
GAZİ ÜNİVERSİTESİ
Gazi
1926

SİBER TEHDİT İSTİHBARATI İÇEREN HETEROJEN BİLGİ AĞLARININ ÜRETİMİ VE ZENGİNLEŞTİRİLMESİ

Burak GÜLBAY

DOKTORA TEZİ
BİLGİ GÜVENLİĞİ MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EYLÜL 2024

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Burak GÜLBAY

11/09/2024

SİBER TEHDİT İSTİHBARATI İÇEREN HETEROJEN BİLGİ AĞLARININ ÜRETİMİ VE ZENGİNLEŞTİRİLMESİ

(Doktora Tezi)

Burak GÜLBAY

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Eylül 2024

ÖZET

Siber uzayda kişiler, şirketler, kurumlar ve devletler dijital varlıklarını saklamakta, işlemekte ve değer üretmek için hizmetler sunmaktadır. En değerli varlıklarımız artık fiziksel varlıklarımızdan çok sahip olduğumuz veri ve bilgi haline gelmiştir. Bilginin değeri, kullanıcısına sayladığı faydadan kaynaklanmaktadır. Günümüzde dijital dönüşümle birlikte, bankacılık, seçim, sınav, enerji, sağlık ve savunma gibi hayatımızı doğrudan etkileyen sistemler gerçek dünyanın karmaşıklığından soyutlanarak bilişim sistemlerine taşınmıştır. Bu sistemler siber uzaydaki bağımsız ve devlet destekli tehdit aktörleri için hedef haline gelmiştir. Savunma tarafındaki yöntemler geliştikçe, saldırı tarafında da gelişmiş ve karmaşık yöntemler ortaya çıkmaktadır. Saldırganlar ve eylemleri henüz bilişim sistemlerine temas etmeden, onlar hakkında bilgi sahibi olmak ve önlem almak güncel siber güvenlik disiplininin önemli bir parçasıdır. Bu tez çalışmasında gelişmiş kalıcı tehdit (APT) aktörlerinin arkalarında bıraktığı izler sürülerek, haklarında yazılan APT analiz raporları ve İnternetteki açık kaynak istihbarat (OSINT) kaynakları ile zenginleştirilerek Siber Tehdit İstihbaratı (STİ) üretilmiş ve sistematik bir şekilde Heretojen Bilgi Ağına (HBA) dönüştürülmesi sağlanmıştır. Önerilen çok katmanlı çerçeve model implemente edilerek 734 analiz raporu, 484.249 düğüm ve 675.696 ilişki üzerinde çalışılmıştır. Savunma tarafındaki kişilerin düşünce dünyası, güvenlik ürünleri, araçları, kurallar ve alarmlar tablolar halinde tasarlanmakta ve yönetilmektedir. Öte yandan saldırganların zihin haritaları çizgelere dayanmaktadır. HBA'ların, saldırganları ve saldırıları anlamak için en iyi yöntem olduğu değerlendirilmektedir. Tez çalışmasında toplanan operasyonel, taktik ve teknik STİ, HBA'ya gönüştürülmüş ve bir tahminleme modeli geliştirilerek ağ üzerinde tahminler yapılmıştır. Geliştirilen tahminleme modelinin doğruluk-duyarlılık eğrisi altındaki alan (AUCPR) ölçümlerinde eğitim skoru %96,36 ve test skoru %97,88 elde edilmiştir. APT grupları arasında takma ad ilişkileri ve faili meçhul saldırıların olası aktörleri tahmin edilmiştir. Önerilen çok aşamalı çerçeve modelin başarısını göstermek için uygulanan aşamalar detaylarıyla sunulmuş ve uygulanarak deney sonuçları paylaşılmıştır.

Bilim Kodu	:	92403
Anahtar Kelimeler	:	Siber Tehdit İstihbaratı, (STİ), Heterojen Bilgi Ağları, (HBA), Çizge, Gelişmiş Kalıcı Tehdit, (GKT)
Sayfa Adedi	:	132
Danışman	:	Doç. Dr. Mehmet DEMİRCİ

PRODUCTION AND ENRICHMENT OF HETEROGENEOUS INFORMATION NETWORKS CONTAINING CYBER THREAT INTELLIGENCE

(Ph. D. Thesis)

Burak GÜLBAY

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

September 2024

ABSTRACT

In cyberspace, individuals, companies, institutions, and governments store and process their digital assets, and provide services to create value. Our most valuable assets have now become our data and information, rather than physical assets. The value of information comes from the benefit it provides to its consumer. Today, with digital transformation, systems that directly affect our lives, such as banking, elections, exams, energy, health, and defense, have been transferred to information systems, abstracting from the complexity of the real world. These systems have become targets for independent and state-sponsored threat actors in cyberspace. As defensive methods evolve, so do advanced and complex attack techniques. Obtaining information about attackers and their actions before they even touch information systems, and taking precautions, is a crucial part of modern cybersecurity discipline. In this thesis, Cyber Threat Intelligence (CTI) was produced by tracking the traces left by Advanced Persistent Threat (APT) actors, enriching it with APT analysis reports written about them and open-source intelligence (OSINT) sources from the Internet, and systematically converting it into a Heterogeneous Information Network (HIN). The proposed multi-layered framework model was implemented and worked on 734 analysis reports, 484,249 nodes, and 675,696 relationships. The thought processes of defense-side individuals, security products, tools, rules, and alarms are designed and managed in tabular form. On the other hand, the mental maps of attackers are based on graphs. It is considered that HINs are the best method for understanding attackers and their attacks. In this thesis, operational, tactical, and technical CTI collected was converted into HIN, and a prediction model was developed to make predictions over the network. In the Precision-Recall Curve Under the Area (AUCPR) measurements of the developed prediction model, a training score of 96.36% and a test score of 97.88% were obtained. Alias relationships among APT groups and the likely actors of unsolved attacks were predicted. The stages implemented to demonstrate the success of the proposed multi-stage framework model are presented in detail, and the experimental results are shared.

Science Code	:	92403
Key Words	:	Cyber Threat Intelligence, (CTI), Heterogeneous Information Networks, (HIN), Graph, Advanced Persistent Threat, (APT)
Page Number	:	132
Supervisor	:	Assoc. Prof. Dr. Mehmet DEMİRCİ

TEŞEKKÜR

Bu tez çalışmasında fikirleriyle rehberlik eden, yolu aydınlatan ve desteklerini esirgemeyen değerli tez danışmanım Sayın Doç. Dr. Mehmet DEMİRCİ'ye ve onunla birlikte birikim ve katkılarıyla benden yardımlarını esirgemeyen değerli Tez İzleme Komitesi üyeleri Prof. Dr. Ahmet Burak CAN'a ve Doç. Dr. İbrahim Alper DOĞRU'ya teşekkür ederim.

Hayatım boyunca her zaman yanımda olan ağabeyim Dr. Yasin GÜLBAY'a, sevgili eşime ve çocuklarıma destek ve sabırlarından dolayı teşekkür ederim. Sevgili Cahit Taha ve Zahit Bera büyüdüklerinde, yaptığım çalışmaların akıl ve bilim yolunda iyilik ile ilerlemeleri için onlara manevi destek olmasını dilerim.

Çalışmamı, hayatımda yokluğunun bıraktığı boşluğu asla dolduramayacağım, sevgi ve özlemle andığım anneme adıyorum.

İÇİNDEKİLER

Sayfa

ÖZET iv

ABSTRACT v

TEŞEKKÜR vi

İÇİNDEKİLER vii

ÇİZELGELERİN LİSTESİ ix

ŞEKİLLERİN LİSTESİ xi

SİMGELER VE KISALTMALAR xii

1. GİRİŞ 1

2. SİBER TEHDİT İSTİHBARATI VE HETEROJEN BİLGİ AĞLARI 11

2.1. Alan Bilgisi ve Ontoloji Oluşturulması 12

2.2. Kavramlar ve Terimler 15

2.3. Siber Tehdit İstihbaratı Döngüsü 17

2.4. Siber Tehdit İstihbaratı Paylaşımı 21

2.4.1. Veri yapısı ve paylaşım standartları 21

2.4.2. STIX 22

2.4.3. TAXII 25

2.5. Heterojen Bilgi Ağları 26

3. LİTERATÜRDEKİ ÇALIŞMALAR 29

3.1. Teknik STİ Üzerine Çalışmalar 30

3.2. Taktik STİ Üzerine Çalışmalar 40

3.3. Operasyonel STİ Üzerine Çalışmalar 46

3.4. Stratejik Siber Tehdit İstihbaratı Üzerine Çalışmalar 54

4. ZENGİNLEŞTİRİLMİŞ STİ İÇEREN HBA ÜRETİMİNE YÖNELİK ÇERÇEVE MODELİ ÖNERİSİ 59

4.1. APT Analiz Raporlarının İşlenmesi 61

Sayfa

4.2. Blok Listelerinin İşlenmesi ve STİ Zenginleştirme 66

4.3. Aktif Veri Toplama ve STİ Zenginleştirme 70

5. ÇİZGE TABANLI TAHMİNLEME MODELİ GELİŞTİRİLMESİ 77

5.1. Nihai HBA'nın Çizge Algoritmaları ile Analizi 78

5.1.1. Düğümlerin merkezilik analizi 78

5.1.2. Topluluk tespiti analizi 84

5.2. Tahminleme Modelinin Eğitilmesi 90

6. ÖNERİLEN ÇERÇEVE MODELİN VE TAHMİNLEME MODELİNİN DEĞERLENDİRİLMESİ 99

7. SONUÇLAR VE ÖNERİLER 111

KAYNAKLAR 117

ÖZGEÇMİŞ .....**Hata! Yer işareti tan**

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 2.1. STIX alan nesneleri listesi ve tanımları 12
Çizelge 2.2. STİ alanındaki ontolojilerin kapsamları 13
Çizelge 2.3. STİ paylaşım standartları ve kapsadıkları bilgi varlıkları 22
Çizelge 2.4. STIX V2.1 alan nesneleri listesi ve tanımları 23
Çizelge 2.5. STIX V2.1 ilişki listesi ve tanımları 25
Çizelge 3.1. Teknik STİ alanındaki çalışmalar 30
Çizelge 3.2. Taktik STİ alanındaki çalışmalar 41
Çizelge 3.3. Operasyonel STİ alanındaki çalışmalar 46
Çizelge 3.4. İnternet haberlerinden elde edilen özellik ve değer örnekleri 53
Çizelge 3.5. Stratejik STİ alanındaki çalışmalar 55
Çizelge 4.1. APT analiz raporlarında etiketlenmiş ve toplam varlık sayıları 64
Çizelge 4.2. APT analiz raporlarından elde edilen düğüm türleri ve sayıları 65
Çizelge 4.3. APT analiz raporlarından elde edilen ilişki türleri ve sayıları 65
Çizelge 4.4. Etkin tehlikeli ve etkisizleştirilmiş adres örnekleri 66
Çizelge 4.5. Blok listesi kaynakları, içerdikleri STİ'nin türü ve içeriği 66
Çizelge 4.6. Blok listelerin entegre edildiği HBA'daki düğüm türleri ve sayıları 67
Çizelge 4.7. Blok listelerin entegre edildiği HBA'daki ilişki türleri ve sayıları 69
Çizelge 4.8. Aktif veri toplama kaynakları, içerdikleri STİ'nin türü ve içeriği 70
Çizelge 4.9. IP çözümlemede kullanılan DNS sunucuları 71
Çizelge 4.10. DDoS koruma ve WAF hizmetlerinin IP adresleri 72
Çizelge 5.1. Çizgeler üzerinde en kısa yolu bulma algoritmaları 79
Çizelge 6.1. HBA üzerindeki tüm düğümlerin Aradalık Merkeziliği istatistiği 100
Çizelge 6.2. HBA üzerindeki düğümlerin Aradalık Merkeziliği skorları 101
Çizelge 6.3. HBA üzerindeki tüm düğümlerin Yakınlık Merkeziliği istatistiği 101
Çizelge 6.4. HBA üzerindeki APT gruplarının Yakınlık Merkeziliği skorları 101

Çizelge **Sayfa**

Çizelge 6.5. HBA üzerindeki APT gruplarının Derece Merkeziliği istatistiği 102

Çizelge 6.6. HBA üzerindeki düğümlerin Derece Merkeziliği skorları 102

Çizelge 6.7. HBA üzerindeki APT Gruplarının k-çekirdek kullanılarak kümelenmesi . 103

