi, yönetim kolaylığı ve gönüllü geliştiricilerin destekleri ile sorunların hızlı çözülebiliyor olmasından dolayı Ubuntu sunucusunun kullanılması tercih edilmiştir.

- *MVC (Model-View-Controller)*: Model-View-Controller kelimelerinin baş harflerinden oluşan ve yazılım mühendisliğinde kulllanılan bir mimari desendir. İlk olarak 1979 yılında Trygve Reenskaug tarafından tanımlanmıştır. MVC bir yazılım dili değildir, bir mimari desendir. MVC, temelde veri ve gösterimin soyutlanması esasına dayanmaktadır. Böylelikle veriler (model) ve kullanıcı arayüzü (view) birbirinden bağımsız olarak düzenlenebilmektedir. Model ve View arasındaki iletişim ise Controller bileşeni ile sağlanmaktadır.

  MVC mimarisinde uygulama üç farklı yapıya ayrılır. Model, View ve Controller. Model en genel tabir ile veritabanı işlemlerinin gerçekleştiği bölümdür. HTML, Javascript, CSS gibi uygulama arayüzüne ait ne varsa View kapsamına girmektedir. Controller ise iş akışının gerçekleştiği, arayüzden gelen kullanıcı etkileşimlerinin değerlendirildiği, işlendiği, gerekli metodların çalıştırıldığı, değişkenlerin ve nesnelerin oluşturulduğu, gerekirse Model ile View bölümleri arasında iletişimin sağlandığı yer burasıdır.



ŞEKİL 4.2: MVC çalışma mantığı

Şekil 4.2'de de görüleceği üzere ilk önce internet tarayıcısı üzerinden bir istekte bulunulur. Bu isteği Controller karşılar ve ilgili isteği uygun formatta Model'e iletir. Model veritabanı bağlantıları ve verilerin çekilmesi gibi işlemleri gerçekleştirerek gerekli veriyi hazırlar ve Controller'a verir. Controller gelen verileri View'e iletir ve View'de grafiksel arayüz ile kullanıcıya sunmak üzere tekrar internet tarayıcısına gönderir.

- *Nesne Yönelimli Programlama*: Günümüzde pek çok çağdaş programlama dili tarafından desteklenen, yoğun bir şekilde kullanılan ve temelleri 1990'lı yıllara dayanan bir programlama tekniğidir. Nesne Yönelimli Programlama'da, programlama ortamındaki her şey bir nesne olarak kabul edilir. Bu nesneler kendi içerisinde veri işleyebilir ve birbirleri ile veri alış-verişinde bulunabilirler.

- *Framework*: Türkçede uygulama çatısı veya uygulama iskeleti olarak ifade edilen framework; veritabanı bağlantıları, form işlemleri, üye kayıt işlemleri, resim işleme sınıfları gibi daha sayılabilecek pek çok sınıf ve eklentinin toplu olarak geliştiricilere sunulduğu bir altyapıdır. Yapılan çalışmaların altyapısı yerine ana fikre odaklanma imkanı tanıdığı için zamandan tasarruf sağlar. Bununla beraber framework kullanımı, daha düzenli ve güvenli yapılar oluşturulmasına da imkan tanımaktadır.

- *HTML5*: HTML 80'li yıllarda bilgi ve dokümanların paylaşılabilmesi için geliştirilen bir metin işaretleme standardıdır. Temel olarak, sayfa içerisindeki metin ve resimlerin nasıl yerleşeceklerini etiketler ile belirleyen bir sistemdir. HTML kodları kendi başına çalışabilen bir yapıda değildir, web tarayıcıları yazılan HTML kodlarını yorumlar. Bundan dolayı aynı HTML kodları farklı web tarayıcılarında farklı sonuçlar verebilir. HTML5 ise HTML standardının beşinci sürümüdür. HTML5 ile birlikte HTML, sadece bir işaretleme dili olmaktan çıkmış çeşitli media etiketleri ile çok daha güçlü bir yapıya bürünmüştür.

- *PHP*: Kısaca web tabanlı, HTML içerisine gömülebilen, sunucu taraflı bir programlama dili olan PHP'nin temelleri 1995 yılında Rasmus Lerdorf tarafından atılmıştır. Günümüzde halen daha PHP Topluluğu tarafından geliştirilmesine devam edilmektedir. Çok geniş bir kitle tarafından kullanılan PHP, açık kaynaklı ve ücretsizdir.

- *Laravel*: MVC yapısında web uygulamaları geliştirebilmek için tasarlanmış söz dizimi basit, esnek, nesne yönelimli, ücretsiz ve açık kaynak kodlu olup web dünyasında en hızlı büyüyen PHP web uygulama framework'üdür. İlk olarak Taylor Otwell

tarafından tasarlanmıştır. Daha sonra GitHub sitesinde MIT lisansı altında yayınlanmıştır. Laravel; bir geliştiricinin bir kaç işlem barındıran çok küçük sitelerden çok büyük kurumsal projelere kadar her türlü web uygulamasını oluşturmak için gereken esnekliği sağlayan felsefe ile tasarlanmıştır. Bununla beraber PHP ve nesne yönelimli programlamanın tüm nimetlerinden yararlanılmasını sağlayan gelişmiş bir çok özellik ve yapıyı üzerinde barındırmaktadır. 2014 ve 2015 yıllarında SitePoint tarafından yapılan en popüler PHP web uygulama framework'u anketlerinde hem bireysel hem de kurumsal bazda en çok tercih edilen framework Laravel olmuştur [28, 29].

- *CSS3*: CSS, web sayfalarının ve sayfalarda bulunan nesnelerin biçimsel özelliklerinin düzenlenebilmesine imkan tanıyan bir web teknolojisidir. Örneğin bir paragrafın ilk satırının rengi, bir resmin boyutu gibi HTML ile yazılan kodların görsel açıdan daha zengin bir hale getirilmesini sağlar. CSS3 ise CSS'in önceki versiyonları üzerine geliştirilmiş üçüncü ve şu an en güncel olan versiyondur. CSS3 ile birlikte web siteleri görsel açıdan çok daha güçlü bir konuma erişmiştir.

- *Bootstrap*: 2010 yılında Twitter'ın arayüz geliştirici ekibi tarafından sunulan açık kaynak kodlu ve ücretsiz olan bir CSS frameworküdür. Yapısında, bir web sitesinde ihtiyaç duyulabilecek tüm elementlere (form öğeleri, tablolar, butonlar, vb.) ait CSS, Javascript ve görselleri farklı çözünürlükteki cihazlarda dahi bozulmadan gösterebilecek şekilde bulundurmaktadır. Bundan dolayı web uygulamaları ve web sitelerinin front-end geliştirmelerinin çok daha hızlı bir şekilde yapılmasına katkıda bulunmaktadır.

- *JavaScript*: Nesne tabanlı olup web sayfalarına dinamizm (resmin üzerine gelince büyümesi, web sayfasının yenilenmeden içeriğin değiştirilmesi, vb.) katan, 1995 yılında Netscape firması tarafından C programlama dilinin web tarayıcılara uyarlanması ile geliştirilmiş bir script dilidir. Bir standardizasyon firması olan ECMA tarafından ECMA-262 kodu ile ECMAScript ismi altında standartlaştırılmış bir dildir. HTML kodları içerisine <script>...</script> etiketleri kullanılarak yazılmaktadır. Bundan dolayı istemci tarafında yazılan kodların yorumlanabilmesi için bir web tarayıcısına ihtiyaç vardır. İlk geliştirildiği zamanlar sadece istemci tarafında çalışan Javascript, Node.js gibi platformlar sayesinde artık sunucu tarafında da başarılı bir şekilde kullanılabilmektedir.

- *JQuery*: HTML5 ve CSS3 desteği olan daha az kod yazarak daha fazla işin daha hızlı ve performanslı bir şekilde yapılabilmesine imkan tanıyan bir Javascript kütüphanesidir. Var olan Javascript kütüphanelerinin karmaşıklığından dolayı daha sade ve basit bir kütüphane geliştirme amacıyla yola çıkan John Resig 2006 yılında JQuery'i duyurmuştur. Şu anda kalabalık bir ekip tarafından açık kaynak kodlu olarak geliştirilmesine devam edilmektedir. JQuery yoğun olarak animasyonlarda kullanılır. Bir kaç küçük kod ile animasyonlar, slider efektleri ve gizleme/gösterme gibi daha sayılabilecek bir çok efekt JQuery ile basit bir şekilde yapılabilir. Küçük dosya boyutuna rağmen işlevsellik olarak neredeyse sınır tanımayan JQuery, Facebook ve WordPress gibi dünyanın en büyük hizmet sağlayıcıları tarafından kullanılmaktadır.

4.3 DokTracker'ın Uygulanması

Bu bölümde takip mekanizması için kullanılan yöntemler ve bunların nasıl uygulandığı anlatılmıştır.

Önerilen sistemin öncelikli ve temel amacı bir dokümanın açılıp açılmadığının tespit edilerek takip edilebilmesinin sağlanmasıdır. Bu takip mekanizmasının verimli bir şekilde çalışabilmesi ve sistemin kullanılabilirliği açısından doküman üzerinde gerçekleştirilecek işlemler esnasında dikkat edilmesi gereken iki önemli nokta vardır. Bunlar:

- Alarm mekanizması kullanıcıdan herhangi bir aksiyon almadan yani sadece dokümanının açılması ile tetiklenebilmelidir,
- Alarm mekanizmasının dokümana entegrasyonu sırasında dokümanın orijinal yapısına herhangi bir zarar verilmemelidir.

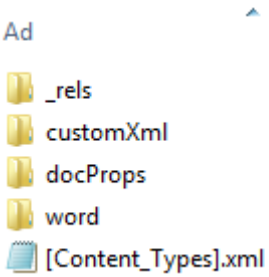
Yukarıda bahsedilen ve kritik öneme sahip olan bu noktaların uygulanması aşamasında gerçekleşecek herhangi bir hata, dokümanın orijinal yapısını bozacağından dolayı dokümanın açılmamasına ve takip edilememesine sebeb olacaktır. Bundan dolayı DoxTracker ilk versiyonu itibari ile en fazla kullanılan iki dosya formatı olan MS Word ve PDF dokümanları üzerine yoğunlaşılmış ve özellikle bu noktalarda hata yapılmaması adına gerekli araştırmalar detaylı bir şekilde yapılmıştır. Yapılan araştırmalar ve analizler neticesinde

de bu iki doküman formatının yapısı ve çalışma mantığı olarak birbirlerinden tamamiyle farklı olduğu saptanmıştır. Bundan dolayı DoxTracker'ın MS Word ve PDF dokümanları üzerine nasıl uygulandığı farklı başlıklar altında ele alınarak açıklanmıştır.

4.3.1 MS Word Dokümanlarına Uygulanması

Bir MS Word dokümanının yapısının ve çalışma mantığının bilinmesi DoxTracker'ın uygulanması aşamalarında gerçekleştirilen işlemlerin daha iyi anlaşılabilmesini sağlayacaktır. Onun için öncelikli olarak MS Word doküman yapısı açıklanmış ve sonra DoxTracker'ın nasıl uygulandığı gösterilmiştir.

Bir MS Word dokümanı, farklı amaçlar doğrultusunda birden fazla dosyanın bir araya getirildiği ve bunların sıkıştırılarak anlamlı bir formatta sunulduğu yapıdan oluşmaktadır. MS Office 2007 sürümünden itibaren bu yapı kullanılmaya başlanmıştır. İçerideki ilişkisel yapının görüntülenebilmesi için öncelikli olarak doküman, ".docx" olan dosya uzantısı yerine ".zip" yazılarak sıkıştırılmış bir dosya formatı olan zip formatına dönüştürülür. Bu işlemden sonra artık bir MS Word dokümanı değil sıkıştırılmış formatta bir dosya ile çalışılıyormuş gibi düşünülebilir. Sıkıştırılmış bir dosyanın içeriğini görüntüleyebilmek için WinRAR veya WinZIP gibi programlar kullanılabilir. Sıkıştırılmış bir dosya formatına dönüştürüldükten sonra winrar programı kullanılarak açılan MS Word dokümanın yapısının aslında Şekil 4.3'te olduğu gibi görünecektir.