Çizelge 6.8. HBA üzerinde Etiket Yayılımı hesaplamasına ait istatistikler 104

Çizelge 6.9. HBA üzerindeki APT gruplarının Etiket Yayılımı kümelenmesi..... 104

Çizelge 6.10. HBA üzerinde Leiden algoritmasına ait istatistikler 104

Çizelge 6.11. HBA üzerindeki APT gruplarının Leiden kümelenmesi 104

Çizelge 6.12. ALSO_KNOWN_AS ilişkisi tahmin modeli adaylarının AUCPR skoru 105

Çizelge 6.13. ALSO_KNOWN_AS ilişkilerini gösteren ilk 20 tahmin 106

Çizelge 6.14. HAS_APT_GROUP ilişkisi tahmin modeli adaylarının AUCPR skoru .. 107

Çizelge 6.15. HAS_APT_GROUP ilişkilerini gösteren ilk 20 tahmin 108

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. Kabiliyet odaklı STİ modeli 15

Şekil 2.2. STİ yaşam döngüsü 18

Şekil 2.13. Güvenlik alt alanları ve aralarındaki etkileşim 19

Şekil 2.4. TAXII'de sunulan STİ paylaşma haberleşme modelleri 25

Şekil 3.1. STİ türlerinin geçerlilik eğilimleri ve kullanım ömürleri 29

Şekil 3.2. Zararlı IP trafiği tespit yöntemi 36

Şekil 3.3. STİ güvenilirliği belirleme modeli 40

Şekil 3.4. Çevrim içi hesaplama fazları 49

Şekil 3.5. Çevrim dışı hesaplama fazları 50

Şekil 3.6. a) Metin ön işleme b) Siber saldırı özelliklerinin tanımlanması c) CRF ve LSA kullanarak internet haberi sınıflandırma 52

Şekil 3.7. Siber tehdit olgunluk seviyesi modeli 56

Şekil 4.1. Toplanan STİ'den HBA üretilmesi ve zenginleştirilmesi 59

Şekil 4.2 STİ toplama, zenginleştirme, HBA üretimi ve bilişenler arasındaki veri akışına ilişkin üst düzey sistem mimarisi 60

Şekil 4.3. Uygulanan NER işleminin sözde çıktısı 63

Şekil 4.4. HBA'nin kısmi görünümü 75

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
ms	Milisaniye

Kısaltmalar	**Açıklamalar**
ABD	Amerika Birleşik Devletleri
API	Application Programming Interface (Uygulama Programlama Arayüzü)
APT	Advanced Persistent Threat (Gelişmiş Kalıcı Tehdit)
ARM	Association Rule Mining (Bağlantı Kuralı Madenciliği)
AS	Autonomous System (Otonom Sistem)
ASN	Autonomous System Number (Otonom Sistem Numarası)
ATT&ACK	Adversarial Tactics, Techniques and Common Knowledge (Düşman Taktikleri, Teknikleri ve Ortak Bilgisi)
AUCPR	Area Under the Precision-Recall Curve (Doğruluk -Duyarlılık Eğrisi Altındaki Alan)
C&C	Command & Control (Komuta Kontrol)
CIDR	Classless Inter-Domain Routing (Sınıfsız Alan Adları Arası Yönlendirme)
CIRCL	Computer Incident Response Center Luxembourg (Bilgisayar Olay Müdahale Merkezi Lüksemburg)
CiviCERT	Computer Incident Response Center for Civil Society (Sivil Toplum için Bilgisayar Olayı Müdahale Merkezi)
CLF	Common Log Format (Ortak Günlük Formatı)
COVID-19	Corona Virus Disease of 2019 (Corona Virüs Hastalığı 2019)
CRF	Conditional Random Field (Koşullu Rastgele Alan)
CTI	Cyber Threat Intelligence (Siber Tehdit İstihbaratı)
CTITC	Cyber Threat Intelligence Technical Committee (Siber Tehdit İstihbaratı Teknik Komitesi)

Kısaltmalar	Açıklamalar
CVE	Common Vulnerabilities and Exposures (Bilinen Zaafiyetler ve Açıklar)
DDoS	Distributed Denial of Service (Dağıtık Hizmet Engelleme)
DGA	Domain Generation Algorithm (Alan Adı Oluşturma Algoritması)
DNS	Domain Name System (Alan Adı Sistemi)
DoS	Denial of Service (Hizmet Kesintisi)
EWP	Element Wise Product (Eleman Bazlı Çarpım)
FastRP	Fast Random Projection (Hızlı Rastgele Projeksiyon)
FP-Growth	Frequent-Pattern Growth (Sık Desenli Büyüme)
FQDN	Fully Qualified Domain Name (Tam Nitelikli Alan Adı)
GMM	Gaussian Mixture Model (Gaus Karışım Modeli)
HIN	Heterogeneous Information Networks (Heterojen Bilgi Ağları)
HTTPS	Hypertext Transfer Protocol Secure (Köprü Metni Aktarım Protokolü Güvenli)
HUMINT	Human Intelligence (İnsan İstihbaratı)
ICAS	Integrated Cyber Analysis System (Entegre Siber Analiz Sistemi)
IDS	Intrusion Detection System (Saldırı Tespit Sistemi)
IoC	Indicator of Compromise (Güvenlik İhlali Göstergesi)
IoT	Internet of Things (Nesnelerin İnterneti)
IPv4	Internet Protocol version 4 (İnternet Protokolü versiyon 4)
IPv6	Internet Protocol version 6 (İnternet Protokolü versiyon 6)
JSON	JavaScript Object Notation
LDA	Latent Dirichlet Allocation (Gizli Dirichlet Tahsisi)
LLM	Large Language Model (Büyük Dil Modeli)
MAEC	Malware Attribute Enumeration and Characterization (Kötü Amaçlı Yazılım Nitelik Sayımı ve Karakterizasyonu)
MD5	Message Digest Method 5 (Mesaj Özeti Yöntemi 5)

Kısaltmalar	Açıklamalar
MISP	Malware Information Sharing Platform (Zararlı Yazılım Bilgi Paylaşım Platformu)
MITRE	Massachusetts Institute of Technology Research and Engineering (Massachusetts Teknoloji Araştırma ve Mühendislik Enstitüsü)
MM-TBM	Markov Multi-phase Transferable Belief Model (Markov Çok Aşamalı Aktarılabilir Düşünce Modeli)
NATO	North Atlantic Treaty Organization (Kuzey Atlantik Antlaşması Teşkilatı)
NER	Named Entity Recognition (Varlık İsmi Tanıma)
NIST	National Institute of Standards and Technology (Ulusal Standartlar ve Teknoloji Enstitüsü)
NLP	Natural Language Processing (Doğal Dil İşleme)
NoSQL	Not Only SQL
NRO	National Reconnaissance Office (Ulusal Keşif Ofisi)
OASIS	Organization for the Advancement of Structured Information Standards (Yapılandırılmış Bilgi Standartlarının Geliştirilmesi Organizasyonu)
OpenIOC	Open Indicators of Compromise (Açık Tehdit Göstergeleri)
OS	Olgunluk Seviyesi
OSINT	Open Source Intelligence (Açık Kaynak İstihbarat)
OVM	Ontology for Vulnerability Management (Güvenlik Açığı Yönetimi için Ontoloji)
PGP	Pretty Good Privacy
REGEX	Regular Expression (Kurallı İfadeler)
REST	Representational State Transfer (Temsili Durum Transferi)
RIPE	Réseaux IP Européens
ROC-AUC	Receiver Operating Characteristic - Area Under the Curve (Alıcı işletim Karakteristiği - Eğri Altındaki Alan)
SDO	STIX Domain Objects (STIX Alan Nesneleri)
SHA1	Secure Hash Algorithm 1 (Güvenli Hash Algoritması 1)
SHA256	Secure Hash Algorithm 256-bit (Güvenli Hash Algoritması 256-bit)

Kısaltmalar	Açıklamalar
SIEM	Security Information and Event Management (Güvenlik Bilgileri ve Olay Yönetimi)
SOC	Security Operations Center (Güvenlik Operasyon Merkezi)
SQL	Structured Query Language (Yapısal Sorgulana Dili)
SRO	STIX Relationship Objects (STIX İlişki Nesneleri)
SSH	Secure Socket Shell (Güvenli Soket Kabuğu)
SSL	Secure Sockets Layer (Güvenli Soket Katmanı)
STIX	Structured Threat Information eXpression (Yapılandırılmış Tehdit Bilgi İfadesi)
SVM	Support Vector Machines (Destek Vektör Makineleri)
TAG	Google Threat Analysis Group
TAXII	Trusted Automated eXchange of Intelligence Information (Güvenilir Otomatik İstihbarat Bilgi Değişimi)
TCP	Transmission Control Protocol (İletim Kontrol Protokolü)
TIG	Trellix Threat Intelligence Group
TTP	Tactics, Techniques and Procedures (Taktikler, Teknikler ve Prosedürler)
UCO	Unified Cybersecurity Ontology (Birleşik Siber Güvenlik Ontolojisi)
URL	Uniform Resource Locator (Tekdüze Kaynak Bulucu)
USDHS	United States Department of Homeland Security (Amerika Birleşik Devletleri İç Güvenlik Bakanlığı)
USOM	Ulusal Siber Olaylara Müdahale Merkezi
UUID	Universally Unique Identifier (Evrensel Benzersiz Tanımlayıcı)
VERIS	Vocabulary for Event Recording and Incident Sharing (Olay Kaydı ve Olay Paylaşımı için Sözlük)
XML	Extensible Markup Language (Genişletilebilir İşaretleme Dili)

1. GİRİŞ

Devletlerin, şirketlerin ve gerçek kişilerin güvenlik algısı tarih boyunca konvansiyonel tehditlere karşı alınan önlemler üzerine kuruluydu. Kara, hava, deniz ve uzayda var olmak ve varlığını devam ettirmek isteyenler, bu ortamların şartlarına uygun streteji geliştirmek, rekabette üstünlük getirecek avantajlar elde etme çabasındaydı. Bu bağlamda fiziksel olarak bilinen ve keşfedilen ortamlarda ilk var olma isteği çeşitli riskleri de beraberinde getirmektedir. Riskler yapısında tehditler ve fırsatlar barındırmaktadır [1]. Fırsatların belirlenmesi ve kullanılması stratejik karar almayı gerektirmektedir. Kara, deniz, hava ve uzaydaki varlıklarını inşa eden ve güçlenen devletler konvansiyonel tehditlere ve fırsatlara karşı farkındalıklarını artırmıştır. Aynı zamanda devletlerin ve şirketlerin uzayda öncü ve egemen güç olma çabaları henüz bilinmeyen fırsatların ve buna bağlı stratejik üstünlüğün elde edilme isteğinden kaynaklanmaktadır. Örneğin, Amerika Birleşik Devletleri Savunma Bakanlığı'na bağlı olarak faaliyet gösteren ulusal keşif ofisi (national reconnaissance office, NRO) uzayda etkin güç olmanın stratejik üstünlüğe yansımasına iyi bir örnektir.

Stratejik üstünlüğün sağlanması için tehdit ve fırsatlara karşı durumsal farkındalık elde edilmesi gerekmektedir. Durumsal farkındalığın oluşması için tehdit ve fırsatlara dair istihbarat elde edilmesi gerekmektedir. Siber uzayda var olmanın da peşinde getirdiği riskler bulunmaktadır. Siber uzaydaki tehditler, geleneksel ortamlara kıyasla oldukça karmaşık ve fark edilmesi zordur [2]. İnternetin hayatımıza girmesiyle birlikte, insanların, kurumların, şirketlerin ve devletlerin bu ağ ile etkileşimleri kaçınılmaz hale gelmiştir. İnternette ilk var olanlar yine stratejik üstünlüğü elde etmiş durumdadır. Strateji bilimine göre, iyi strateji geliştirmek için planlama yapılması, şaşırtma, örüntülerin keşfedilmesi, kendini konumlandırma ve perpektif kazanılması gerekmektedir [3].

Dijital dünyada doğru strateji geliştirmek için siber istihbarat akışının karar vericilere sağlanması gerekmektedir. Siber istihbarat, tehdit ve fırsatların farkedilmesini konu alan bir disiplin olarak karşımıza çıkmaktadır [4]. Siber tehdit istihbaratı (STİ / cyber threat intelligence, CTI) ise siber uzaydaki tehditlere yönelik özelleşmiş bir disiplindir. Siber uzaydaki varlıklar, göründüğünden çok daha derinlere uzanmaktadır. İnterneti algılayış şeklimizin ötesine geçtiğimizde bir meta gerçeklik ile karşılaşılmaktadır. Bu durum siber mikrobiyom olarak adlandırılmaktadır [5]. Siber mikrobiyomun görünür yüzünde yani internetin üst yüzeyinde Google, Facebook, Youtube, Wikipedia, bloglar, web siteleri, e-

ticaret siteleri ve kurumsal iş uygulamaları bulunurken; derin internette ise, Gizli Wiki (Hidden Wiki), Tor, İpek Yolu (Silk Road), gizli ağlar, gizli forum siteleri, derin internet bankacılığı, yasadışı bilgiler, veri sızıntılarından oluşan veri tabanları, çalıntı erişim bilgileri ve çok daha fazlası bulunmaktadır.

Siber mikrobiyomun hem gönünür hem karanlık tarafında faaliyet gösteren önemli aktörlerden birisi gelişmiş kalıcı tehdit (advanced persistent threat, APT) gruplarıdır. Gelişmiş kalıcı tehdit, terör örgütleri veya devletler tarafından fonlanan ve organize olmuş tehdit aktörleri tarafından düzenlenen karmaşık, çok yönlü, uzun vadeli siber tehditleri ifade etmektedir. APT grupları, devletlerin ve orduların siber savaş birimleri veya devletler tarafından desteklenen gelişmiş siber suç grupları olarak karşımıza çıkabilmektedir. Gruplar, izinsiz erişim elde etmek ve uzun bir süre (aylar, hatta yıllar) fark edilmeden hedef ağlarda kalabilmek için sıfır-gün (zero-day) saldırıları, sosyal mühendislik ve hedefe özgü özelleşmiş zararlı yazılımlar gibi gelişmiş tekniklerin bir kombinasyonunu kullanmaktadır [6]. APT grupları, casusluk, veri hırsızlığı, fidye veya hizmet kesinti gibi motivasyonlar ile şirketler, endüstriler, kritik altyapılar ve devletlere odaklı çalışmaktadır [7]. APT saldırıları, saldırganlar amacına ulaşına kadar uzun süren bir çalışma gerektirmektedir [8]. Kamu tarafında devletlerin istihbarat servisleri, özel sektörde Google Threat Analysis Group (TAG), Cisco Talos, Trellix Threat Intelligence Group (TIG) ve Recorded Future Insikt Group gibi küresel siber tehdit araştırma grupları APT grupları ve saldırılarını mercek altına alan ve haklarında derin araştırmalar yapan organizasyonlardan başlıcalarıdır.

Tezin önemi

Siber tehditler ve APT grupları hakkında bilgi sahibi olmak için literatürde ve siber güvenlik endüstrisinde birçok çalışma, ürün ve yöntem bulunmaktadır. İstihbarat ve siber istihbarat alanlarında olduğu gibi STİ'deki en önemli unsurlardan birisi zaman bağlamıdır. Sahip olunan istihbaratın geçerli kabul edildiği bir an veya zaman aralığı bulunmaktadır. İstihbaratın oluşmasına sebep olan olayların farkedilmesi ve istihbarat üretim sürecinin işletilerek, oluşan bilgiyi ilgililerine ulaştırma aksiyonunu hızlı alabilen kurum ve şirketler siber tehditler ile olan mücadelede avantaj elde etmektedir. STİ alanında başarılı olan örneklere bakıldığında, son kullanıcı cihazlarından sensörler aracılığıyla veri toplayabilen kurum ve şirketler ile elde ettiği verileri yorumlayarak yüksek katma değerli STİ üretebilen

kurum ve şirketler görülmektedir. CrowdStrike, Kaspersky ve Bitdefender son kullanıcı cihazlarından sensör verisi toplayarak STİ üretmek konusunda dünyadaki öncü şirketlerdendir.

Zaman bağlamıyla birlikte bir diğer önemli konu ise istihbarat bütünlüğü sağlayabilmektir. Sahip olunan tüm STİ'yi bir araya getirerek zaman boyutu ve siber uzaydaki mekânsal boyut dikkate alınarak olabildiğince kesintisiz bir STİ ağı elde edilmesi önem arz etmektedir. Siber uzayda mekân olarak kabul edilebilecek çeşitli ardesler bulunmaktadır. Bu adresler temelde kapalı ağlara veya İnternete bağlı, fiziksel ve sanal makineleri gösteren IP adresleridir ve bu adreslere çözülen lokal veya global alan adlarıdır. Ayrıca bu IP adreslerinin fiziksel dünyada hangi ülke ve otonom sisteme (autonomous system, AS) ait olduğu, STİ'nin üretilmesinde bağlamsal öneme sahiptir.

Siber güvenlik sektöründe incelenen STİ kaynaklarına bakıldığında adres ve zaman bağlamından koparılmış, STİ bildirimleri arasındaki ilişkilerin kurulmadığı, durumsal farkındalığın oluşmasına imkân tanımayacak kadar fazla yakına odaklanmış STİ bildirimleri ile karşılaşılmaktadır. Aynı şekilde APT saldırılarına dair raporlar incelendiğinde çoğunlukla ilgili saldırının sınırları içerisinde kalınarak tespitler ortaya çıkarıldığı görülmektedir. Öte yandan bu doktora tezinin yazılmasındaki temel motivasyon, güvenilir kaynaklardan bir araya getirilmiş STİ içeren HBA oluşturulması, çok sayıdaki APT saldırı raporlarından elde edilen STİ ile zenginleştirilmesi ve ağdaki varlıklar arasındaki ilişkilerin keşfedilmesi ile elde edilecek yeni STİ çıkarımlarının siber saldırıların çözümlenmesinde ve gelecek saldırıların henüz tehdit aşamasındayken farkedilerek engellenmesine imkân sağlanacak bir çerçeve modelin STİ disiplinine kazandırıma fikrine dayanmaktadır.

Problemin durumu

İnternet'in ulaştığı boyut ve karmaşıklık düşünüldüğünde, devletlerin ve kurumların kendilerine yönlenmiş APT gruplarından ve ilişkili tehditlerden tamamen haberdar olması oldukça zordur. Bu nedenle, harici STİ paylaşım ortakları bulmak, onlara güvenmek, iş birliği yapmak, edinilen STİ'leri çapraz kontrol etmek ve doğrulamak önemlidir [9]. STİ'nin etkili, aksiyon alınabilir ve geçerli olabilmesi için zamanla ilişkilendirilmesi gerekir. STİ, harici bir kaynaktan temin edilebildiğinde, ilgili istihbarat sağlayıcısının da

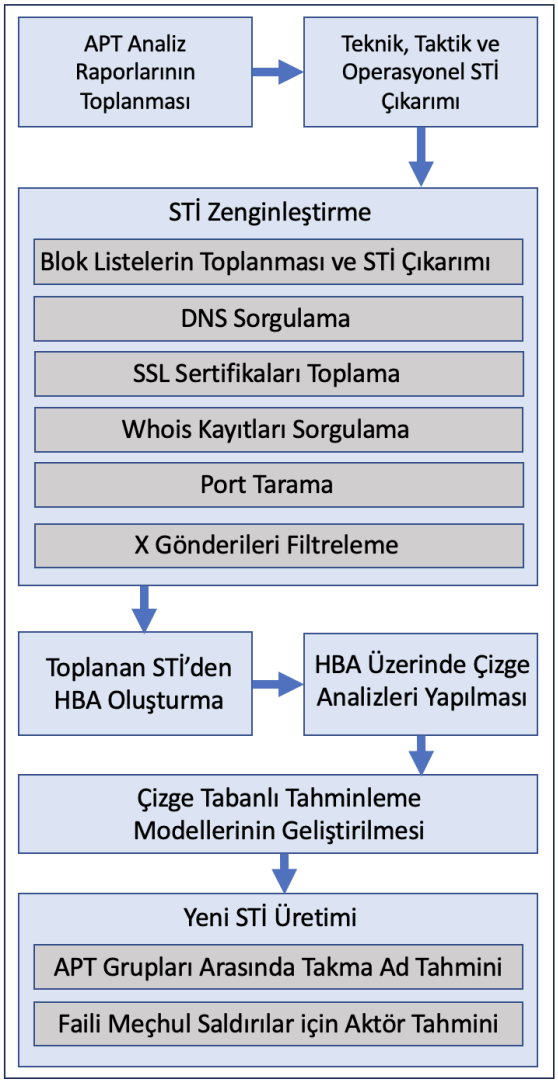
güvenilir ve yetkin olması gerekmektedir. Özellikle teknik STİ, hızla yaşlanan ve geçerliliğini hızla yitiren özelleşmiş ve olgunlaşmış bilgilerdir, bu nedenle yüksek kaliteli STİ oluşturma sürecinde doğrulama ve geçerleme önemlidir. STİ'nin hızla yaşlanan doğası nedeniyle, tehditlerle ilişkilendirilen kimlik, göstergeler ve hedefe ilişkin saldırı deseni, davranış ve bağlam da sürekli olarak bozulma eğilimindedir [10]. APT gruplarının dahil olduğu değerlendirilen geçmiş saldırılardan çıkarılan geçmiş ve güncel verilerin devamlı toplandığı ve internetten sürekli elde edilen güncel açık kaynak istihbarat ile ilişkilendirilerek geniş bir heterojen bilgi ağının (HBA / heterogeneous information networks, HIN) oluşturulduğu kapsamlı bir akademik çalışma bulunmamaktadır. APT odaklı olarak yüksek düzeyde bağlam kazandırılan STİ oluşturulmaması sebebiyle, APT'ler ve APT grupları arasında kanıta dayalı ilişkiler kurulmamış durumdadır. STİ endüstrisinde, APT saldırılarının incelenmesi sayesinde keşfedilen taktik, teknik ve prosedürlerin (TTP / tactics, techniques and procedures, TTP) benzerliği üzerinden APT gruplarına takma isimler verilmektedir. Güvenlik şirketleri aynı gruplara farklı isimler vererek bu grupların faaliyetlerini takip altına almaktadır. Takma isimler arasında, saldırı gruplarının kullandığı benzer TTP'ler arasında örüntüler gözlemlenerek APT gruplarına ait takma isimler arasında ilişkiler kurulmaya çalışılmaktadır.

APT grupları arasındaki ilişkileri ortaya çıkarmak, bu ilişkilere kanıta dayalı veriler üzerinden bağlam kazandırmak ve devamında analizler yaparak geleceğe yönelik teknik, taktik, operasyonel ve stratejik seviyede tehditlere yönelik önleyici STİ üretebilmek için sistematik ve analitik bir yöntem belirlemek ihtiyacı, bu problemin etraflıca tanımlanmasını ve bu tez çalışmasının yapılmasını sağlamıştır.

Tezin amacı

STİ'nin amacı, mevcut ve gelecekte ortaya çıkabilecek tehditlere ilişkin teknik göstergeler, bağlam ve harekete geçirici bilgi sağlamaktır. HBA ise gerçek dünya bilgilerini entegre etmek ve modellemek için bir grafik modeldir. Tezin amacı ise açık kaynak STİ içeren HBA'ların üretimi ve APT saldırılarından elde edilen STİ ile zenginleştirilmesi sayesinde APT grupları arasında bilinmeyen ilişkileri keşfetmektir. Bu sayede, HBA'lar aracılığıyla, büyük hacimdeki STİ verisi çizge veri yapısına dönüştürülerek modellenmekte ve gelecek analizler için temel olarak kullanılmaktadır. Temel alınan HBA farklı kaynaklardan gelen bilgiler ile zenginleştirilmektedir. Böylece siber tehditler ile ilgili bütüncül, sürekli ve bilgi

kopuklukları giderilmiş bir HBA elde edilerek, siber tehditler ve saldırılar arasındaki bilinmeyen ilişkilerin keşfedilmesi amaçlanmıştır. Tezin amacına ulaşma sürecinde uygulanan işlem basamakları Şekil 1.1'de sunulmaktadır.