ŞEKİL 4.3: Bir word dokümanının iç yapısı

Görünen bu dosya ve klasörlerin her biri farklı özellikleri bünyesinde barındırmakla beraber birbirleri ile ilişkisel bir yapıdadırlar. Şöyleki birisinin içerisinde dokümanın sahibi, dokümanın oluşturulma tarihi, doküman üzerinde en son güncelleme yapılan tarih gibi bilgiler saklanıyor iken diğerinin içerisinde ise doküman içerisinde bulunan nesnelerin tipleri (JPEG türünde bir resim nesnesi gibi) tanımlanmaktadır. Bu dosyaların herhangi birinde yapılacak olan kontrolsüz değişiklik, bir karakterlik dahi olsa, Word dokümanın

genel yapısını bozacağı için doküman açılırken hata alınmasına sebeb olacaktır. Onun için bu dosyalar üzerinde çalışırken doküman bütünlüğüne etki etmeyecek, dokümanın genel ve orijinal yapısını bozmayacak şekilde çok dikkatli olunması gerekmektedir. Bu yüzden Şekil 4.3'te görünen dosya ve klasörlerin tam olarak neleri içerdikleri ve ne işe yaradıkları aşağıda sırasıyla açıklanmıştır.

- *_rels*: Şekil 4.3'te görünen yapı bir paket olarak düşünülecek olursa; bu paket dahilinde bulunan dosyalar arasındaki ilişki bu dizin içerisinde tanımlanmaktadır. Normal şartlar altında ".rels" isimli bir dosya içermektedir.

- *_rels/.rels*: Doküman içeriğinin tutulduğu ana XML dosyalarının bulunduğu dosya konumları, bunların birbirleri ile olan bağlantıları ve dokümanın genel yapısı(dokümanın oluşturulması esnasında dosyaların nasıl birleştirileceğine dair özellikler) hakkındaki bilgiler burada tutulur. Dolayısı ile doküman açılırken uygulamalar ilk olarak buradaki bilgileri okur.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship
        Id="rId3"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/extended-properties"
        Target="docProps/app.xml"/>
    <Relationship
        Id="rId2"
        Type="http://schemas.openxmlformats.org/package/2006/relationships/metadata/core-properties"
        Target="docProps/core.xml"/>
    <Relationship
    Id="rId1"
    Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument"
    Target="word/document.xml"/>
</Relationships>
```

ŞEKİL 4.4: MS Word dosya ilişkilerinin tutulduğu .rels dosyasının içeriği

Şekil 4.4'te görüldüğü üzere her ilişki bir id (Id), bir tip (Type) ve bir hedef (Target) bilgisine sahiptir. Buradaki hedef ilişkiye konu olan dosyanın ismidir. Id değeri ise bir ilişkiyi diğeriden ayırtedebilmek için konulmuş benzersiz bir tanımlayıcıdır. Tip alanı ise MS Word için ön tanımlı olan özel dosya tiplerinden hangisinin bu dosya ile ilişkilendirileceğini belirtir. Örneğin "word" dizini altında bulunan "document.xml" dosyası Type değişkeninde belirtilen "http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument" dosya tipi ile ilişkilidir.

- *docProps*: Dokümanın meta-verisi genellikle bu dizin altında yer alan dosyalarda saklanır. Normal şartlar altında app.xml ve core.xml olmak üzere iki tane XML dosyası bu dizin altında bulunur.

- *docProps/app.xml*: Dokümanın kaç sayfa, kaç paragraf, kaç kelime ve kaç karakterden oluştuğu gibi uygulama üzerinden elde edilen meta-verilerin saklandığı XML dosyasıdır. Örnek bir "app.xml" dosya içeriği Şekil 4.5'te gösterilmiştir.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Properties
    xmlns="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
    xmlns:vt="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes">
    <Template>Normal.dotm</Template>
    <TotalTime>0</TotalTime>
    <Pages>1</Pages>
    <Words>1</Words>
    <Characters>8</Characters>
    <Application>Microsoft Office Word</Application>
    <DocSecurity>0</DocSecurity>
    <Lines>1</Lines>
    <Paragraphs>1</Paragraphs>
    <ScaleCrop>false</ScaleCrop>
    <Company/>
    <LinksUpToDate>false</LinksUpToDate>
    <CharactersWithSpaces>8</CharactersWithSpaces>
    <SharedDoc>false</SharedDoc>
    <HyperlinksChanged>false</HyperlinksChanged>
    <AppVersion>15.0000</AppVersion>
</Properties>
```

ŞEKİL 4.5: Dokümana ait uygulama seviyesi bilgilerin tutulduğu app.xml dosyası içeriği

- *docProps/core.xml*: Yazarın adı, dokümanın oluşturulma tarihi, değiştirilme tarihi, son yazdırılma tarihi gibi dokümana ait temel özellikler burada tutulur. Örnek bir "core.xml" dosya içeriği Şekil 4.6'da gösterilmiştir.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cp:coreProperties xmlns:cp="http://schemas.openxmlformats.org/package/2006/metadata/core-properties"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:dcterms="http://purl.org/dc/terms/"
    xmlns:dcmitype="http://purl.org/dc/dcmitype/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <dc:title/>
    <dc:subject/>
    <dc:creator>uk</dc:creator>
    <cp:keywords/>
    <dc:description/>
    <cp:lastModifiedBy>uk</cp:lastModifiedBy>
    <cp:revision>3</cp:revision>
    <dcterms:created xsi:type="dcterms:W3CDTF">2015-11-06T07:24:00Z</dcterms:created>
    <dcterms:modified xsi:type="dcterms:W3CDTF">2015-11-26T17:19:00Z</dcterms:modified>
</cp:coreProperties>
```

ŞEKİL 4.6: Dokümanının temel özelliklerinin tutulduğu core.xml dosyası içeriği

- *word*: Dokümanın asıl içeriğinin yer aldığı dizindir.

- *word/_rels*: Bir MS Word dokümanında, her elemanın diğer elemanlar ile arasındaki ilişkinin tanımlı olduğu bir ilişkiler bölümü olmak zorundadır. Dokümanın asıl içeriğinin yer aldığı "word" dizini altındaki elemanların da birbirleri ile olan ilişkileri bu dizinde tanımlanmaktadır. Normal şartlar altında bu dizin altında "document.xml.rels" isimli bir XML dosyası yer alır.

- *word/_rels document.xml.rels*: Dokümanın ana içerik bölümünde kullanılan stil, tema, resim, harici linkler gibi elemanlara ait ilişkilerin tanımlandığı dosyadır. Örnek bir "document.xml.rels" dosya içeriği Şekil 4.7'de gösterilmiştir.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship
        Id="rId3"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/settings"
        Target="settings.xml"/>
    <Relationship
        Id="rId7"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme"
        Target="theme/theme1.xml"/>
    <Relationship
        Id="rId2"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles"
        Target="styles.xml"/>
    <Relationship
        Id="rId1"
        Type="http://schemas.microsoft.com/office/2006/relationships/vbaProject"
        Target="vbaProject.bin"/>
    <Relationship
        Id="rId6"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/fontTable"
        Target="fontTable.xml"/>
    <Relationship
        Id="rId5"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/image"
        Target="media/image1.jpg"/>
    <Relationship
        Id="rId4"
        Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/webSettings"
        Target="webSettings.xml"/>
</Relationships>
```

ŞEKİL 4.7: Dokümanın asıl içeriğini oluşturan elemanların ilişki durumları

Şekil 4.7'de görüldüğü üzere örnekte verilen doküman içerisinde bir tane resim varmış ve bu resim "media/image1.jpg" dosya konumunda yer alıyormuş.

- *word/media*: Doküman içerisinde yer alan medya elemanlarının saklandığı dizindir. Örneğin doküman içerisine bir resim eklendiği zaman, eklenen resim bu dizin altında saklanır.

- *word/theme*: Doküman kapsamında kullanılan yazı tipleri, biçimler ve renkler gibi dokümanın temasını oluşturan özellikler hakkındaki bilgiler burada tutulur.

- *word/document.xml*: Dokümanın ana içerik bölümü burasıdır, içeriğinin çoğu burada bulunur. Bundan dolayı "document.xml" dosyası bir MS Word dokümanının ana parçasını oluşturmaktadır.

- *word/fontTable.xml*: Doküman kapsamında kullanılan yazı tipleri hakkında bilgiler burada belirtilir. Belirtilen yazı tipinin sistemde tanımlı olmaması gibi durumlarda da kullanılacak yazı tipi burada belirtilir.

- *word/settings.xml*: Yazım ve dil bilgisi hatalarının gizlenmesi/gösterilmesi gibi doküman üzerinde uygulanabilecek ayarlamaların belirtildiği dosyadır.
- *word/styles.xml*: Doküman kapsamında kullanılan stillerin yer aldığı XML dosyasıdır.
- *word/webSettings.xml*: Doküman tarafından kullanılan web'e özgü ayarlamaların tanımlandığı dosyadır. Dokümanın HTML formatında kaydedilmesi durumunda nasıl işleneceği bilgisi buradan okunur.
- *[Content_ Types].xml*: Doküman kapsamında kullanılan XML dosyaları ve nesnelere göre hangi tip paketlerin kullanılacağı ilişkileri burada tanımlanır. Ayrıca dokümanın MIME tipi/türü de bu XML dosyası içerisinde tanımlanmaktadır. Örnek bir "[Content_Types].xml" dosya içeriği Şekil 4.8'de gösterilmiştir.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Types xmlns="http://schemas.openxmlformats.org/package/2006/content-types">
    <Default
        Extension="bin"
        ContentType="application/vnd.ms-office.vbaProject"/>
    <Default
        Extension="rels"
        ContentType="application/vnd.openxmlformats-package.relationships+xml"/>
    <Default
        Extension="xml"
        ContentType="application/xml"/>
    <Default
        Extension="jpg"
        ContentType="image/jpeg"/>
    <Override
        PartName="/word/document.xml"
        ContentType="application/vnd.ms-word.document.macroEnabled.main+xml"/>
    <Override
        PartName="/word/vbaData.xml"
        ContentType="application/vnd.ms-word.vbaData+xml"/>
    <Override
        PartName="/word/styles.xml"
        ContentType="application/vnd.openxmlformats-officedocument.wordprocessingml.styles+xml"/>
    <Override
        PartName="/word/settings.xml"
        ContentType="application/vnd.openxmlformats-officedocument.wordprocessingml.settings+xml"/>
    <Override
        PartName="/word/webSettings.xml"
        ContentType="application/vnd.openxmlformats-officedocument.wordprocessingml.webSettings+xml"/>
    <Override
        PartName="/word/fontTable.xml"
        ContentType="application/vnd.openxmlformats-officedocument.wordprocessingml.fontTable+xml"/>
    <Override
        PartName="/word/theme/theme1.xml"
        ContentType="application/vnd.openxmlformats-officedocument.theme+xml"/>
    <Override
        PartName="/docProps/core.xml"
        ContentType="application/vnd.openxmlformats-package.core-properties+xml"/>
    <Override
        PartName="/docProps/app.xml"
        ContentType="application/vnd.openxmlformats-officedocument.extended-properties+xml"/>
</Types>
```

ŞEKİL 4.8: Dokümanın asıl içeriğini oluşturan elemanların ilişki durumları

Şekil 4.8'de görüldüğü üzere tanımlanan ilk Extension değeri "bin"'dir. Böyle bir Extension'ın tanımlanmış olması bu MS Word dokümanının bir makro kodu içerdiğini göstermektedir. Bundan dolayı dokümanın türü makro içeren bir word dokümanı olarak tanımlanmış olmalıdır. Bunun için de şekil üzerinde, PartName= "/word/document.xml" olan Override elemanının ContentType değerinin "macroEnabled" olduğu gözlemlenebilir.

Bir MS Word dokümanının açılması ile tetiklenecek olan alarm mekanizması, kaynak adresi DoxTracker sunucusu olan bir resim nesnesinin doküman içerisine gizli kalacak şekilde yerleştirilmesi üzerine inşa edilmiştir. Bundan dolayı farklı senaryolar altında bir Word dokümanının içerisine resimler yerleştirilerek yukarıda detaylı olarak bahsedilen tüm dosya ve klasörler arasındaki ilişkiler en ince ayrıntısına kadar incelenmiştir. Yapılan incelemeler neticesinde elde edilen verilerden yola çıkarak Word dokümanının bütünlüğüne herhangi bir zarar vermeyecek şekilde istenilen manipülasyonların nasıl gerçekleştirilebileceği tespit edilmiştir. Örneğin PNG dosya tipinde bir resim nesnesinin Word dokümanına eklenmesi sırasında ilişkisel yapının korunabilmesi ve dokümanın açılırken hata vermemesi için bu dokümanın içerisinde artık PNG dosya tipinde bir resim nesnesi vardır diye tanımlama yapmak gerekmektedir. Bunun için de "[Content_Types].xml" dosyasının içerisine şu satır eklenir:

```
<Default Extension=''png'' ContentType=''image/png'' />
```

Bu ve benzeri ilişkiler korunarak gerçekleştirilen değişiklikler ve düzenlemelerden sonra dokümanın orijinal yapısı bozulmadan alarm mekanizması kurulur. Bu işlemden sonra artık doküman her açıldığında içerisine gömülen gizli resim nesnesi, resmin bulunduğu kaynak adresi yani DoxTracker sunucusuna HTTP 80. port üzerinden bir istekte bulunur. Böylelikle gelen istekten yola çıkılarak hangi dokümanın, hangi coğrafi konumdan, ne zaman ve kim tarafından açıldığı gibi bilgiler elde edilir.

Takip edilen dokümanların tanımlanabilmesi amacıyla doküman içerisine yerleştirilen gizli resim nesnesinin URL adresine, her dokümana özel olan benzersiz bir ID değeri eklenir. Dolayısı ile bir doküman açıldığı zaman DoxTracker sunucusuna bu ID değeri ile geri bildirimde bulunur. DoxTracker merkezi sunucusu üzerinde bulunan ve gelen olay isteklerini karşılayarak yöneten Olay Kayıt Merkezi hangi dokümanın açıldığını bu ID

değeri üzerinden tespit eder. İlgili veritabanı kayıt işlemlerini de buna göre gerçekleştirir. Bahsi geçen ve her dokümana özgü olacak şekilde oluşturulan benzersiz ID değerleri şöyle üretilmektedir:

1. Öncelikle PHP'nin kütüphanelerinden biri olan random ve benzersiz bir AES anahtar üretme fonksiyonu olan "openssl_random_pseudo_bytes" ile AES şifreleme için kullanılacak anahtar üretilir.

2. İlk adımda üretilen anahtar kullanılarak kullanıcıdan alınan doküman ismi AES şifreleme tekniği ile şifrelenir ve takip kodu elde edilir.

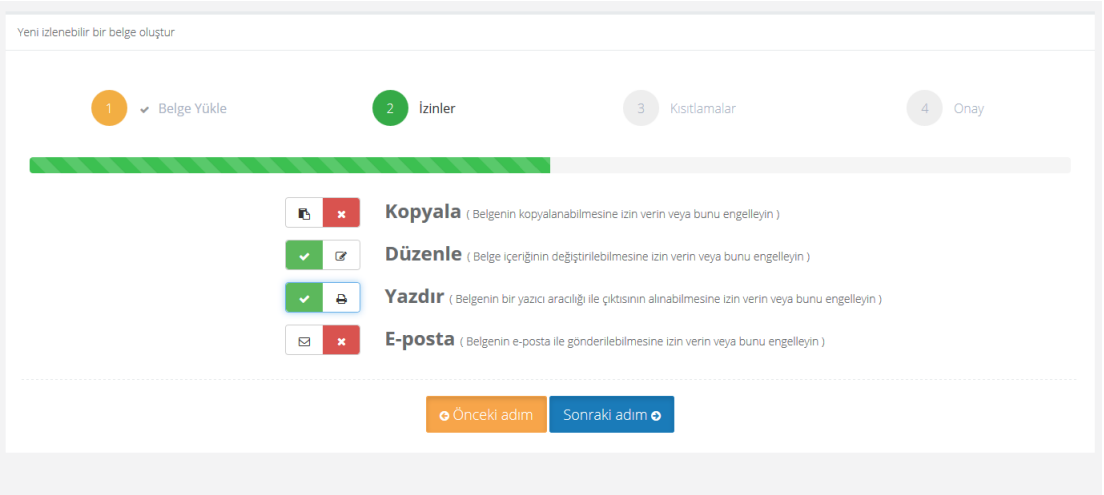
   ```
   TakipKodu = AES (UniqueKey, Doküman Adı)
   ```

3. Doküman içerisine yerleştirilecek olan gizli resim nesnesinin kaynak adresi oluşturulur ve "word/_rels/document.xml.rels" XML dosyasının içerisine aşağıdakine benzer yapıda bir XML satırı olarak eklenir.

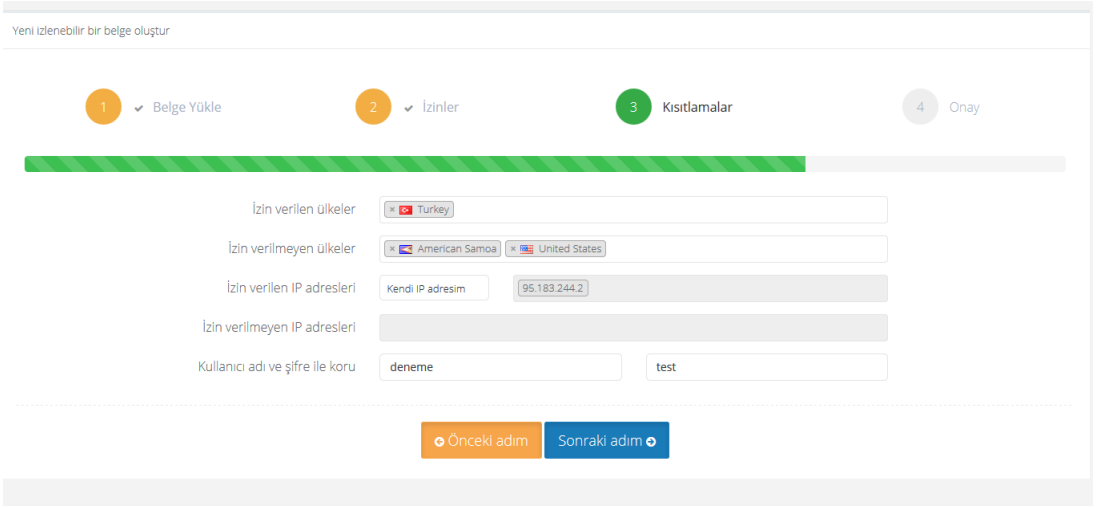
   ```
   <Relationship Target="http://doxtracker.com/opened?docId=TakipKodu"
   Type="http://schemas.openxmlformats.org/officeDocument/2006/
   relationships/image" Id="rId4" TargetMode="External" />
   ```

Dokümanın açılıp açılmadığının takip edilebilmesi amacıyla yerleştirilen gizli bir resmin, "TargetMode" parametresinde belirtilen "External" değeri sayesinde internet ortamından çekileceği belirtilir. Doküman açıldığında çekileceği internet adresi de "Target" parametresi ile belirtilmektedir. Doküman açıldığı zaman resim nesnesi, "Target" parametresi ile belirtilen internet adresine, sanki ID değeri TakipKodu olan resim dosyasını istiyormuş gibi HTTP 80. port üzerinden bir istekte bulunur. Kendisine böyle bir istek gelen DoxTracker sunucusu, cevaben istenilen resim dosyasını göndermek yerine bunu bir doküman açma olayı olarak değerlendirir ve ilgili bilgileri veritabanına kaydeder.

DoxTracker kapsamında gerçekleştirilen ve bu zamana kadar yapılmış olan çalışmalarda üzerinde durulmayan özellik, doküman içeriğinin şifrelenerek korunmasıdır. Bundan dolayı önerilen sistem için geliştirilen prototipte, bir dokümanın takip edilebilir formata dönüştürülmesi aşamasında çeşitli izin ve kısıtlamaların tanımlanabilmesine olanak sağlanmıştır. Önerilen sistem prototibinde yer alan dokümanın düzenlenebilmesi, yazdırılabilmesi, içeriğinin kopyalanabilmesi ve çeşitli erişim kısıtlamaları gibi doküman içeriğini korumaya yönelik sunulan ek özellikler Şekil 4.9 ve Şekil 4.10'da gösterilmiştir.



ŞEKİL 4.9: DoxTracker izin tanımlama ekranı



ŞEKİL 4.10: DoxTracker kısıt tanımlama ekranı

MS Word dokümanlarını korumaya yönelik doküman üzerinde gerçekleştirilebilecek işlemlerin (düzenleme, kopyalama, yazdırma) kontrolü MS Word makroları aracılığı ile sağlanmaktadır. Dokümanın oluşturulması aşamasında tanımlanan izinlere göre ilgili MS Word makro kodu doküman içerisine yerleştirilmektedir. Örneğin doküman sahibi, dokümanın yazdırılamaması yönünde bir engelleme tanımlamış ise en basit hali ile aşağıdaki gibi bir word makro kodu doküman içerisine gömülür.

```
1 Sub FilePrint()
2   MsgBox 'Bu doküman yazdırılamaz!'
3 End Sub
```

MS Word dokümanının içerisinde ilişkisel yapıda olan birden fazla dosyanın yer aldığı daha önce belirtilmişti. Dolayısı ile hata alınmaması için bir doküman içerisinde yapılan

herhangi bir değişiklikten ilgili diğer dosyaların da haberdar edilmesi gerekmektedir. Örneğin "word/_rels/document.xml.rels" XML dosyasının içerisine, dokümana eklenen makronun çalıştırılabilir/derlenmiş kodlarının "vbaProject.bin" dosyası içerisinde olduğu bilgisini belirten aşağıdaki XML satırı eklenir:

```
<Relationship Target="vbaProject.bin" Type="http://schemas.microsoft.com/
office/2006/relationships/vbaProject" Id="rId1"/>
```

"[Content_Types].xml" XML dosyasının içerisine, doküman içerisinde artık bir makro kodu olduğunu belirten aşağıdaki XML satırları eklenir:

```
<Default Extension="bin" ContentType="application/
vnd.ms-office.vbaProject"/>
<Override PartName="/word/vbaData.xml" ContentType="application/
vnd.ms-word.vbaData+xml"/>
```

Bu işlemlerden sonra doküman artık eskisi gibi normal bir word dokümanı değil de içerisinde makro kodu bulunduran bir doküman olduğu için doküman formatı da değiştirilmelidir. Bunun için öncelikle doküman tipinin tanımlandığı "[Content_Types].xml" XML dosyası içerisinde bulunan ve PartName değeri "/word/document.xml" olan XML satırı bulunarak bu satırda bulunan ContentType değeri "application/vnd.ms-word.document.macroEnabled.main+xml" olarak değiştirilir. Böylelikle doküman tipi makrolu doküman olarak değiştirilmiş olunur. Son olarak da dosya uzantısı ".docx" yerine ".docm" yapılarak kullanıcıya verilir. Tüm bu işlemler büyük bir titizlikle yerine getirilmelidir. Örneğin sadece dosya uzantısının bile ".docx" yerine ".docm" yapılmaması dokümanın açılırken hata vermesine ve açılmamasına neden olacaktır.

Kısıtlamalar ise dokümana özel olarak üretilen ID değerlerine göre, DoxTracker veritabanında saklanır. Gelen erişim isteklerini veritabanında kayıtlı olan kısıtlamalara göre değerlendirilerek gerekli işlemler yapılır. Örneğin doküman sahibi bir MS Word dokümanı için içeriğin sadece Türkiye'den görüntülenebilmesi gibi bir kısıtlama koyabilir. Bu durumda öncelikli olarak doküman içeriğinin korunabilmesi için şifrelenmesi gerekir. Yukarıda da bahsedildiği üzere bir MS Word dokümanı içerisinde bulunan her bir dosya

ve klasör birbiri ile ilişkili olduğu için sadece Word dokümanının içeriğinin alınarak şifrelenmesi yöntemi bu noktada işe yaramamaktadır. Farklı yöntemler ile bu işlem gerçekleştirilebilir fakat geliştirilen prototip kapsamında uygulanan yöntemin adımları sırasıyla şöyledir:

- Öncelikli olarak bir MS Word dokümanının içeriği ve bu içeriğin düzgün bir şekilde oluşturulabilmesi/gösterilebilmesi ile ilgili olan orijinal dosyalar ana dizin altına DoxTracker ismi ile açılan yeni bir klasörün içerisine taşınır. Taşınan bu dosyalar şunlardır:

  1. [Content_Types].xml
  2. docProps/app.xml
  3. word/document.xml
  4. word/settings.xml
  5. word/_rels/document.xml.rels

- Bu dosyaların yerine ise dokümanın DoxTracker tarafından korunduğunu gösteren DoxTracker karşılama ekranı sayfası için hazırlanmış olan aynı isimlere sahip versiyonları yerleştirilir,

- DoxTracker dosyası içerisine taşınan orijinal doküman içeriği AES-256 şifreleme tekniği kullanılarak şifrelenir,

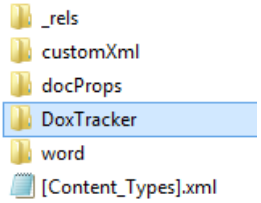
- Son olarak tüm bu dosyalar tekrar birleştirilerek word doküman formatına dönüştürülür ve kullanıcıya geri verilir.

Bu şekilde koruma altına alınan bir MS Word dokümanı açıldığı zaman, dokümana ait orijinal içerik değil de Şekil 4.11'de olduğu gibi DoxTracker karşılama ekranı görüntülenecektir.

Bu belge DoxTracker tarafından korunmaktadır. Belge içeriğini görüntülemek için yetkiniz var ise makroyu çalıştırınız.

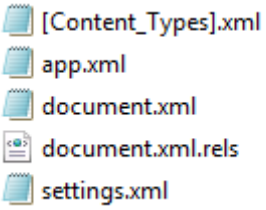
ŞEKİL 4.11: DoxTracker tarafından korunan word dokümanı karşılama sayfası

İçeriğin gizlenerek saklandığı bu gibi durumlarda kötü niyetli ve biraz da yetenekli kullanıcılar orijinal doküman içeriğini görüntüleyebilmek için winrar gibi programlar aracılığıyla word dosyasını açabilirler. Bu durumda ise Şekil 4.12'de olduğu gibi bir dizin yapısı ile karşılaşılacaktır.

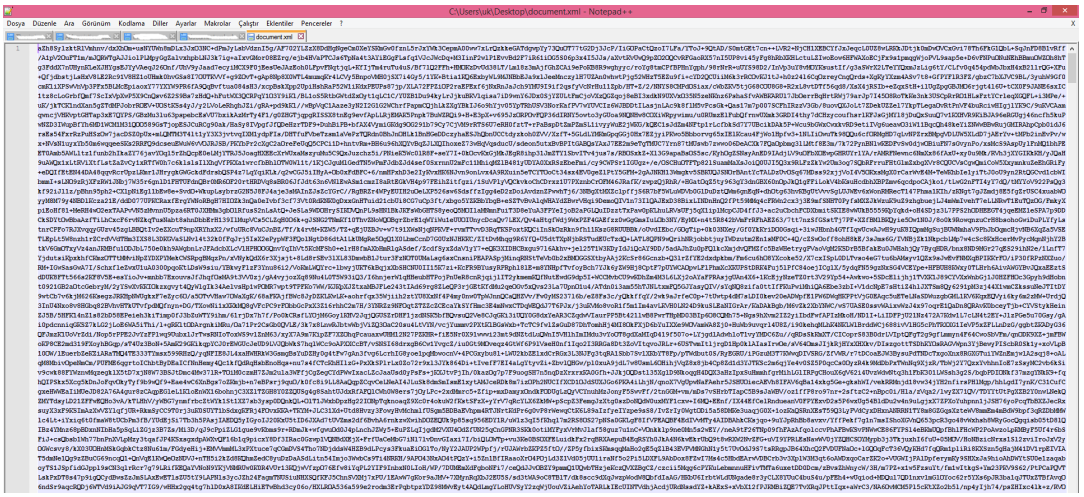


ŞEKİL 4.12: DoxTracker tarafından korunan word dokümanının iç yapısı

Orijinal dokümana ait içerik ve bununla ilişkisi bulunan diğer dokümanlar DoxTracker klasörü içerisindedir. Bu klasör içerisinde bulunan "document.xml" orijinal doküman içeriğinin bulunduğu dosyadır. Bu dosya şifrelenerek koruma altındadır. Orijinal word dokümanına ait ilgili dosyalar Şekil 4.13'de, şifreli olan "document.xml" dosyasının içeriği ise Şekil 4.14'te görünmektedir.



ŞEKİL 4.13: DoxTracker tarafından korunan word dokümanının iç yapısı



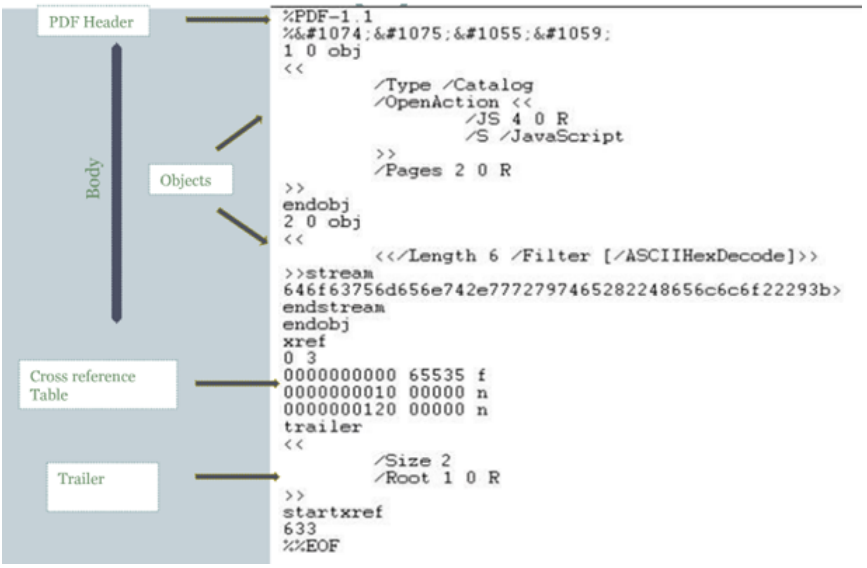
ŞEKİL 4.14: Şifreli orijinal word dokümanı içeriği

DoxTracker ile hazırlanan bir word dokümanın içeriği şifreli olarak koruma altında olduğu için içeriğin görüntülenebilmesi için kullanıcı ya makronun çalıştırılması ya da şifre çözme/şifreleme işlemlerini yapabilen bir MS Office eklentisi olan DoxTracker eklentisini yüklenerek çalıştırılması gerekmektedir. Makroların aktif edilmesi ile DoxTracker makroları otomatik olarak çalışmaya başlayacaktır. Doküman açıldığı zaman içerisine yerleştirilmiş olan gizli resim aracılığı ile dokümanı açan kullanıcının haberi olmadan arka planda HTTP 80. port üzerinden DoxTracker sunucusuna gönderilen şifreli ve gizli olan orijinal içeriğin görüntülenebilmesi için aktifleştirilmesi ile yine arka planda otomatik olarak bu sefer HTTPS 443. port üzerinden gelmesi DoxTracker tarafından doküman içeriğinin görüntülemek isteyen kullanıcının IP adresi ve coğrafi konumu gibi bilgiler alınacaktır. Alınan bu bilgiler veritabanında tanımlı olan kısıtlamalar süzgecinden geçirilerek gerekli kontroller yapılır. Yapılan kontrollere göre de sunucudan cevap gönderilir. Örneğin yukarıda verilen örnek üzerinden gidilirse eğer bu isteği gönderen kullanıcı Türkiye sınırları içerisinde bulunmuyor ise cevap olarak şifre çözme için gerekli olan anahtar yerine hata mesajı gönderilecektir. Fakat belirtilmiş olan kısıt yani bu doküman içeriği sadece Türkiye'de görüntülenebilir kısıtını sağlayan bir durum var ise o zaman yine aynı güvenli bağlantı üzerinden cevap olarak şifre çözme için gerekli olan anahtar gönderilecektir. Eğer sunucudan şifre çözme işlemi için anahtar gönderilirse de DoxTracker klasörü içerisine konumlandırılmış olan şifreli orijinal içerik

çözümlenecek ve diğer orijinal XML dosyaları ile birlikte olması gereken dizinlere taşınacaktır.

4.3.2 PDF Dokümanlarına Uygulanması

Yapısı ve çalışma mantığı farklı olan PDF dosya formatının takip edilebilmesi için aynı yaklaşım ve yöntemler farklı teknikler kullanılarak uygulanmıştır. Temel yaklaşım ve uygulama yöntemi MS Word dokümanı için detaylı olarak anlatıldığından dolayı bu bölümde çok detaya girilmeden kullanılan teknikler ve bu tekniklerin uygulanması basitçe gösterilmiştir. Öncelikle DoxTracker'ın PDF dokümanlarına nasıl uygulandığının daha iyi anlaşılabilmesi adına bir PDF dokümanının iç yapısının nasıl olduğu genel hatları ile bilinmelidir.



ŞEKİL 4.15: PDF doküman yapısı [30]

Şekil 4.15'te görüldüğü üzere bir PDF dosyası; versiyon bilgisinin gösterildiği tek satırlık başlık kısmı (Header), kullanıcılara gösterilen içeriğin ve nesnelerin yer aldığı gövde (Body), nesnelerin doküman içerisinde bulunduğu konum bilgilerini tutan çapraz-referans tablosu (Cross Reference Table) ve PDF okuyucuların çapraz-referans tablosunu ve diğer özel nesneleri nasıl bulacağı bilgisini barındıran römork (Trailer) olmak üzere 4 ana parçadan oluşmaktadır.

Önerilen sistemde en temel amaç dokümanın açılıp açılmadığının tespiti olduğu için PDF dokümanlarına uygulanacak yöntem öncelikli olarak bu amacın gerekliliklerini yerine getirmelidir. MS Word dokümanlarında bu işlem doküman içerisine yerleştirilen ve kaynak

adresi merkezi sunucu olan gizli bir resim nesnesi ile sağlanmaktaydı. Fakat PDF dokümanlarında resim nesneleri derleme sırasında doküman içerisine gömüldüğü için yani PDF dokümanlarına dış kaynaklı bir resim nesnesi eklenemediği için bu yöntem burada işe yaramamaktadır. Bundan dolayı PDF dokümanlarında alarm mekanizması doküman içerisine yerleştirilen Javascript kodları ile kurulmuştur. MS Word için gerçekleştirilen adımlarda olduğu gibi doküman takibi sırasında hangi dokümanın açıldığının tespit edilebilmesi adına her dokümana özel benzersiz bir ID değeri oluşturulur ve javascript kodu ile merkezi sunucuya gönderilecek şekilde doküman içerisine yerleştirilir. En basit hali ile yerleştirilen javascript kodu şöyle düşünülebilir.