Şekil 1.1. Tezde uygulanan işlem basamakları

Tez kapsamında hedeflenen amaca ulaşmak için çeşitli araştırma soruları belirlenmiş ve aşağıda sunulmuştur:

1- APT analiz raporlarından STİ çıkarımı yapılarak HBA oluşturulabilir mi?
2- STİ zenginleştirme işlemleri uygulanarak HBA'nın bağlamsallığı yüksek seviyeye ulaştırılabilir mi?
3- Çizge tabanlı tahminleme modelleri geliştirilerek APT grupları arasında takma ad ilişkileri ve faili meçhul saldırılar ile APT grupları arasında ilişkiler kurulabilir mi?

Tezin kapsamı

Tez çalışması kapsamında hedeflenen amaca yönelik oluşturulan araştırma sorularını cevaplandırmak için bir araştırma stratejisi oluşturulmuş, literatür taraması yapılmış, deneysel bir çalışma yapılmıştır. Siber güvenlik endüstrisinde ve STİ özelinde faaliyet yürüten organizasyonlar ile şirketlerin yürüttüğü çalışmalar, geliştirdikleri yöntem ve ürünler incelenmiştir. Kapsam dahiline alınan araştırma sorularını cevaplamak için aşağıda ifade edilen çalışmalar gerçekleştirilmiştir.

Birinci araştırma sorusu kapsamında, APT analiz raporları PDF dosya formatında indirilerek başlanmıştır. PDF dosyalarında yer alan GİG'lerin tespit edilmesi için RegEx eşleştirme iştemi yapılmıştır. RegEx ile ayırt edilemeyen özel isme sahip zararlı yazılımlar ve APT grupları doğal dil işleme (natıral language processing, NLP) yöntemi ile metin içerisinden ayrıştırılmıştır. Bu iki özel isim türündeki ifadeler elde etmek için bir varlık ismi tanıma (named entity recognition, NER) modeli eğitilmiştir. RegEx ve NER ile elde edilen tüm GİG'ler APT analiz raporları ile ilişkilendirilerek başarılı şekilde bir çizge veri yapısına dönüştürülmüştür. Oluşturulan çizge heterojen türdeki düğümlerden meydana geldiği için nihai olarak bir HBA elde edilmiştir.

İkinci araştırma sorusu kapsamında, APT analiz raporlarından elde edilen HBA'nın bağlamsallığını artırmaya yönelik veri kaynakları araştırılmıştır. Birbirleri ile ilişkilendirilebilir, düğüm ve ilişki çeşitliliğini artırmaya elverişli veri kaynakları incelenmiş ve toplanmıştır. Zenginleştirme işlemleri arasında pasif veri toplama yöntemi olarak blok listelerin ayrıştırılması ve HBA'ya eklenmesi yer almıştır. Diğer bir zenginleştirme yöntemi olarak ise aktif veri toplama işlemleri gerçekleştirilmiştir. Pasif ve aktif veri toplama yöntemleri uygulanarak HBA'nın bağlamsallığı artırılmıştır.

Üçüncü araştırma sorusu kapsamında, HBA üzerinde yeni ilişkiler tahminlemek için çizge tabanlı bir makine öğrenmesi hattı oluşturulmuştur. Bu hattın oluşturulması için uygulanan adımlar ve elde edilen çıktılar 5.2. Tahminleme Modelinin Eğitilmesi ve 6. Önerilen Çerçeve Modelin ve Tahminleme Modelinin Değerlendirilmesi başlıklarında sunulmuştur. Zenginleştirme süreci sonunda elde edilen nihai HBA üzerinde, APT grupları arasında takma ad ilişkileri tahminlenmiştir. Ayrıca faili meçhul olarak değerlendirilen ve HBA

üzerinde herhangi bir APT grubu ile ilişkilendirilemeyen saldırılar için tehdit aktörleri tahmin edilmiştir.

Tez kapsamında, araştırma sonuçlarının yorumlanması noktasında, kullanılan veri kaynakları, çalışmaya olan katkıları ve elde edilen sonuçlar değerlendirilmiştir. Veri kaynaklarının çeşitliliği biraz daha genişletilirse, geleceğe yönelik olarak tehdit aktörleri ile hedefleri arasında tahminleme yapılabileceği değerlendirilmiştir. Fakat geleceğe yönelik saldırı tahminlemesi tez kapsamı dışında bırakılmıştır.

Önerdiğimiz yöntem ve teknikleri diğer çalışmalardan farklılaştıran en önemli yönlerinden biri, STİ'nin bağlamını oluşturmak ve yaymak üzerine odaklanılmasıdır. STİ zenginleştirme işlemlerinin ve aktif veri toplamanın etkilerini açığa çıkarmak için veri kümesi üretirken iki ana senaryonun sonuçları karşılaştırılarak deneysel bir yöntem uygulanmıştır.

Senaryo 1: Açık kaynaklardan toplanan APT raporları, HBA'yı üretmek için tek veri kaynağı olarak kullanılmıştır.

Senaryo 2: APT raporları HBA'yı üretmek için temel veri kaynağı olarak kullanılmıştır. Buraya kadarki işlemler Senaryo 1 ile aynıdır. Ardından, Senaryo 2 kapsamına alınan veriler toplanmış, ayrıştırılmış ve HBA'ya eklenmiştir. Bu veriler; GİG, güvenilir blok listeleri (IPv4, IPv6, URL, alan adı, alt alan adı, SHA256, MD5, vb.), DNS ve Whois sorgu sonuçları, port tarama sonuçları, bilinen kötü amaçlı tam nitelikli alan adlarına (fully qualified domain name, FQDN) ait güvenli soket katmanı (secure sockets layer, SSL) sertifikaları ve parmak izleri, kötü amaçlı FQDN'ler içeren tweet'ler, zararlı dosya özetleri, X'ten (Twitter) filtrelenen ve toplanan IoC'lerdir.

Kısıtlar

Tez çalışması kapsamında toplanan STİ ve STİ üretimine yönelik zenginleştirici veri ve APT raporları açık kaynak istihbarat (open source intelligence, OSINT) yöntemleri uygulanarak açık kaynaklardan elde edilmiştir. Ücretli kaynaklar ve telif hakkı gerektiren veri kaynakları çalışma kapsamında kullanılmamıştır.

Literatüre sağlanan katkılar

Tez çalışması sonucunda literatüre aşağıdaki katkılar sunulmuştur:

1. Açık kaynaklardan genel amaçlı STİ'nin ve APT odaklı STİ'nin toplanması ve HBA'nın bağlam kazandırılarak büyütülmesi için bir sistem mimarisi önerilmiştir.
2. Yüksek kaliteli, güncel ve geçerli STİ elde etmek için zaman ile ilişkilendirilmiş ve sürekli bir yaklaşımla HBA oluşturulmuştur. Elde edilmiş nihai ağ akademideki diğer çalışmalar için bir altlık niteliği taşıyan kapsamlı bir veri tabanını meydana getirmiştir.
3. X (Twitter) gönderileri, port tarama sonuçları, DNS ve Whois sorguları, zararlı yazılım dosya özetleri ve kapsamlı blok listeleri gibi hem teknik hem operasyonel nitelikte STİ oluşturulmuş ve bağlam kazandırılarak ağa dahil edilmiştir.
4. Milyonlarca düğümden oluşan HBA analiz edilerek meta yollar ve meta-çizge keşfedilmiştir.
5. Oluşturulan HBA üzerinde çizge algoritmaları ile merkezilik ve topluluk analizleri yapılarak ağ üzerindeki etkisi yüksek düğümler keşfedilmiştir.
6. APT grupları arasındaki ilişkileri ve takma adları tahmin etmek için bir makine öğrenmesi hattı oluşturulmuştur.
7. Faili meçhul APT saldırılarına yönelik tehdit aktörlerini keşfetmek için çizge tabanlı bir tahminleme modeli geliştirilmiştir.

Bilgimize göre, bu tez çalışması birçok gerçek dünya veri kaynağını kullanarak APT raporları, APT saldırıları ve APT gruplarını barındıran en kapsamlı HBA oluşturan ve güvenilir kaynaklardan faydalanarak oluşturulan STİ'yi zenginleştiren, böylesine kapsamlı ve özgün bir veri kümesi oluşturan ilk çalışmadır.

Tez içerik planı

Tez çalışması kapsamında yapılan çalışmalara ilerleyen başlıklarda yer verilmektedir. Her bir bölüm ve içeriğini açıklayan tez içerik planı aşağıda sunulmaktadır.

İkinci bölümde, STİ ve HBA'lara yönelik alan bilgisi verilmektedir ve bu kapsamda kavram ve terimler açıklanmaktadır. STİ yaşam döngüsü incelenmektedir. Süreç perspektifinden STİ'nin oluşum süreci ele alınmaktadır. STİ elde edilebilecek kaynaklar araştırılmakta ve sunulmaktadır. STİ'nin paylaşılması için geliştirilen standartlar

incelenmekte ve kullanım senaryoları tartışılmaktadır. Son olarak ise HBA alanındaki temel bilgiler sunulmaktadır ve HBA ile STİ çalışma alanları arasındaki etkileşime yer verilmektedir.

Üçüncü bölümde, literatür taraması çalışmalarından elde edilen bilgiler sunulmaktadır. Literatür taraması üç ana başlık altında tasnif edilerek sunulmaktadır. İlk grupta STİ alanındaki literatür taraması bulunmaktadır. İkinci grupta HBA üzerine yapılan literatür taraması yer almaktadır.

Dördüncü bölümde, STİ içeren HBA'nın oluşturulması ve bu ağın zenginleştirilmesine yönelik model önerisine yer verilmektedir. Modeli oluşturan alt bileşenler açıklanmaktadır.

Beşinci bölümde, bir tahminleme modeli geliştirilmiştir. Tahminleme modelinin üretilmesine kadarki işlem basamakları açıklanmıştır.

Altıncı bölümde, HBA üretme ve zenginleştirme modeli ve tahminleme modelinin uygulama adımları test ortamında gerçekleştirilerek çıktılar değerlendirilmektedir. Modelin başarısına yönelik elde edilen ölçütler tartışılmakta ve yorumlanmaktadır.

Yedinci ve son bölümde ise tez çalışma kapsamında belirlenen araştırma soruları ve tezin amacı doğrultusunda yapılan çalışmalar özetlenmektedir. Tez kapsamında önerilen modelin iyileştirilmesine yönelik gelecek çalışmalar önerilmektedir. STİ'nin zenginleştirilmesi için edinilen bilgi ve tecrübeler paylaşılmaktadır.

2. SİBER TEHDİT İSTİHBARATI VE HETEROJEN BİLGİ AĞLARI

STİ, bilgiye dayalı tehdit yönetimini ve bilgiye dayalı karar verme mekanizmasını işler kılmayı hedefler. Etkili bir tehdit istihbaratı oluşturma ve yorumlama sürecinin ön şartı durumsal farkındalıktır. Durumsal farkındalık, tehdit istihabaratı oluşturan aktörün tehdit ve fırsatların farkında olması ve durumunu doğrudan veya dolaylı etkileyebilecek varlık ve eylemlerin farkında olması anlamına gelmektedir. Durumsal farkındalığın oluşması için STİ üretebilecek veri kaynaklarından haberdar olunması, bu kaynaklara nasıl erişilebileceği hakkında yöntem geliştirilebilmesi, tehdit aktörleri ve bu aktörlerin yöntemleri hakkında bilgi sahibi olunması gerekmektedir. Durumsal farkındalık, siber güvenlik ekosistemi içinde ele alındığında siber durumsal farkındalık diyebileceğimiz bir kavrama evrilmektedir. Fiziksel dünyanın aksine, siber uzayda durumsal farkındalık kazanmak için duyular yetersiz kalmaktadır. Teknik STİ üretebilmek için yüksek bilgi işleme kapasitesi, mühendislik yaklaşımları, yazılımsal çeviklik gerekmektedir. Stratejik STİ üretebilmek için uluslararası ilişkiler, devletler ve şirketler arası ilişkilerin de hesaba katılması gerekmektedir.

STİ'nin etkili, aksiyona geçilebilir ve geçerli olması için zaman ile ilişkilendirilmesi gerekmektedir. Tehdit isthibaratı bilgisi hazır olarak harici bir kaynaktan edinildiğinde, ilgili istihbarat sağlayıcısının da güvenilir ve yetkin olması gerekmektedir. Tehdit istihbaratı yaşlanan ve geçerliliğini çabuk kaybeden bir bilgidir. Bu sebeple kaliteli tehdit istihbaratı oluşturma sürecinin içinde doğrulama ve geçerleme yapılmasına ihtiyaç duyulmaktadır. STİ'nin çabuk yaşlanan ve geçerliliğini yitiren doğası sebebiyle, tehdite ilişkin saldırı örüntüsü, davranış biçimi ve kimlik, göstergeler ve hedef ile ilişkilendirilen bağlam da bozulma eğilimindedir.

STİ'nin güçlü bir bağlama sahip olması gerekmektedir. Bağlamı olmayan bir tehdit istihbaratı eksik kabul edilebilir. Örneğin; Windows zafiyetli bir işletim sistemidir, önermesi bir STİ ifade etmemektedir. Tehdite ilişkin kimlik, yöntem, araç, zararlı ve hedef gibi ne kadar çok bilgiye sahip olunursa, görülebilecek zararın etkisi azaltılabilir veya tamamen kaçınılabilir. STİ üretiminde girdi olarak toplanan veri ve nihai tehdit istihbaratı verisine bağlam kazandırmak için, elde edilen veri, enformasyon, bilgi ve istihbarat arasında ilişkiler kurulması, varolan fakat henüz farkında olunmayan ilişkilerin keşfedilmesi ve istihbarat zenginleştirme faaliyetlerinin yürütülmesi gerekmektedir. Böylece istihbaratın sahip olduğu bağlam güçlenmiş, çelişkiler varsa ortaya çıkmış, istihbarat güvenilirliğinin sınanabilmesi için bir imkân elde edilmiş olmaktadır.
STİ'yi ifade eden bilgi varlıkları ve bu bilgi varlıkları arasındaki ilişkiler tanımlandığında ortaya

bir bilgi ağı çıkmaktadır. Bu bilgi varlıkları ve arasındaki ilişkilerin farklı türlerde olması bir HBA meydana getirmektedir. HBA formunda ifade edilen bilgi yapıları üzerinde gelişmiş analiz, zenginleştirme ve tahminleme yöntemleri uygulanabilmektedir.

2.1. Alan Bilgisi ve Ontoloji Oluşturulması

Bilgisayar bilimlerinde, ontolojiler anlamsal / semantik çalışmaların bir parçasıdır. Ontoloji, belirli bir alanın varlıklarını, varlıkların sınıflarını, özelliklerini ve ilişkilerini modellemekte ve tanımlamaktadır. Bilgiyi paylaşmak, yeniden kullanmak ve farklı kaynaklardan gelen bilgileri entegre etmek için kullanılabilmektedir. Ontolojiler terimlerin anlamlarıyla ilgilendiğinden terimleri açık bir şekilde tanımlanması gerekmektedir. Ontoloji, varlıklar arasındaki anlamsal ilişkileri de tanımlamaktadır. Basit bir ontoloji, kavramlardan, ilişkilerden ve özelliklerden oluşurken, kapsamlı bir ontoloji, ilişkiler üzerindeki aksiyomları ve kısıtları da içermektedir [11]. STİ alanındaki ontolojiyi oluşturan kavramlar ve bu kavramlar arasındaki ilişkiler Çizelge 2.1'de sunulmaktadır.

Çizelge 2.1. STIX alan nesneleri listesi ve tanımları

Özne	İlişki	Nesne
Saldırı örüntüsü	Hedef almak	Kimlik, Zaafiyet
	Kullanmak	Zararlı yazılım, Araç
Harekât	Atfedilmek	İhlal kümesi, Tehdit aktörü
	Hedeflemek	Kimlik, Zaafiyet
	Kullanmak	Saldırı örüntüsü, Zararlı yazılım, Araç
Hareket tarzı	Azaltmak	Saldırı örüntüsü, Zararlı yazılım, Araç, Zafiyet
Zafiyet	Göstermek	Saldırı örüntüsü, Harekât, İhlal kümesi, Zararlı yazılım, Tehdit aktörü, Araç
İhlal kümesi	Atfedilmek	Tehdit aktörü
	Hedef almak	Kimlik, Zafiyet
	Kullanmak	Saldırı örüntüsü, Zararlı yazılım, Araç
Zararlı yazılım	Hedef almak	Kimlik, Zafiyet
	Kullanmak	Araç
	Varyantı olmak	Zararlı yazılım
Tehdit aktörü	Atfedilmek	Kimlik
	Kimliğine bürünmek	Kimlik
	Hedef almak	Kimlik, Zafiyet
	Kullanmak	Saldırı örüntüsü, Zararlı yazılım, Araç
Araç	Hedeflemek	Kimlik, Zafiyet

STİ alanında ontoloji çalışmalarının yapılması oldukça zor bir çalışmadır. Bunun başlıca sebeplerinden birisi, bu alandaki kullanılan terminolojinin net bir tanımlamasının uzun yıllar yapılmamış olması sebebiyle bu konuda çalışan araştırmacıların ve sektördeki uzmanların

kavramlara zaman içerisinde farklı anlamlar yüklemeye başlamasıdır. Bu alanda ürün ve çözüm geliştiren şirketlerin yazdığı birçok rapor ve makale bulunmaktadır. Endüstrideki literatürün oluşumu esnasında, çalışmaları kaleme alan uzmanların kavramların içini doldururken kullandıları bağlamlar farklılık göstermektedir. Bu sebeple terimlerin ve kavramların sınırları ve kapsamları net bir şekilde çizilememiştir.

Diğer bir sebebi ise, resmi olarak tanımlanmş standartların olmaması ile toplulukların, şirketlerin ve düşünce kuruluşlarının ortaya koyduğu standartların zamanla çeşitlenmiş olmasıdır. STİ alanındaki taksonominin zamanla stadartlaşması oluşan kafa karışıklarının ortadan kalmasına fayda sağlayacağı değerlendirilmektedir. Diğer bir sebep ise var olan ontolojilerin, oluşturduğı soyutlamaların farklı detay seviyelerinde tutarlı olmasıdır.

Çizelge 2.2. STİ alanındaki ontolojilerin kapsamları

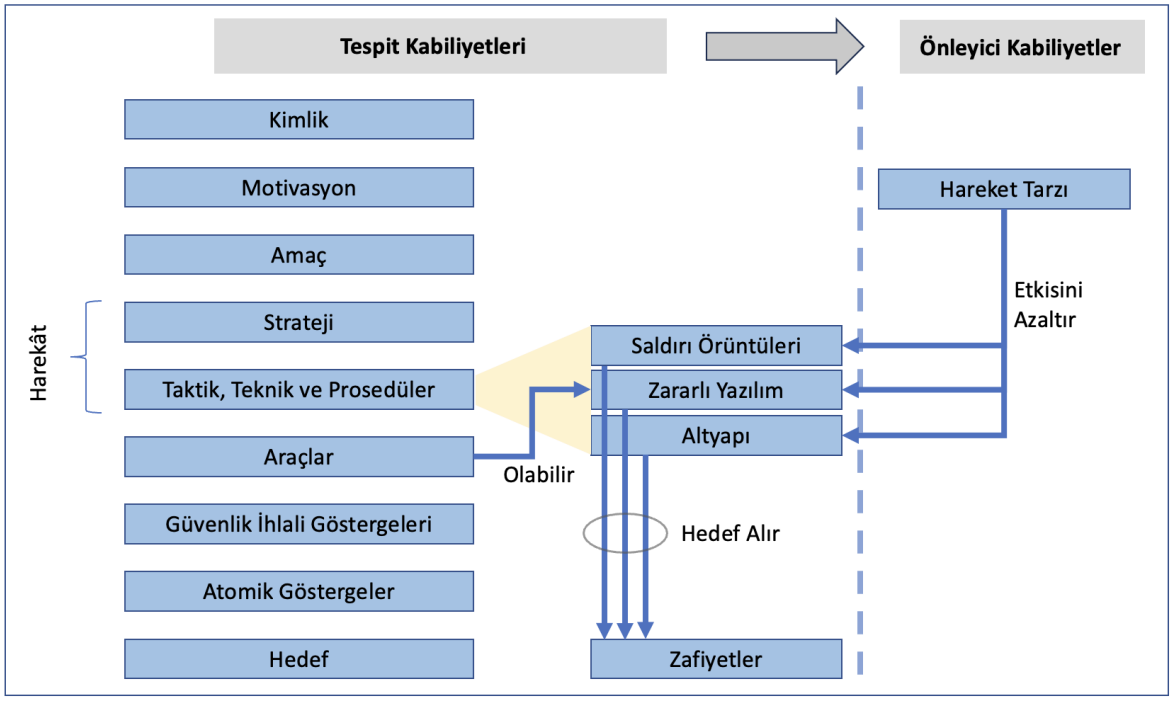
Çalışma	Kimlik	Motivasyon	Amaç	Strateji	TTP	Araç	GİG	Atomik GİG	Hedef	Hareket Tarzı
Formalizing Information Security Knowledge [12]								✓		
OVM: An Ontology for Vulnerability Management [13]					✓			✓		✓
Developing an Ontology of the Cyber Security Domain [14]	✓					✓		✓	✓	
A Knowledge-Based Approach to Intrusion Detection Modeling [15]					✓			✓		
Building an Ontology of Cyber Security [16]	✓				✓			✓	✓	

Çizelge 2.2. (devam) STİ alanındaki ontolojilerin kapsamları

Çalışma	Kimlik	Motivasyon	Amaç	Strateji	TTP	Araç	GİG	Atomik GİG	Hedef	Hareket Tarzı
Ontology for Malware Behavior: a Core Model Proposal [17]						✓		✓		
Enabling New Technologies for Cyber Security Defense with the ICAS Cyber Security Ontology [18]					✓			✓		
Developing an Ontology for Cyber Security Knowledge Graphs [19]	✓				✓	✓		✓		
An Ontology of Suspicious Software Behavior [20]						✓	✓	✓		
Using Ontologies to Quantify Attack Surfaces [21]					✓			✓	✓	
A Survey on Systems Security Metrics [22]					✓					✓
UCO: A Unified Cybersecurity Ontology [23]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

STİ alanındaki ontolojinin oluşturulmasına yönelik yapılan çalışmalar ve sahip oldukları kapsamlar Çizelge 2.2'de verilmiştir. Görüldüğü üzere bu alandaki ilk çalışmalar siber saldırılar hakkında tehdit istihbaratını tanımlama noktasında göstergeler üzerinden yola çıkmaktadır. Siber saldırıların oluşumunu uçtan uca ele alan ve tanımlayan Massachusetts teknoloji araştırma ve mühendislik enstitüsü (Massachusetts Institute of Technology Research and Engineering, MITRE) düşman taktikleri, teknikleri ve ortak bilgisi (Adversarial Tactics, Techniques and Common Knowledge, ATT&ACK) (2015) ve

Lockheed Martin Siber Ölüm Zinciri (Cyber Kill Chain) (2011) gibi taksonomi çalışmalarının olgunlaşması ile birlikte bu alandaki ontoloji çalışmaları da artık saldırganların motivasyonları, amaçları ve stratejilerini tanımlamaya çalışarak, stratejik ve operasyonel STİ oluşturabilecek seviyede ontolojiler oluşturmaya başlamıştır [23]. Ontoloji oluşturulmasına yönelik literatürdeki güncel çalışmalar Şekil 2.1'deki STİ modelinde yer alan bilgi varlıklarını kapsam içerisine almaktadır.



Şekil 2.1. Kabiliyet odaklı STİ modeli [24]

STİ'yi tanımlamak, tehditi tespit etmek ve saldırıyı önleyecek kabiliyetleri geliştirmek için mevcut tehdit istihbaratı paylaşım standartları ve bu alandaki bilgi varlıklarının taksonomisini gözeten bir modele ihtiyaç duyulmaktadır. Mavroeidis ve diğerlerinin geliştirdiği model bu ihtiyacı karşılamaya yöneliktir [24].

2.2. Kavramlar ve Terimler

Bu bölümde tez konusu ve araştırma sorularına sistematik olarak cevaplar aranırken karşılaşılan, STİ'den oluşan HBA oluşturulması ve zenginleştirilmesi sürecini incelerken bilinmesi gereken kavramlar sunulmaktadır. Bu kavramlar literatürün anlaşılması için temel oluşturmaktadır.