```
$js = <<<EOD
  window.location.href = 'http://www.doxtracker.com/opened?docId=
  TakipKodu';
EOD;
```

MS Word dokümanlarında izinler (kopyalama, düzenleme ve yazdırma) word makroları aracılığı ile yönetilmekteydi. PDF dokümanlarında ise bu işlem PDF dokümanının kendi sahip olduğu özelliklerin düzenlenmesi ile dışarıdan herhangi bir kontrolcüye ihtiyaç duyulmadan yerine getirilmektedir. Örneğin bir PDF dokümanının kopyalanması ve düzenlenebilmesi engellenerek yalnızca yazdırılabilir olmasının istendiği bir durumda, bu işlemi gerçekleştiren sözde kod (pseudocode) basitçe şöyledir:

```
import 'PHP Libraries' // Gerekli PHP kütüphaneleri içeri aktarılır
get 'PDF Document' // Kullanıcıdan alınan PDF dokümanı şablon olarak
  ayarlanır
SetProtection('print') // Sadece yazma izni verilir
Output // Doküman kullanıcıya geri verilir
```

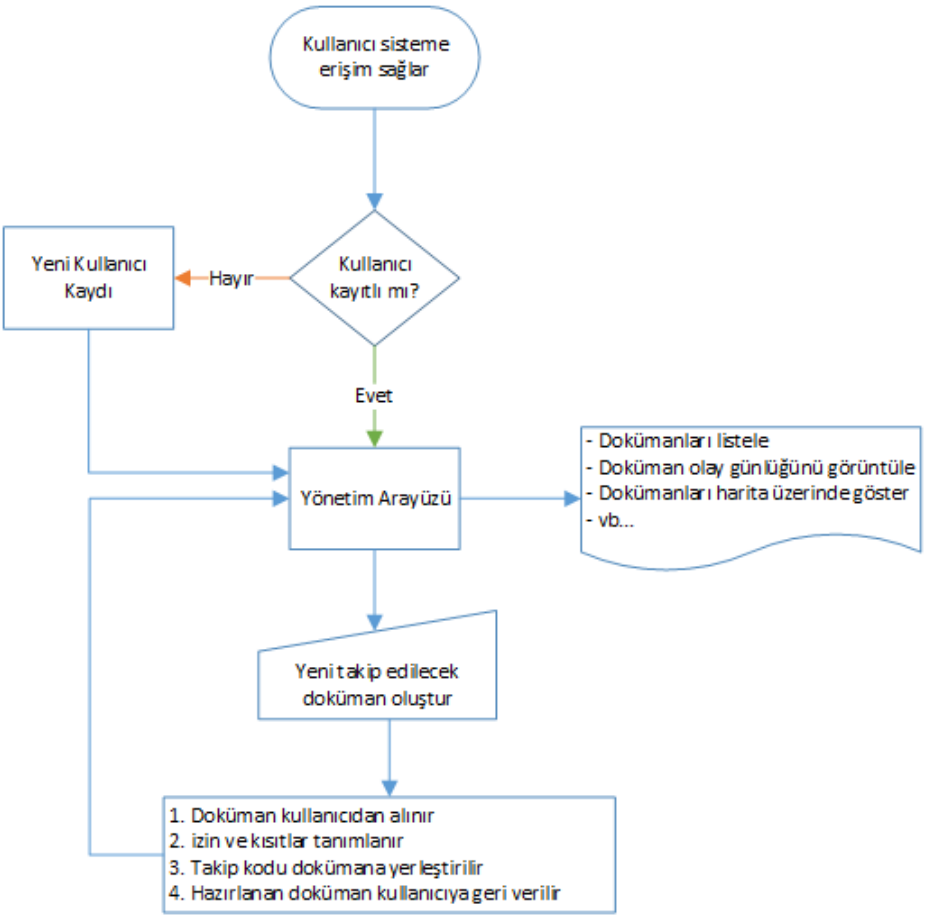
Doküman içeriğinin korunumu için MS Word dokümanında olduğu gibi AES şifreleme yöntemi kullanılmıştır. Kullanıcıdan alınan PDF dokümanının içeriği şifrelenerek gizlenmektedir. MS Word dokümanlarında olduğu gibi burada da şifreli dokümanlar için karşılama sayfası hazırlanmıştır. Tek farkı MS Word dokümanlarında makroların çalıştırılması ile otomatik olarak devreye giren şifre çözme mekanizması burada bir butona tıklanması ile tetiklenmektedir. Şifrelenerek oluşturulan bir PDF dokümanının açılması durumunda kullanıcının karşısına çıkan sayfa Şekil 4.16'da gösterilmiştir.

ŞEKİL 4.16: DoxTracker tarafından korunan PDF Dokümanı karşılama sayfası

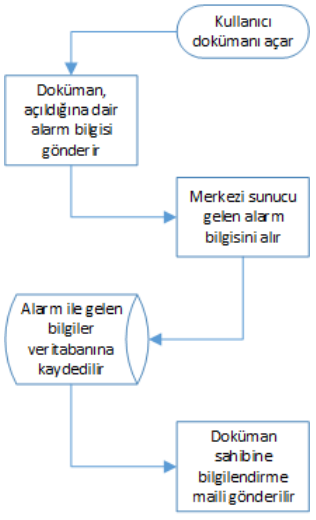
4.4 DoxTracker'ın Kullanımı

Sisteme kayıtlı olan bir kullanıcı, internet erişimi olan herhangi bir yerden giriş yaparak kullanıcı dostu olan DoxTracker Yönetim Arayüzü üzerinden daha önceden takip etmek için oluşturduğu dokümanları listeleyebilir, bunların son durumları hakkında çeşitli bilgileri görüntüleyebilir, takip etmek istemediği dokümanları silebilir ve yeni bir dokümanı takip edilebilecek formata dönüştürerek takip işlemini başlatabilir. Önerilen sistem için geliştirilen DoxTracker prototibinin temel kullanımı Şekil 4.17'de yer alan akış diyagramında gösterilmiştir.

Geliştirilmesi devam eden DoxTracker ile internet olmayan ortamlarda da açılan dokümanların takip edilebilmesi üzerine çalışmalar yapılmaktadır. Fakat bu çalışma kapsamında gerçeklenen prototip ile takip edilmekte olan bir dokümanın alarm bilgisi gönderebilmesi için internet erişimi olan herhangi bir yerde açılması gerekmektedir. Doküman açıldığı zaman hemen merkezi sunucuya HTTP üzerinden bir alarm bilgisi gönderilir. Alarm bilgisi ile gelen diğer bilgiler (doküman ismi, kullanıcı IP adresi, vb.) merkezi sunucu tarafından yorumlanarak veritabanına aktarılır. Burada önemli olan noktalardan birisi merkezi sunucu ile doküman arasındaki haberleşmelerin HTTP:80 numaralı port üzerinden gerçekleşiyor olmasıdır. Bundan dolayı haberleşmeler herhangi bir güvenlik duvarı engeline takılmayacaktır. Bahsedilen alarm mekanizmasının çalışması Şekil 4.18'de yer alan akış diyagramında gösterilmektedir.

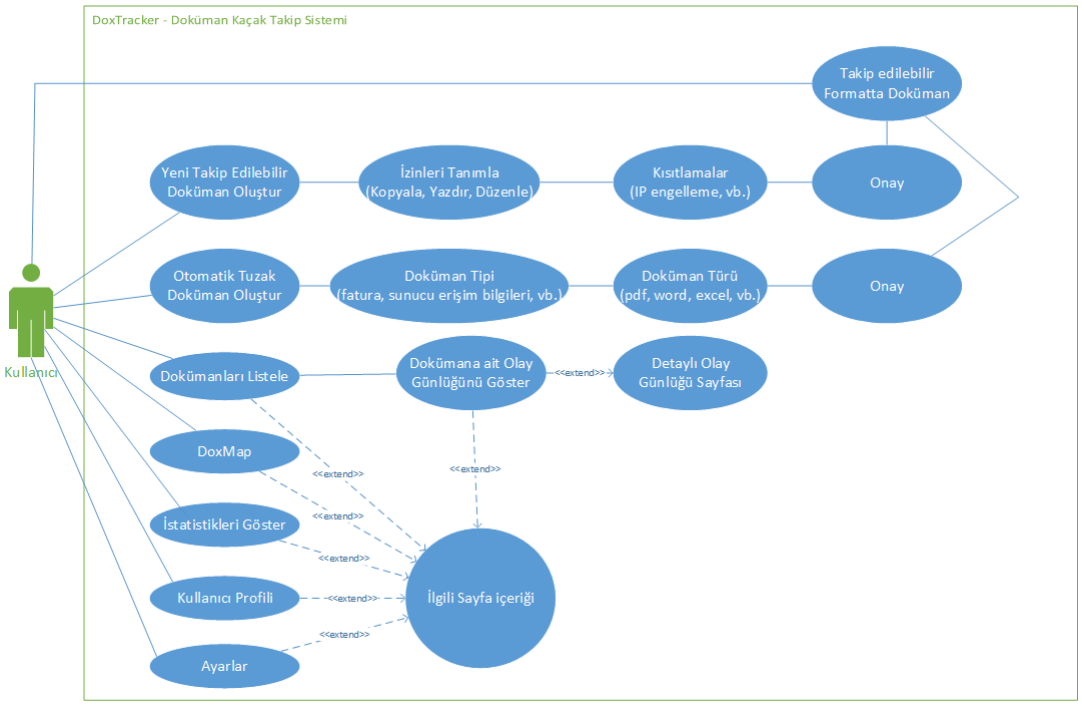


ŞEKİL 4.17: DoxTracker genel kullanım akış diyagramı



ŞEKİL 4.18: Alarm mekanizmasının çalışma akış diyagramı

Önerilen sistem için geliştirilen DoxTracker prototibi dahilinde, bir kullanıcı ile sistem arasında gerçekleşebilecek diyaloglar Şekil 4.19'da yer alan diyagramda gösterilmiştir. Bu diyaloglardan bir kaçının kullanımı geliştirilen prototip üzerinden alınan ekran görüntüleri ile desteklenerek sonraki alt başlıklarda gösterilmiştir.

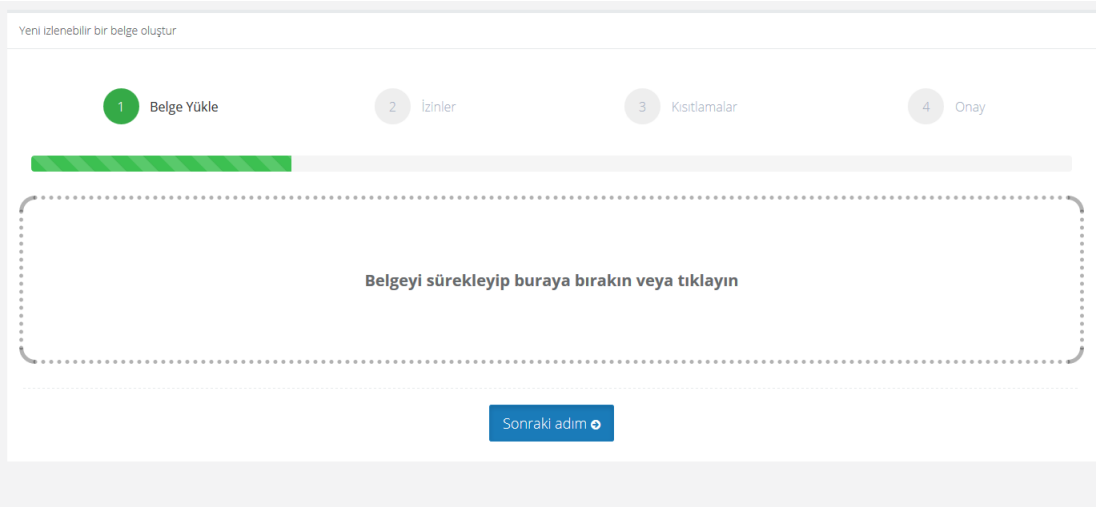


ŞEKİL 4.19: DoxTracker use-case diyagramı

4.4.1 Yeni Takip Edilebilir Doküman Oluşturma

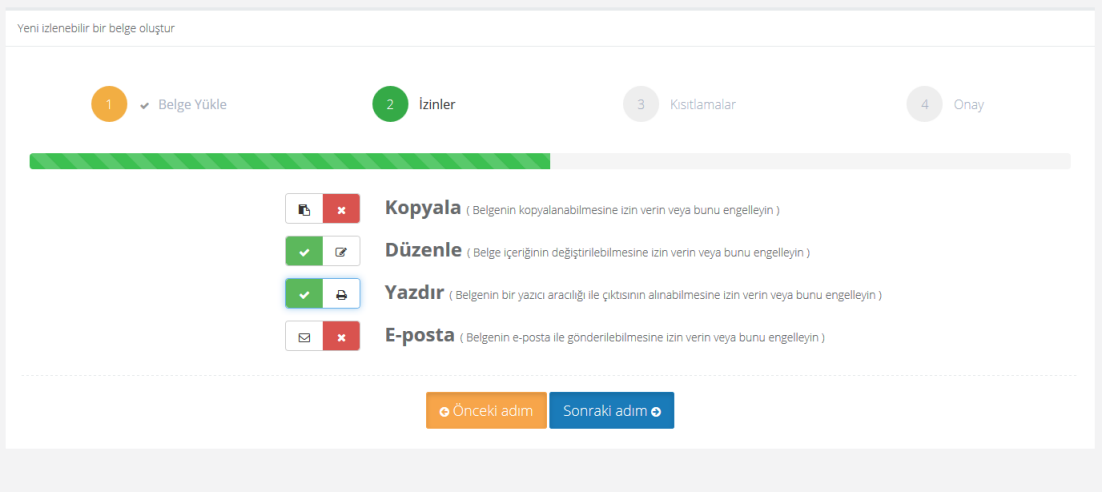
Kayıtlı bir kullanıcı geliştirilen DoxTracker prototibini kullanarak sahip olduğu bir dokümanı takip edebilir veya içeriğini şifreleyerek koruma altına alabilir. Bu doğrultuda takip edilebilir formatta yeni bir doküman oluşturulması için gerçekleşen adımlar şunlardır:

1. İlk olarak kullanıcı, Dashboard sayfasında bulunan "Yeni Takip Edilebilir Doküman Oluştur" butonunu tıklayarak ilgili sayfayı çağırır. Şekil 4.20'de yer alan ekran görüntüsünde de görüldüğü üzere takip edilmek istenilen dokümanın seçilmesi gereken ilk adım kullanıcıyı karşılar. Kullanıcı bu aşamada takip etmek istediği dokümanı seçer. Bu doküman DoxTracker sunucusu üzerinde /files dizini altında işlem tamamlanana kadar geçici olarak saklanır. İşlem tamamlandıktan sonra ise kalıcı olarak silinir.



ŞEKİL 4.20: Doküman yükleme ekranı

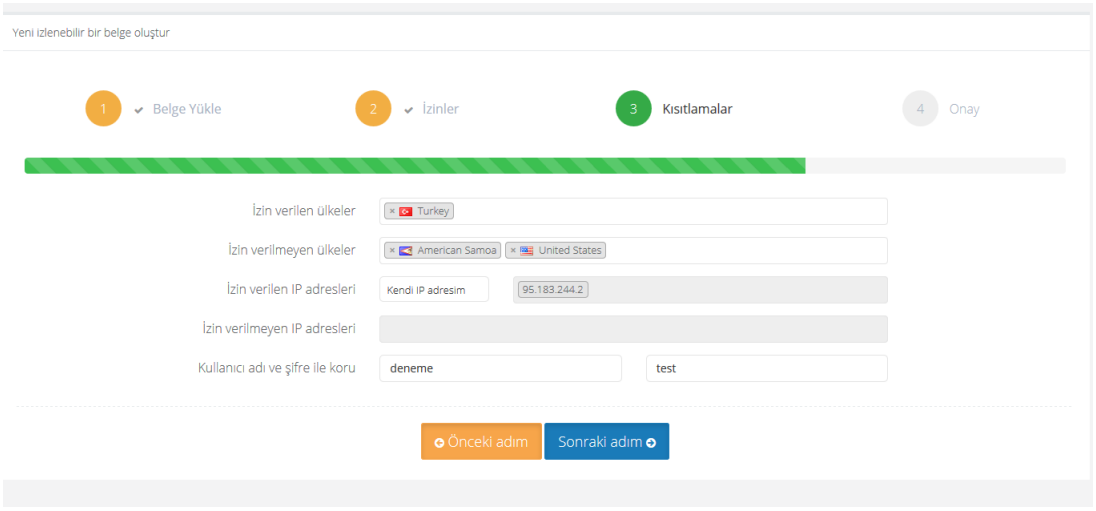
2. Takip edilecek doküman seçildikten sonra doküman üzerinde gerçekleştirilebilecek işlemler için izin tanımlamaları yapılır. Örneğin Şekil 4.21'de gösterilen ekran görüntüsünde de gösterildiği üzere doküman içeriğinin başkaları tarafından kopyalanması ve dokümanın e-posta olarak gönderilmesi engellenmiştir.



ŞEKİL 4.21: İzin tanımlama ekranı

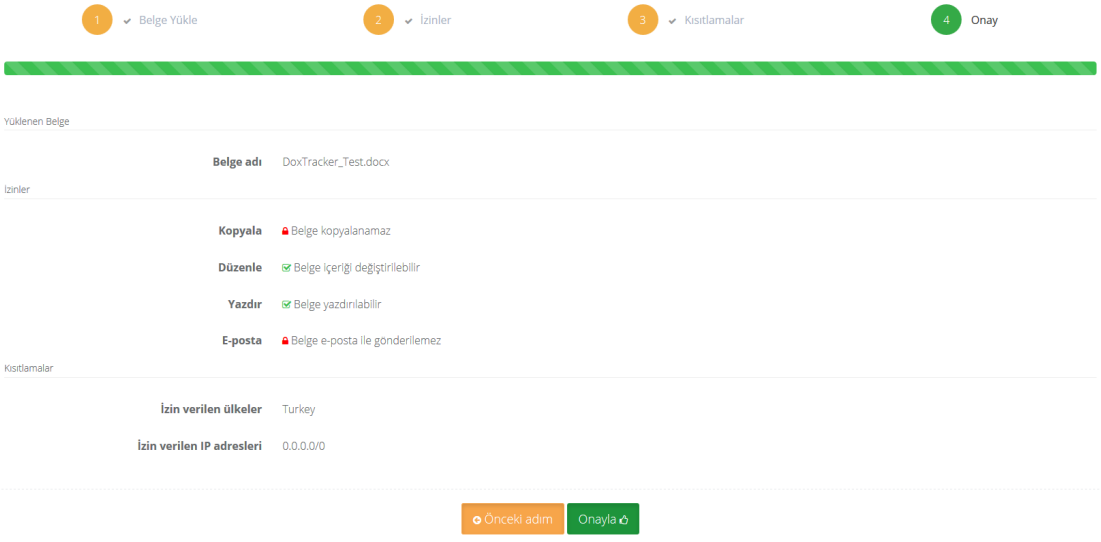
3. Gerekli izinler de tanımlandıktan sonra çeşitli kısıtlamaların tanımlanacağı aşamaya geçilir. Bu aşama genellikle içeriği korunmak istenilen dokümanlar için gerekli olmaktadır. Örneğin bu aşamada doküman içeriğinin sadece izin verilen belirli IP'lerden gelen kullanıcılar tarafından görüntülenmesi kısıtı tanımlanabilir. Böyle bir kısıt var iken doküman açıldığı zaman Olay Kayıt Merkezine gelen alarm bilgisi değerlendirmeye alınır. Bu alarm bilgisi üzerinden elde edilen bilgiler, vertabanında kayıtlı olan kısıtlar ile karşılaştırılır. Bu karşılaştırma sonucuna göre dokümanın