Veri: Veri kavramı farklı disipler tarafından farklı şekillerde tanımlanmaktadır. Temel anlamıyla, işlenmemiş ve bir gerçeğe ilişkin bilgi parçasına verilen isimdir [25]. Bilgisayar bilimleri açısından ise veri, hesaplama ya da manipülasyon amacı ile kullanılan bir gerçeği belirtmektedir [26]. Haberleşme teknolojileri perspektifinden bakıldığında ise, veri, anlam ve bağlamdan bağımsız, bütünlüğü ve yapısı bozulmamış ve bir noktadan diğerine iletilmesi istenen metinler, sesler, görseller ve videoları ifade etmektedir [27].

Enformasyo: Enformasyon kavramı genel anlamıyla belirli ve görece sınırlı bir bağlam ile ilgili, bir araya getirilmiş bilgi parçasıdır. Bilgisayar bilimleri özelinde bakıldığında ise bir bilişim sistemi tarafından kayıt, depolama, sorgulama, düzenleme ve özetleme işlemlerinden geçirilerek biçimlendirilmiş ve anlam kazandırılmış veri olarak tanımlanmaktadır. Haberleşme teknolojileri perspektifinden bakıldığında ise, bir veri bütününün anlamından bağımsız olarak, o bütüne dair sayısal bilgi olarak tanımlanmaktadır [28].

Bilgi: En genel anlamıyla, insanın kendisini ve çevresini anlamasını şekillendiren zihinsel süreç olarak tanımlanmaktadır [29][30]. Bilgi, belirsizliğin çözümü olarak da tanımlanmaktadır. Bir varlığın ne olduğuna açıklık getirmekte ve varlığın hem özünü hem de özelliklerinin doğasını tanımlamaktadır. Bilgi kavramının farklı bağlamlarda farklı anlamları bulunmaktadır ve bilgi bir bağlamdan bağımsız ele alınamamaktadır [31]. STİ disiplini bağlamında kapsayıcı bir tanımlama yapılacak olursa; bilgi, ikinci bir sistemin birinciyle ilişkisinin veya iki sistemin belirli bir çevreyle paylaştığı ilişkinin başka bir sistem tarafından yorumlanmasını etkileyen bir sistemden kaynaklanan uyarıcıdır [32]. Bu tanımda da görüldüğü gibi bilgi bağlam ve ilişkiden bağımsız ele alınamamaktadır.

İstihbarat: İstihbarat, kamuya açık olmayan bilgiler veya en azından kısmen bu tür bilgilere dayanan, karar vericiler için hazırlanmış analizlerdir. İstihbaratı benzersiz kılan, gizlice toplanan ve karar vericilerin ihtiyacını karşılamak için zamanında hazırlanan bilgileri kullanmasıdır [33]. Bu tanımda ifade edildiği haliyle istihbarat, bilginin girdi olarak kullanıldığı bilgi işleme ve yeni bilgi üretim sürecinin çıktısıdır. Burada istihbaratın önemli bir bileşeninin, onun ne zamana ait olduğu görülmektedir.

Tehdit: Güvenliği ihlal edebilecek ve zarar verebilecek bir durum, eylem veya olay olduğunda ortaya çıkan güvenlik ihlali potansiyeli olarak tanımlanmaktadır [34]. Diğer bir deyişle, tehdit, bir güvenlik zafiyetinden yararlanabilecek olası bir tehlikedir. Bir tehdit hedefe yönelik veya

tasarsız oluşabilmektedir. Bilgi güvenliği perspektifinden bakıldığında, Amerika Birleşik Devletleri Ulusal Standartlar ve Teknoloji Enstitüsü (National Institute of Standards and Technology, NIST) tarafından hazırlanan Federal Bilgi İşleme Standartları 200 (Federal Information Processing Standards 200, FIPS 200), Federal Bilgi ve Bilgi Sistemleri için Minimum Güvenlik Gereksinimleri'nde ise yetkisiz erişim, bilgilerin yok edilmesi, ifşa edilmesi, değiştirilmesi ve/veya hizmet kesintisi yoluyla bir bilgi sistemi aracılığıyla kurumsal operasyonları, kurumsal varlıkları veya bireyleri olumsuz etkileme potansiyeli olan herhangi bir durum veya olay olarak tanımlanmaktadır [35]. Yine NIST'in önerdiği bir tanıma göre, bir tehdit kaynağının belirli bir bilgi sisteminin güvenlik açığından başarıyla yararlanma potansiyeli olarak da tanımlanmaktadır.

Siber istihbarat ve siber tehdit istihbaratı: Yalnızca ağ operasyonlarını ve faaliyetlerini değil, aynı zamanda bunları kimin, neden yaptığını ve bir sonraki adımda ne olabileceğini de anlamaya yönelik bilginin ortaya çıkarılmasıdır [36]. Siber tehdit istihbaratı; bağlam, göstergeler, sonuçlar ve eyleme geçirilebilir tavsiyeler dahil olmak üzere, mevcut ve sonradan ortaya çıkan tehditler hakkında, öznenin bu tehditlere dair tepkisine ilişkin kararlarını yönlendirmek için kullanılabilecek kanıta dayalı bilgi olarak tanımlanmaktadır [37].

Güvenlik ihlali göstergeleri: GİG'ler, bir sistemin güvenliğinin bir saldırı sonucunda ele geçirildiğinin veya belirli bir kötü amaçlı yazılımdan etkilendiğinin işaretleri olarak tespit edilen delillerdir [38].

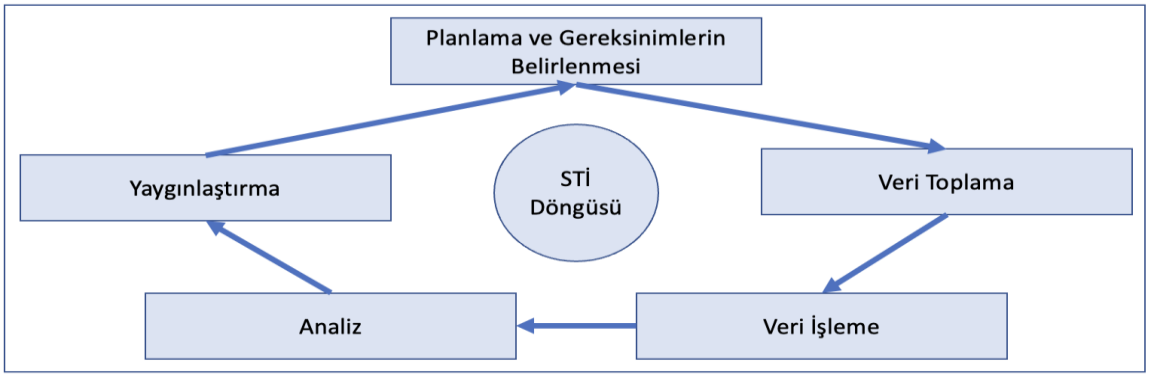
Bilgi ağı: Nesnelere ve bu nesneler arasındaki etkileşimlere odaklanarak gerçek dünyanın bir soyutlamasını temsil etmektedir [39].
Heterojen bilgi ağı: HBA, çok sayıda düğüm ve bağlantıdan oluşan, özelleşmiş bir bilgi ağı türüdür ve güçlü bir bilgi modelleme yöntemi olarak tanımlanmaktadır [40].

2.3. Siber Tehdit İstihbaratı Döngüsü

Geleneksel istihbarat döngüsü, istihbarat oluşum sürecini tarifleyen tekrar eden adımlardan oluşan kapalı bir yol olarak tanımlanmaktadır. İstihbarat döngüsünün aşamaları, karar vericiler tarafından planlama ve gereksinimlerin belirlenmesi, toplama, işleme, analiz, yaygınlaştırma ve geri bildirim fazlarından meydana gelmektedir [41].

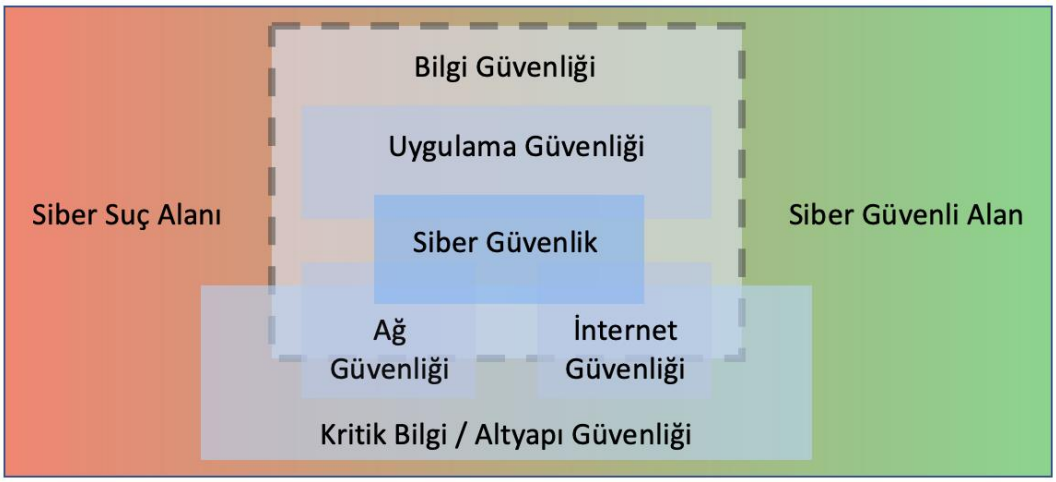
Temel olarak STİ'nin amacı; tehdit aktörlerini belirlemek, motivasyonlarını, yöntemlerini ve özelliklerini anlamaktır. Bu amaca ulaşmak için gerekli olan siber tehditlerle ilgili istihbarat genellikle iki kategorideki kaynaktan elde edilmektedir. Birinci kategoride ağ trafiği gibi teknik kaynaklar, kötü amaçlı yazılım imzaları içeren açık ve gizli veritabanları bulunmaktadır. Bu kategori altında gözden kaçırılmaması gereken diğer bir unsur ise tehdit aktörlerinin kabiliyetlerinden bağımsız olarak, sistemlerin sahip olduğu güvenlik zafiyetlerinden haberdar olmaktır. İkinci kategoride ise insan kaynaklı tehdit istihbaratı kaynakları bulunmaktadır. Bu kategori altında, saldırı gruplarına sızarak bilgi edinmek, sosyal medya mesajlarından bilgi toplamak gibi sosyal mühendislik faaliyetleri bulunmaktadır. Her iki kategorideki veri hacminin sürekli büyüdüğü bir gerçektir ve STİ üretmek için etkili analiz yöntemlerine ve iyi tasarlanmış modellere ihtiyaç duyulmaktadır.

STİ toplama, bağlam kazandırma, APT grupları ile ilişkilendirme üzerine yöntemler geliştirmeden önce yaşam döngüsünün incelenmesi gerekmektedir. Şekil 2.2'de STİ yaşam döngüsüne yer verilmiştir. Yaşam döngüsünün çıktılarının zenginleştirilmesi için her bir fazın neyi hedeflediği anlaşılmalıdır. Bu fazlarda yerine getirilen faaliyetlerin etkili kılınması ve nitelikli hale getirilmesi gerekmektedir.



Şekil 2.2. STİ yaşam döngüsü [41]

Bilgi güvenliği çalışmaları, kritik bilgi sistemi altyapılarının korunmasını amaçlamaktadır. Bilgi güvenliği mühendisliği, korunması gereken bilginin güvenliğinin sağlanmasına yönelik planların ve politikaların geliştirilmesi, güvenlik önlemlerin uygulanması, zafiyetlerin tespit edilmesi ve test edilmesi, güvenlik ihlallerinin izlenmesi ve incelenmesi konularında mühendislik çözümleri geliştirmektedir.



Şekil 2.13. Güvenlik alt alanları ve aralarındaki etkileşim [42]

Bu kapsamda Şekil 2.3'te görüldüğü üzere bilgi güvenliği faaliyetleri, siber suç alanı ile siber güvenli alanı birbirinden ayırmayı hedeflemektedir. Siber suç alanı ile siber güvenli alan arasındaki uzay, mantıksal ve fiziksel olarak birbirinden ayrılan pek çok ortam içermektedir [42]. Bilgi güvenliğinin altındaki özelleşmiş alt alanlara bakıldığında uygulama güvenliği, ağ güvenliği ve internet güvenliği yer almaktadır. Siber güvenlik ise bu üç alanla çeşitli yerlerde kesişen çalışma alanıdır.

STİ ise bilgi güvenliğinin sağlanabilmesi için uygulama, ağ ve internet ortamlarından gelen tehditleri belirlemekte, bu tehditleri sistematik bir şekilde incelemekte ve karşıt önlemlerin alınması için siber güvenli alanın bütünlüğünü sağlamaya hizmet etmektedir. Bu sebeple uygulama, ağ ve internet ile ilgili derin bir anlayışa sahip olunması, saldırganların motivasyonu ve yöntemleri ile ilgili zamana, yere, politik sebeplere, teknik ipuçlarına bağlı yorum ve analiz yapma kabiliyetlerine sahip olunması gerekmektedir. Bu sebeple literatürdeki ve siber güvenlik sektöründeki çalışmalar incelendiğinde, STİ'nin aksiyona geçilebilir hale gelmesi noktasında uzman yorumuna ve kararına bağımlı kalındığı pek çok kaynakta da ifade edilmektedir [43-48].

STİ'nin üretilmesi süreci baştan sona ele alındığında veri kaynaklarının belirlenmesinden aksiyona geçilebilir istihbaratın üretilmesine kadarki süreçte birçok yüksek maliyetli operasyon yürütülmesi gerekmektedir. Veriyi keşfetme, ulaşma, edinme, işleme, analiz etme, doğrulama, yorumlama ve nihai STİ'yi geçerleme işlemleri yüksek düzeyde kaynak (iş gücü, zaman, bütçe) gerektiren bir süreçtir. Kaynakların etkili kullanılabilmesi için STİ yaşam döngüsünün ilk fazı planlama ve gereksinimlerin belirlenmesi ile başlamaktadır.

Planlama ve gereksinimlerin belirlenmesi fazında, istihbarat sürecinin yönü ve gereksinimleri organizasyondaki karar verici/verciler tarafından belirlenmektedir. Organizasyon için gereksinimlerin belirlenmesi, istihbaratın organizasyonun performansının iyileştirilmesine katkıda bulunacağı politika ve güvenlik konularının belirlenmesi anlamına gelmektedir. Gereksinimler organizasyonun birçok sorunu içinden önceliklendirme yaparak meydana getirilmektedir. İstihbarat oluşturma kabiliyetleri ve kaynakların sınırlı olmasından dolayı önceliklerin doğru belirlenmesi gerekmektedir. Gereksinimlerin belirlenmesinde önceliklere organizasyondaki üst yönetimin karar vermesi gerekmektedir [49]. Pek çok kurum ve kuruluştaki siber istihbarat araştırmasının yönü, organizasyonun sahip olduğu ağı doğrudan hedef alabilecek tehditlere yönelik belirlenmektedir. Diğer taraftan ise ulusal düzeyde şirketlerin ve kurumların kritik altyapıları göz önüne alındığında STİ oluşturma yönelimi çeşitlenmektedir.

STİ oluşumunu sağlayacak veri teknik kaynaklardan ve insan merkezli kaynaklardan elde edilmektedir. Veri kaynaklarının türleri, toplanabilecek verinin boyutu ve yeni verinin üretilme hızı oldukça yüksek olduğu için veri toplama sürecinin otomatize edilmesi ihtiyacı bulunmaktadır. Açık kaynaklardan toplanan veri, istihbarat oluşumunu besleyecek veri havuzu olarak kabul edildiğinde yetersiz kalmaktadır. Toplanan verinin derin internet gibi kaynaklardan beslenerek zenginleştirilmesi gerekmektedir [50].

Veri toplama işlemlerinin otomasyona dönüşmesi gerektiği gibi, verinin işlenerek anlamlı bilgiye dönüşmesi süreci de otomatize edilmelidir. Verinin işlenmesinde otomasyon gücü ve alan uzmanının katkısı bir araya gelerek verinin nasıl işleneceğine dair bir bağlam sağlanmalıdır.

STİ yaşam döngüsünün planlama, toplama ve işleme adımlarında olduğu gibi analiz aşamasında da alan uzmanı görüşü değerlidir. STİ analizinin çıktıları, saldırganın motivasyonunu anlamaya ve gelecekteki faaliyetlerini tahmin etmeye faydası olabilir.

Yaygınlaştırma fazına gelindiğinde, elde edilen analiz çıktıları kullanılmaktadır ve yaygınlaştırılmaktadır. Analiz çıktıları döngüye tekrar girdi olarak dönmektedir. Tehdit istihbaratının bir döngüsünün etkinliği, veri toplarken hangi alanlara odaklanılacağı ve ne kadar hızlı önlem alınması gerektiği de dahil olmak üzere bir sonraki döngü için gereken bilgilerin temel unsurlarını belirleyecektir. STİ döngüsünün son aşaması oluşturulan istihbaratın etkinliğinin ölçülmesi, elde edilen geri beslemelerin gelecek döngülere iyileştirme ve yön verme anlamında fayda sağlaması için önemlidir.

2.4. Siber Tehdit İstihbaratı Paylaşımı

STİ, üretilmesi oldukça zor bir bilgidir. İstihbaratın üretilmesine yönelik faaliyetler saldırı öncesi, saldırı esnası ve saldırı sonrası olmak üzere temelde üç zamanda gerçekleşmektedir. Etkili bir STİ'nin saldırı gerçekleşmeden önce üretilmesi ve önleyici nitelikte olması kıymetlidir. Nitelikli ve yüksek maliyetlerle üretilen tehdit istihbaratı, değeri sebebiyle genellikle ücretsiz kaynaklarda sunulmamaktadır. Öte yandan STİ paylaşımı amacıyla kurulmuş ve karşılıklı fayda ilkesiyle faaliyet gösteren komuniteler de bulunmaktadır.

Zararlı yazılım bilgi paylaşım platformu (malware information sharing platform, MISP) yazılımı çok bilinen açık kaynaklı tehdit istihbaratı platformudur. MISP kullanıcıları oluşturdukları tehdit istihbaratını diğer kullanıcılar ile paylaşabilmektedir [11] [51]. MISP komunitelerinin bazıları kendi içlerine kapalı ağ şeklinde organize olmuştur, bazıları ise komınitenin geri kalanıyla tehdit istihbaratı paylaşacak şekilde organize olmuştur. Yaygın bilinen ve aktif çalışan MISP komuniteleri, bilgisayar olay müdahale merkezi Lüksemburg (computer incident response center Luxembourg, CIRCL), sivil toplum için bilgisayar olayı müdahale merkezi (computer incident response center for civil society, CiviCERT), Fidelis, First, kuzey atlantik antlaşması teşkilatı (north atlantic treaty organization, NATO) ve COVID-19'dur.

STİ'nin yapısal bir formatta depolanabilmesi ve paylaşılabilmesi için akademik dünyada, özel sektörde ve düşünce kuruluşlarının öncelik ettiği çalışmalar neticesinde çeşitli veri yapıları geliştirilmiştir. Bu veri yapıları, paylaşımı ve entegrasyonu kolaylaştırmak için JSON ve XML gibi bilindik veri formatlarında hazırlanmıştır.

2.4.1. Veri yapısı ve paylaşım standartları

STİ paylaşımı için birçok standart geliştirilmiştir. Zamanla olgunlaşan ve endüstride kabul görenleri yapılandırılmış tehdit bilgi ifadesi (structured threat information expression, STIX), güvenilir otomatik istihbarat bilgi değişimi (trusted automated exchange of intelligence information, TAXII), siber gözlemlenebilir ifade (cyber observable expression, CybOX), kötü amaçlı yazılım nitelik sayımı ve karakterizasyonu (malware attribute enumeration and characterization, MAEC), açık tehdit göstergeleri (open indicators of compromise, OpenIOC) ve olay kaydı ve olay paylaşımı için sözlük (vocabulary for event recording and incident sharing, VERIS)'dir. Bu standartların olgunlaşmasında açık kaynaklı yazılım projeleri ve ticari ürünlerin

bu standartları desteklemeleri etkili olmaktadır. STİ'nin paylaşımında, tehdit istihbaratı alanındaki bilgi varlıklarını kapsayacak şekilde oluşturulmuş güncel ontolojiyi adresleyecek veri yapılarına ihtiyaç duyulmaktadır. Çizelge 2.3'te paylaşım standartlarının ilgili bilgi varlıklarını adresleme kapasitelerine yer verilmektedir. CybOX standartı STIX ile birleştiği için çizelgede ayrıca yer verilmemiştir.

Çizelge 2.3. STİ paylaşım standartları ve kapsadıkları bilgi varlıkları

Çalışma	Kimlik	Motivasyon	Amaç	Strateji	TTP	Araç	GİG	Atomik GİG	Hedef	Hareket Tarzı
Standardizing cyber threat intelligence information with the structured threat information expression (STIX) [52]	x	x	x	x	x	x	x	x	x	x
Structured Threat Information Expression (STIX) 2.1 [53]	x	x	x	x	x	x	x	x	x	x
Malware Attribute Enumeration and Characterization V5.0 (MAEC) [54]							x			
OpenIOC V1.1 [55]					x	x	x	x		
VERIS	x				x	x	x	x	x	

STİ alanındaki güncel ontolojilere bakıldığında, bilinen bilgi varlıklarını adresleyebilen en olgun paylaşım standartı STIX 2.1 olarak değerlendirilmektedir.

2.4.2. STIX

STIX, STİ paylaşımı için geliştirilmiş bir dil sunar ve verinin sistemler arasında paylaşımı için bir serileştirme (serialization) formatıdır. Serileştirme, bir nesnenin durumunu kalıcı hale getirilebilecek veya taşınabilecek bir forma dönüştürme işlemidir. STIX formatı kullanılarak, bir tehdite ait özellikler ve bu özelliklerin birbirleri ile ilişkileri açık ve tanımlayıcı bir şekilde ifade edilebilmektedir. Bu ifadeler güvenlik analistlerinin yorumlaması için görselleştirilerek sunulabilir, JSON veri formatında saklanabilir ve

paylaşılabilmektedir. STIX'in entegrasyona açık yapısı analistlerin ve ağların ihtiyacına göre çeşitli siber güvenlik araç ve ürünleriyle entegrasyona imkân tanımaktadır. STIX, siber güvenlik endüstrisinde fiili standart (de facto standard) olarak kullanılmaktadır. STIX günümüzde çeşitli yapısal değişikliklere uğrayarak ve yeni kabiliyetler kazandırılarak V2.1 sürümüne ulaşmıştır. Tarihsel gelişimine bakıldığında, STİ paylaşımı için standart bir yapıya ihtiyaç olduğu, ilk kez 2010 yılında siber güvenlik uzmanlarının bilgi paylaşımı için kullandıkları e-posta listelerinde ifade edilmiştir. Takip eden yıllarda Amerika Birleşik Devletleri İç Güvenlik Bakanlığı (United States Department of Homeland Security, USDHS) desteğiyle MITRE tarafından STIX, TAXII ve CybOX geliştirilmeye başlanmıştır.

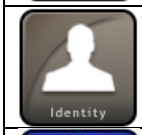
Yapılandırılmış bilgi standartlarının geliştirilmesi organizasyonu (organization for the advancement of structured information standards, OASIS) çatısı altında faaliyet ve çalışmalarını yürüten, siber tehdit istihbaratı teknik komitesi (cyber threat intelligence technical committee, CTITC), STIX ve TAXII standartlarının mevcut organizasyonları içinde geliştirilmeye devam edilmesini önermiştir.

2017 yılında V2.0 ve 2019 yılında ise mevcut V2.1 halini alarak standartlaşmıştır. İki ana kategoride tanımlamalar içermektedir. Bunlar STIX alan nesneleri (STIX domain objects, SDO) and STIX ilişki nesneleri (STIX relationship objects, SRO) şeklinde ifade edilmektedir. Alan nesneleri kapsamında 12 adet bilgi varlığı bulunmaktadır. STIX Alan Nesnelerini tanımlayan sembol, nesne adı ve tanımları içeren liste Çizelge 2.4'de sunulmaktadır.

Çizelge 2.4. STIX V2.1 alan nesneleri listesi ve tanımları

Sembol	Adı	Tanımı
	Saldırı Örüntüsü (Attack Pattern)	Saldırganların hedefleri tehlikeye atmaya çalıştığı yolları tanımlayan bir Taktik, Teknik ve Prosedür türüdür.
	Harekât (Campaign)	Belirli bir hedef grubuna karşı belirli bir süre boyunca meydana gelen bir dizi kötü niyetli faaliyeti veya saldırıyı tanımlayan bir grup düşmanca davranıştır.
	Davranış Biçimi (Course of Action)	Bir istihbarat üreticisinin tüketiciye, bu istihbarata karşılık olarak gerçekleştirebileceği eylemlere ilişkin tavsiyesidir.