şifreli olan içeriğini çözümleyecek olan şifreleme anahtarı cevap olarak gönderilir veya gönderilmez.

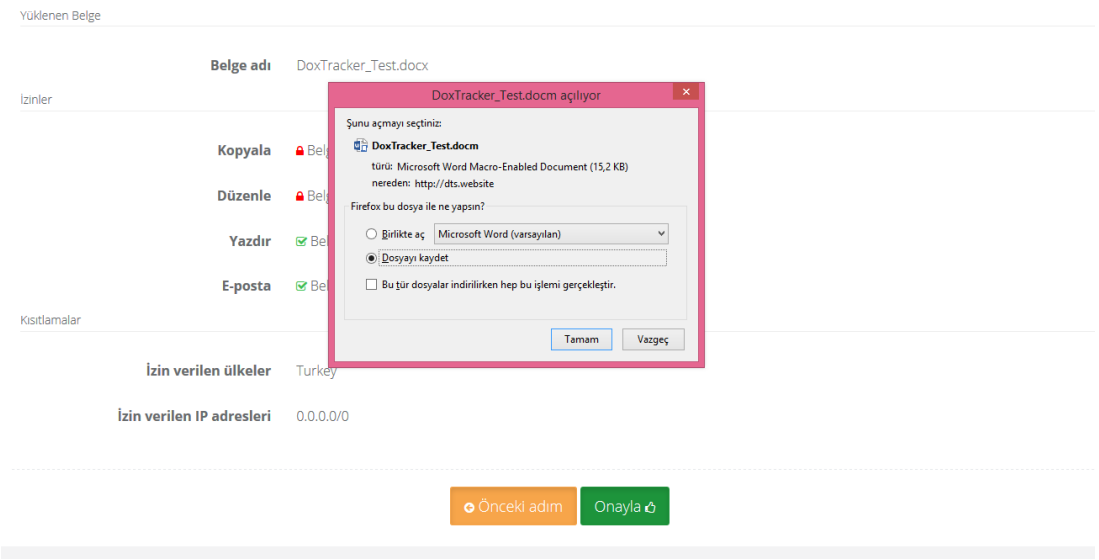


ŞEKİL 4.22: Kısıtlama tanımlama ekranı

4. Önceki adımlarda tanımlanan izin ve kısıtlar dokümana uygulanmadan yani doküman takip edilebilir formata dönüştürülmeden önce tanımlanan izinler ve kısıtlamalar kullanıcının onayına sunulur. Kullanıcı girmiş olduğu değerleri kontrol ettikten sonra süreci başlatmak için onaylar veya geri dönerek tanımlamalarında değişiklikler yapabilir. Onay işleminden sonra ise tanımlanan izinler ve tanımlara göre doküman, çok kısa bir sürede takip edilebilecek formata dönüştürülerek kullanıcıya geri verilir.



ŞEKİL 4.23: Özet ve onay ekranı

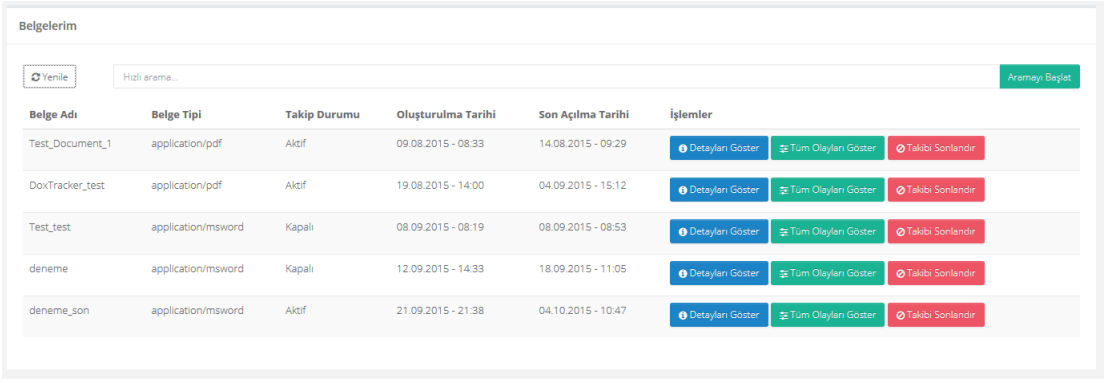


ŞEKİL 4.24: Hazırlanan dokümanın kullanıcıya sunulması

5. Artık dokümanın nerede, ne zaman ve kim tarafından açıldığı takip edilebiliyor olduğu için tereddüt edilmeden internet ortamında paylaşılabilir.

4.4.2 Doküman Yönetimi ve Olay Günlüğü

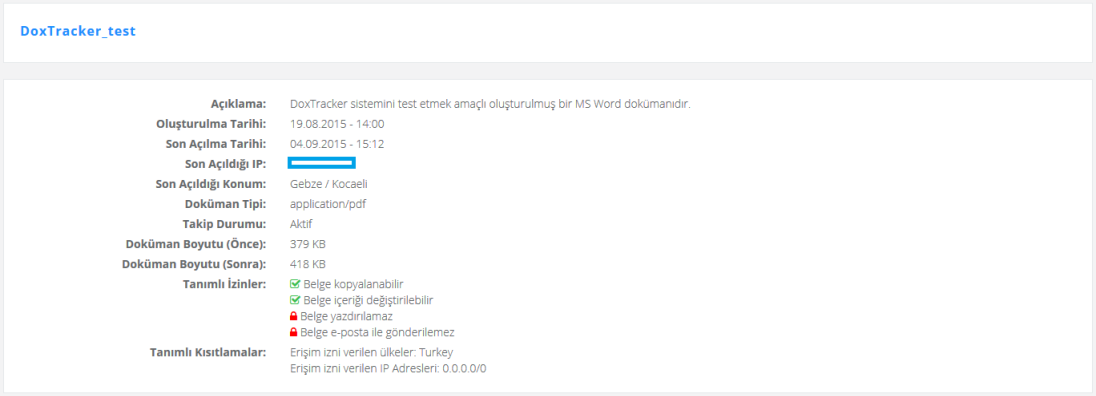
Önerilen sistem için geliştirilen prototip, kullanıcılara sahip oldukları dokümanları yönetebileceği web tabanlı bir arayüz sunmaktadır. Bu arayüz aracılığı ile takip edilen tüm dokümanlar listelenebilir, silinebilir ve her bir doküman için detaylı olay kayıtları görüntülenebilir. Şekil 4.25'te sahip olunan dokümanların listelendiği ekrandan alınan bir ekran görüntüsü yer almaktadır.



ŞEKİL 4.25: Oluşturulan dokümanların görüntülendiği ekran

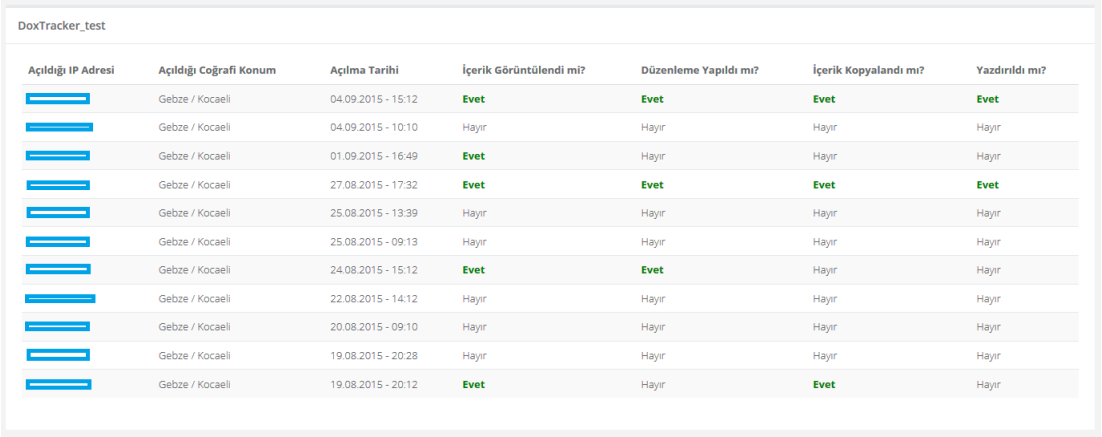
Dokümanların listelendiği sayfada her dokümana özel işlemler bulunmaktadır. Kullanıcı listede yer alan bir dökümana ait detaylı bilgileri 'Detayları Göster' butonuna tıklayarak

görüntüleyebilir. Bir dokümana ait detaylı bilgilerin yer aldığı sayfaya ait ekran görüntüsü Şekil 4.26'da gösterilmiştir.



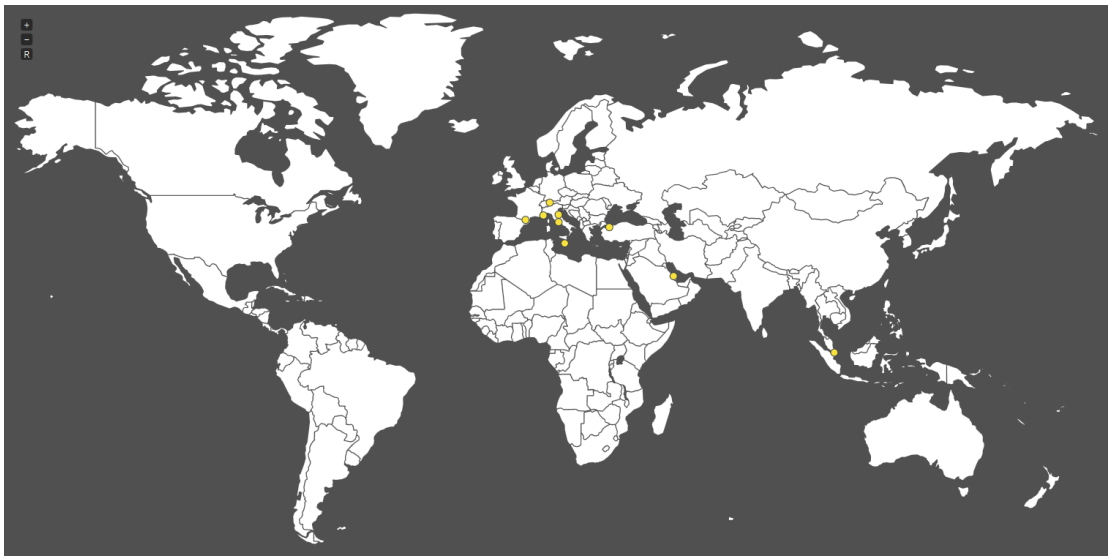
ŞEKİL 4.26: Bir dokümana ait detaylı bilgi görüntüleme ekranı

Bir dokümanın ne zaman, nereden ve kim tarafından açıldığı; üzerinde değişiklik yapılıp yapılmadığı, içeriğinin kopyalanıp kopyalanmadığı, dokümanın yazdırılıp yazdırılmadığı gibi bilgiler ise ilgili dokümana ait 'Tüm Olayları Göster' butonuna tıklanarak görüntülenebilir. Bir dokümana ait detaylı olay günlüğüne ait ekran görüntüsü Şekil 4.27'de gösterilmiştir.



ŞEKİL 4.27: Bir dokümana ait olay günlüğü görüntüleme ekranı

Prototip kapsamında güncel olayların anlık olarak takip edilebilmesine imkan tanıyan bir haritada hazırlanmıştır, DoxMap. Kritik öneme sahip olup anlık takip edilmesi gereken dokümanlar için düşünülmüş olan bu harita üzerinde gerçekleşen olaylar anlık olarak gösterilmektedir. Kullanıcı, sadece DoxMap'ı açarak gün boyu başka hiç birşey yapmadan hangi dokümanların nereden açıldığını canlı olarak takip edebilmektedir. DoxMap'e ait ekran görüntüsü Şekil 4.28'de yer almaktadır.



ŞEKİL 4.28: Canlı doküman takibi için kullanılan DoxMap haritası

Bölüm 5

Örnek Kullanım Senaryoları

Bu bölümde, önerilen sistemin kullanım alanları ve amaçları örnek bir kaç senaryo altında gösterilmiştir.

- *Senaryo-1*: Günümüz siber dünyasında gerçekleşen saldırıların boyutu ve çeşitliliği her geçen gün hızla artmaktadır. Gerçekleşebilecek siber saldırı olasılıklarına karşı gerekli önlemlerin alınmadığı ve hazırlıkların yapılmadığı durumlarda çoğu zaman çaresiz kalınmaktadır. Bundan dolayı önceden düşünüp gerekli önlemleri almak gerekmektedir. Bu noktada yapılacak çalışmaların savunma derinliğini arttırmaya yönelik olmasına da dikkat etmek gerekir. Savunma derinliğini arttırmanın en iyi yollarından birisi tuzak doküman kullanmaktır. DoxTracker kullanılarak çeşitli özellik ve içeriklere sahip tuzak dokümanlar hazırlanabilir. Hazırlanan bu tuzak dokümanlar bilgisayar ve dosya sisteminin çeşitli dizinlerine yerleştirilerek özellikle saldırı tespit sistemleri tarafından tespit edilemeyen saldırıların tespit edilmesi noktasında güvenlik alarm mekanizması olarak kullanılabilir. Örneğin saldırı motivasyonu sistem kullanıcılarına ait kullanıcı adı ve parolaları elde etmek olan bir bilgisayar korsanını tuzağa düşürmek için "sunucu hesap bilgileri" veya "kullanıcı hesap bilgileri" gibi isimler kullanılarak tuzak doküman hazırlanabilir. Hazırlanan bu tuzak dokümanlar saldırganın kolayça erişilebileceği bir dosya dizinine yerleştirilerek saldırgan tuzağa düşürülebilir. Böylece hem doküman açılması ile hem de sahte kullanıcı adı ve parolaların sisteme erişim için kullanılması ile sisteme düşen log kayıtlardan bir saldırı olduğu tespit edilebilir.

- *Senaryo-2*: DoxTracker kullanılarak gizli dokümanlara sadece ilgili kişiler tarafından erişilebildiğinden emin olunabilir. DoxTracker tarafından korunan gizli bir doküman kim tarafından ne zaman açıldığı bilgisi ile takip edilebiliyor iken doküman oluşturulurken belirtilen erişim kısıtlamaları ve içeriğin şifrelenerek korunmasından dolayı yetkisi olmayan kişilerce görüntülenemeyecektir.

  Bu senaryo gizli dokümanların kötü niyetli kişiler tarafından görüntülenmek istenmesi ve istem dışı/yanlışlıkla gizli dokümanların açığa çıkması olarak iki farklı bakış açısı altında değerlendirilebilir. Şöyleki; 2014'ün en popüler mobil oyunlarından biri olan 'Game of War:Fire Age' oyununun üreticisi olan Machine Zone firmasında Küresel Altyapı Müdürü olarak çalışmakta olan Jing Zeng, daha yüksek bir kıdem tazminatı alabilmek için şantaj yapmak amacıyla şirkete ait ticari sırları çalmıştır. Machine Zone firması bu durumdan şantaj sırasında haberdar olmuştur. Böyle bir durumda kritik öneme sahip olan dokümanlar DoxTracker gibi bir sistem tarafından korunuyor ve takip edilebiliyor olsaydı hem gerçekleşen sızmadan haberdar olunabilecek hem de ticari sırların başkasının eline geçmesi gibi bir ihtimal olmayacaktı. Öte yandan yakın zamanda TÜBİTAK iç ağı yanlışlıkla kısa bir süreliğine internet ortamına açılmıştı. Bir dokümanın kısa bir süreliğine dahi internet ortamına düşmesi, bu dokümanın önizlemesinin Google tarafından hafızaya alınması için yeterlidir. Dolayısı ile çok geçmeden bir çok TÜBİTAK çalışanının kişisel bilgileri Google aramalarında önizleme şeklinde erişilebilir duruma gelmiştir. Bu gibi durumlar düşünüldüğünde de eğer dokümanlar DoxTracker tarafından koruma altına alınırsa internet ortamında sadece DoxTracker karşılama sayfası görünür olacaktır.

- *Senaryo-3*: Bir çok firma işe alacağı yeni elemanını seçerken adayın güvenilirliğine dair yeteri kadar inceleme yapmamaktadır. İşe alım sürecinde tabi tutulan ve genellikle adayın güçlü ve zayıf yönlerini ortaya çıkarmak amacıyla yapılan personel profil analizleri üzerinden çalışanlarını tanımaya/tanımlamaya çalışmaktadır. Fakat konu bilgi güvenliği olunca ve Amerikan Savunma Endüstrisinde normal bir çalışan olarak görülen fakat içeriden bilgi sızdıran Çinli Chi Mak örneği [12] gibi bugüne kadar yaşanmış olan güvenlik ihlali olayları düşünülünce bu kadarlık bir çalışmanın ne kadar yetersiz olduğunu söylemek çok da zor olmasa gerek. Bundan dolayı basit, ek getirisi olmayan, kullanımı da kolay olan DoxTracker aracılığı ile hazırlanacak tuzak dokümanların kimler tarafından, ne sıklıkla, ne zaman açıldığı;

herhangi bir kopyalama ve düzenleme yapılıp yapılmadığı gibi bilgilerin korelasyonu ile daha sağlıklı çalışan profilleri çıkartılabilir. Böylelikle iç tehdit kaynaklı yaşanabilecek olaylara karşı da önceden tedbir alınmış olunur.

- *Senaryo-4*: Son yıllarda yaşanan olaylardan dolayı siber güvenlik farkındalığının artması ve bu alanda artık çeşitli standartların uygulanmasının devlet eliyle zorunlu tutulması gibi etkenler bilgi güvenliğine verilen önemin artmasını sağlamıştır. Gelişen teknolojiye ayak uydurmak zorunda olan iş dünyasında, kurumsal bilgilerin dijitalleşmesi bu bilgilerin ifşası veya çalınması gibi daha sayılabilecek onlarca kurumsal bilgi güvenliği risklerinin ortaya çıkmasına sebeb olmaktadır. Bu risklerin en aza indirgenebilmesi veya mümkün olanların tamamen ortadan kaldırılabilmesi amacıyla belirli periyotlarla sızma testleri gerçekleştirilir. Çoğu şirket bu testleri yapabilecek kalifiye elemana sahip değildir. Onun için bu testler genellikle dışarıdan farklı bir firmadan destek alınarak gerçekleştirilir. Sızma testlerini gerçekleştirilecek olan dış kaynaklı firma ile her ne kadar karşılıklı bir gizlilik anlaşması imzalanıyor olsa da bu testler sırasında şirkete ait ticari sırlar veya önemli dokümanlar test işlemini gerçekleştiren personelin eline geçebilir. Testi gerçekleştiren ve şirket için kritik öneme sahip dokümanlara ulaşan dış kaynaklı firma personeli o gün için olmasa da gelecekte sorun teşkil edebilecek potansiyel bir tehlike konumundadır. Bu gibi durumlarda akıllarda soru işareti kalmaması adına ya kritik öneme sahip olan tüm dokümanlar sistemden toplanarak farklı bir yere aktarılmalı ya da DoxTracker kullanılarak tüm dokümanlar güvence altına alınarak takip edilebilir.

- *Senaryo-5*: Bu senaryoda ise güvenlik amaçlı olmayan gündelik bir ihtiyaç dahilinde DoxTracker sisteminin kullanımından bahsedilmiştir. Çoğu zaman hazırlanan dokümanların veya çalışmaların ilgili kişiler tarafından incelenip incelenmediği noktasında tereddütler yaşanır. Bu gibi durumlarda dokümanı incelemesi beklenilen kişinin haberi dahi olmadan, dokümanı inceleyip incelemediği DoxTracker kullanılarak takip edilebilir.

Bölüm 6

Sonuç ve Gelecek Çalışmalar

Çalışma kapsamında; MS Word ve PDF dokümanlarının içeriğini koruyan, uygulama seviyesinde güvenlik sağlayarak mevcut saldırı tespit ve önleme sistemlerine destek olan, iç tehdit tespiti noktasında çalışanların güvenilirlik profillerinin çıkartılabilmesini sağlayan bir sistem önerilmiş ve prototip seviyesinde geliştirilmesi de yapılmıştır.

Günümüzde en iyi teknolojik altyapıya ve beyin gücüne sahip devletlerin dahi çoğu zaman siber saldırılar karşısında çaresiz kaldığı bir çok kez gözler önüne serilmiştir. Bu yüzden %100 güvenli bir sistemin olamayacağı varsayımından da yola çıkılarak maksimum güvenlik seviyesine ulaşabilmek amacıyla saldırı yüzeyi ve savunma derinliği arttırılmalıdır. Klasik yöntemlerin pek çok zaman yetersiz kaldığı bilgisayar korsanları ve gelişmiş siber saldırılar(APT) karşısında, en etkili savunma ve uyarı mekanizmalarından birisi dokümanların açılıp açılmadığının takip edilmesidir. Bundan dolayı günümüz savunma teknolojilerinin yanında DoxTracker gibi bir güvenlik önleminin de kullanılması faydalı olacaktır.

Geliştirilmesi devam etmekte olan DoxTracker'ın, sonraki versiyonlarda eklenecek olan özelliklerle beraber daha verimli kullanılabilmesi amaçlanmaktadır. Bu hedef doğrultusunda aşağıdaki özelliklerin sisteme entegrasyonu düşünülmektedir:

- Farklı formattaki doküman tiplerinin (MS Excel, MS PowerPoint, vb.) desteklenmesi,
- Bilgilendirme sistemi altyapısının geliştirilmesi ile SMS gibi farklı alternatiflerin sunulması,

- Raporlama altyapısının geliştirilmesi (En fazla açılan dokümanlar, Dokümanların açılma sıklıkları, Dokümanların en çok açıldığı konumlar, vb.),
- İnternet olmayan ortamlarda da doküman takibinin sağlanabilmesi,
- Otomatik olarak tuzak doküman oluşturulabilmesi (Sahte erişim bilgileri içeren doküman, Sahte faturalar, vb.)

Kaynaklar

[1] L. Boettger. The morris worm: How it affected computer security and lessons learned by it, 2000. URL http://www.giac.org/paper/gsec/405/morris-worm-affected-computer-security-lessons-learned/100954.

[2] W. Howe. A brief history of the internet, 2014. URL http://www.walthowe.com/navnet/history.html.

[3] Internet Live Stats. Internet users in the world by year, 2014. URL http://www.internetlivestats.com/internet-users/#trend.

[4] TÜİK. Hanehalkı bilişim teknolojileri kullanım araştırması, 2015, 2015. URL http://www.tuik.gov.tr/PreHaberBultenleri.do?id=18660.

[5] Privacy Rights Clearinghouse (PRC). Chronology of data breaches security breaches 2005 - present, 2015. URL http://www.internetlivestats.com/internet-users/#trend.

[6] D Elliott Bell and Leonard J LaPadula. Secure computer systems: Mathematical foundations. Technical report, DTIC Document, 1973.

[7] David D Clark and David R Wilson. A comparison of commercial and military computer security policies. In *Security and Privacy, 1987 IEEE Symposium on*, pages 184–184. IEEE, 1987.

[8] Computer Security Institute (CSI). Computer crime and security survey, 2006. URL http://pdf.textfiles.com/security/fbi2006.pdf.

[9] C.W. Probst, J. Hunker, M. Bishop, and D. Gollmann. *Insider Threats in Cyber Security*. Advances in Information Security. Springer US, 2010. ISBN 9781441971333. URL https://books.google.com.tr/books?id=MCoYPjRAEAEC.

[10] the free encyclopedia Wikipedia. Edward snowden, 2015. URL `https://tr.wikipedia.org/wiki/Edward_Snowden`.

[11] C. Preimesberger. The seven largest insider-caused data breaches of 2014, 2014. URL `http://www.eweek.com/security/slideshows/the-seven-largest-insider-caused-data-breaches-of-2014.html`.

[12] FBI. An introduction to detecting and deterring an insider spy, 2015. URL `https://www.fbi.gov/about-us/investigate/counterintelligence/the-insider-threat`.

[13] M. J. Schwartz. 'game of war: Fire age' insider arrested, 2015. URL `http://www.bankinfosecurity.com/game-war-fire-age-insider-arrested-a-8501/op-1`.

[14] Cloud Security Alliance. Top threats to cloud computing v1.0, 2010. URL `https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf`.

[15] F. Beals. The insides of several high-profile accounts have been exposed, 2009. URL `http://news.softpedia.com/news/Twitter-Admin-Account-Hacked-via-Social-Engineering-110694.shtml`.

[16] Cliff Stoll. *The cuckoo's egg: tracking a spy through the maze of computer espionage*. Simon and Schuster, 2005.

[17] Lance Spitzner. Honeypots: Catching the insider threat. In *Computer Security Applications Conference, 2003. Proceedings. 19th Annual*, pages 170–179. IEEE, 2003.

[18] Lance Spitzner. Honeytokens: The other honeypot, 2003.

[19] Jim Yuill, Mike Zappe, Dorothy Denning, and Fred Feer. Honeyfiles: deceptive files for intrusion detection. In *Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC*, pages 116–122. IEEE, 2004.

[20] Malek Ben Salem, Shlomo Hershkop, and Salvatore J Stolfo. A survey of insider attack detection research. In *Insider Attack and Cyber Security*, pages 69–90. Springer, 2008.

[21] Brian M Bowen, Shlomo Hershkop, Angelos D Keromytis, and Salvatore J Stolfo. *Baiting inside attackers using decoy documents*. Springer, 2009.

[22] Brian M Bowen, Malek Ben Salem, Angelos D Keromytis, and Salvatore J Stolfo. Monitoring technologies for mitigating insider threats. In *Insider Threats in Cyber Security*, pages 197–217. Springer, 2010.

[23] Malek Ben Salem and Salvatore J Stolfo. Decoy document deployment for effective masquerade attack detection. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 35–54. Springer, 2011.

[24] Spinellis D. Prevelakis, V. The athens affair, 2007. URL `http://spectrum.ieee.org/telecom/security/the-athens-affair`.

[25] James P Anderson. Computer security threat monitoring and surveillance. Technical report, Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, 1980.

[26] U. Kaya and Ö. Erdem. Saldırı tespit sistemleri (snort, suricata, bro), 2014. URL `https://www.bilgiguvenligi.gov.tr/saldiri-tespit-sistemleri/saldiri-tespit-sistemleri-snort-suricata-bro.html`.

[27] The Free Encyclopedia Wikipedia. Cloud computing, 2015. URL `https://en.wikipedia.org/wiki/Cloud_computing`.

[28] B. Skvorc. Best php frameworks for 2014, 2013. URL `http://www.sitepoint.com/best-php-frameworks-2014/`.

[29] B. Skvorc. The best php framework for 2015, 2015. URL `http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/`.

[30] F. Erdoğan. Pdf malware analiz teknikleri, 2014. URL `http://ferdogan.net/2014/11/06/PDF-Malware-Analiz-Teknikleri/`.