Çizelge 2.4. (devam) STIX V2.1 alan nesneleri listesi ve tanımları

Sembol	Adı	Tanımı
	(Gruplama) Grouping	Başvurulan STIX Nesnelerinin, paylaşılan bir içeriğe sahip olduğunu beyan etmektedir.
	Kimlik (Identity)	Gerçek kişiler, kuruluşlar, kurumlar, sistemler ve sınıfları ifade etmektedir (örneğin, finans sektörü).
	Gösterge (Indicator)	Şüpheli veya kötü amaçlı siber etkinlikleri tespit etmek için kullanılabilecek bir örüntü içermektedir.
	Altyapı (Infrastructure)	Bir TTP türünü temsil eder ve herhangi bir amacı desteklemeye yönelik herhangi bir sistemi, yazılım hizmetini ve ilgili fiziksel veya sanal kaynakları tanımlamaktadır (örneğin, bir saldırının parçası olarak kullanılan komuta kontrol sunucuları, savunmanın parçası olan cihaz veya sunucu, bir saldırı tarafından hedeflenen veritabanı sunucuları).
	Saldırı Kümesi (Intrusion Set)	Bir organizaasyon tarafından yönetildiğine inanılan, ortak özelliklere sahip, gruplandırılmış bir dizi düşmanca davranış ve kaynağı ifade etmektedir.
	Lokasyon (Location)	Coğrafi bir konumu temsil etmektedir.
	Zararlı Yazılım (Malware)	Kötü amaçlı bir kodu temsil eden bir TTP türüdür.
	Zararlı Yazılım Analizi (Malware Analysis)	Kötü amaçlı yazılım örneğinde veya ailesinde gerçekleştirilen belirli bir statik veya dinamik analizin meta verileri ve sonuçlarını ifade etmektedir.
	Not (Note)	Daha fazla bağlam sağlamak ve Notun ilgili olduğu STIX Nesnelerinde yer almayan ek analizler sağlamak için bilgilendirici metin içermektedir.
	Gözlemlenen Veri (Observed Data)	Dosyalar, sistemler ve ağlar gibi siber güvenlikle ilgili varlıklar hakkında bilgi içermektedir.
	Görüş (Opinion)	Farklı bir tarafça üretilen bir STIX Nesnesindeki bilgilerin doğruluğunun değerlendirilmesini içermektedir.
	Rapor (Report)	Tehdit aktörünün tanımı, kötü amaçlı yazılım, bağlam, saldırı tekniği ve ilgili ayrıntılar gibi bir veya daha fazla konuya odaklanan tehdit istihbaratı bütünüdür.
	Tehdit Aktörü (Threat Actor)	Kötü niyetle faaliyet gösterdiğine inanılan gerçek kişiler, gruplar veya kuruluşları ifade etmektedir.

Çizelge 2.4. (devam) STIX V2.1 alan nesneleri listesi ve tanımları

Sembol	Adı	Tanımı
	Araç (Tool)	Tehdit aktörlerinin saldırı gerçekleştirmek için kullanabileceği yazılımlardır.
	Güvenlik Açığı (Vulnerability)	Bir saldırganın bir sisteme veya ağa erişim sağlamak için doğrudan kullanabileceği yazılımdaki bir hatayı ifade etmektedir.

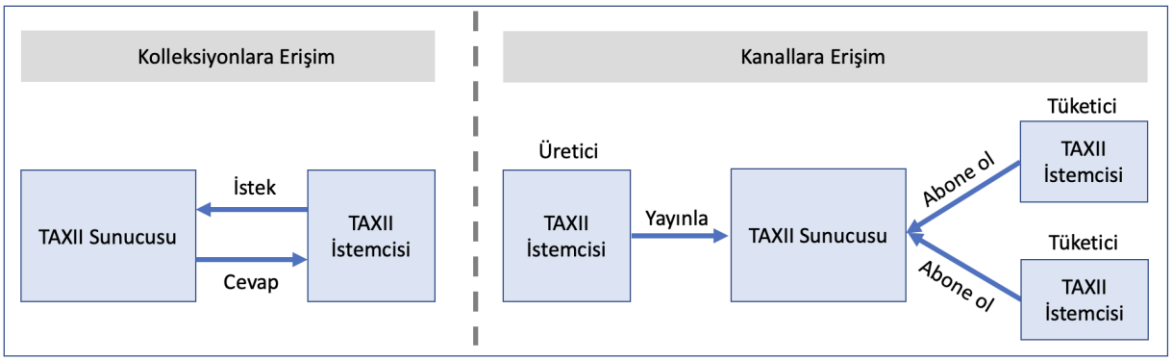
STIX yapısında tanımlanan ilişki nesneleri ve tanımları ise Çizelge 2.5'de yer almaktadır.

Çizelge 2.5. STIX V2.1 ilişki listesi ve tanımları

Nesne	Adı	Tanımı
	İlişki (Relationship)	Birbirleriyle nasıl ilişkili olduklarını açıklayarak iki STIX Alan Nesnesini birbirine bağlamak için kullanılmaktadır.
	Nişan (Sighting)	Siber Tehdit İstihbaratında bir şeyin (örneğin, bir gösterge, kötü amaçlı yazılım, araç, tehdit aktörü vb.) görüldüğü düşüncesini belirtmektedir.

2.4.3. TAXII

TAXII, güvenli köprü metni aktarım protokolü (hypertext transfer protocol secure, HTTPS) protokolü üzerinden STİ paylaşımı için bir uygulama katmanı protokolüdür. TAXII Şekil 2.4'te görüldüğü gibi iki tür veri paylaşım modeli sunmaktadır.



Şekil 2.4. TAXII'de sunulan STİ paylaşma haberleşme modelleri

Bunlardan birincisi temsili durum transferi (representational state transfer, REST) uygulama programlama arayüzü (application programming interface, API) arayüzüdür, diğeri ise STİ üreticilerinin oluşturduğu bilgiyi tüketicilere anlık gönderebileceği, yayın yapan tarafa abone olma yöntemiyle çalışan web soket haberleşme arayüzüdür. TAXII sunucusu, özellikle STIX gösterim şeması başta olmak üzere, pek çok formatta hazırlanmış olan STİ sunabilecek şekilde tasarlanmıştır ve ihtiyaçlara göre özelleştirilebilmektedir.

2.5. Heterojen Bilgi Ağları

HBA, farklı türdeki veri veya bilgi türlerini içeren ağ yapılarıdır. Bu ağlar, çeşitli formatlarda ve yapısal farklılıklara sahip verileri entegre etmek ve bir araya getirmek için kullanılmaktadır. Bu sayede, HBA farklı kaynaklardan gelen bilgileri birleştirerek daha kapsamlı bir görünüm oluşturmaya imkân sağlamaktadır [56]. Sosyal ağlar, bankacılık, biyoloji ve bilişim sistemleri [57] gibi çeşitli etkileşimli bileşenlerden oluşan birçok gerçek dünya sistemi alanlarında karmaşık çoklu sistem ilişkilerini tanımlayan HBA olarak soyutlanabilmektedir. Soyutlama sayesinde gerçek dünyadaki karmaşıklığın istenilen detay seviyesine indirgenmesi mümkündür. Soyutlama sonucunda gerçek dünyadaki varlık türleri ve aralarındaki ilişki türleri içinden seçim yapılarak, anlaşılabilir, analiz edilebilir, bilimsel olarak anlaşılır modeller üretilebilmektedir.

Hem homojen hem de heterojen bilgi ağları üzerine yapılan araştırmalar, teorik model oluşturma [58], topluluk tespiti [59-61], bağlantı tahmini [62,63], yüksek düzey organizasyonu [64], kümeleme [65] ve ağ etkisi [66,67] gibi konuları içeren bir disiplin haline gelmiştir. Homojen ağlar üzerine yapılan araştırmaların çoğu oldukça olgun olsa da yalnızca bir tür düğüm ve ilişkiyi dikkate almakla sınırlıdırlar. Bu da birden fazla düğüm ve ilişki türünü içeren çoğu gerçek dünya sistemine karşıt bir durum oluşturmaktadır. Bu nedenle, bir homojen ağı kullanarak gerçek dünyaya ait karmaşık bir sistem tanımlandığında, bilgi kaybı yaşanmaktadır.

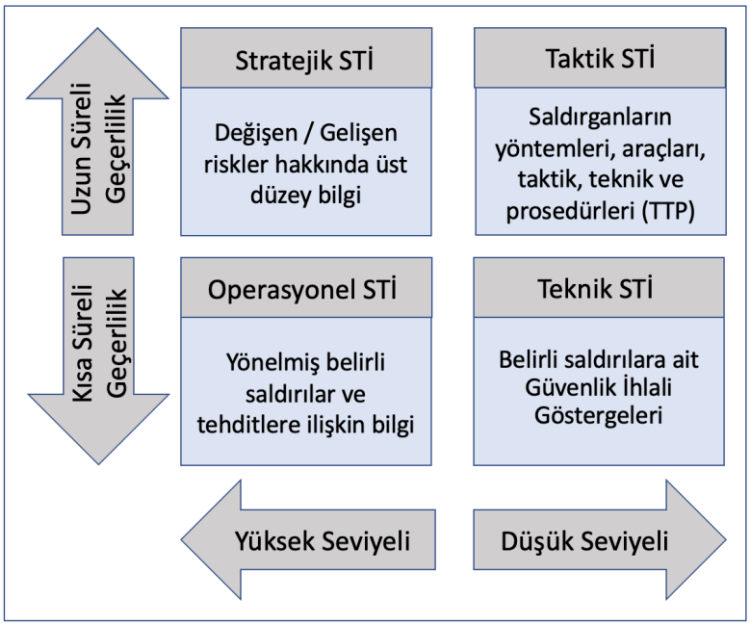
HBA ile ilgili temel kavramlar incelediğinde, düğüm, ilişki, meta-yol (meta-path) ve meta-çizge (meta-graph) ile karşılaşılmaktadır. Düğüm, ağdaki bir varlığı ifade etmektedir. Örneğin, IP adresi, tehdit aktörü veya alan adı, düğüm olarak ifade edilebilmektedir. İlişki, düğümler arasındaki bağlantıyı belirtmektedir, varlıklar arasındaki etkileşimleri veya ilişkileri göstermektedir. Meta-yol, ağdaki düğüm türlerinden ve kenar türlerinden oluşan

bir diziyi tanımlar. Belirli anlamsal örüntüleri içeren yapılardır. Meta-çizge, bir ağın genel yapısını ve özelliklerini belirlemek için kullanılmaktadır. Bir HBA'nın yapısını ve ilişkilerini tanımlayan, ağın kendisinden daha yüksek seviyeli bir modeldir. Meta-yollar, ağdaki desenleri ve yapısal özellikleri daha geniş bir perspektiften anlamak için kullanılmaktadır.

3. LİTERATÜRDEKİ ÇALIŞMALAR

Tez çalışmasında STİ içeren HBA'ların oluşumu ve bu ağları APT saldırı grupları odağında zenginleştirmek için bir model oluşturmak amacıyla yola çıkılmıştır. STİ'yi oluşturan veri, birçok kaynaktan elde edilmektedir. Farklı kaynaklardan elde edilen verinin, kendi içinde ilişkilendirilmesi, varlığından haberdar olunmayan bağlantıların kurulması ve tahminler üreterek yeni bağlantıların önerilmesi hedeflenmektedir.

Bu bölümde siber tehditlerin tespit edilmesinde ve saldırıların önlenmesinde yararlanılan STİ kaynaklarının ve yöntemlerinin belirlenmesi amaçlanmaktadır. STİ, literatürde dört seviyede incelenmektedir. Her bir seviyenin amacı, üretilme şekli, kaynakları, zorlukları ve kullanıcısı birbirinden farklıdır. Tez çalışmasında tamamı açık kaynaklı olan teknik, operasyonel ve taktiksel STI ifade eden veri ve bilgi kaynaklarından yararlanılmaktadır.



Şekil 3.1. STİ türlerinin geçerlilik eğilimleri ve kullanım ömürleri

STİ türleri iki boyutta incelendiğinde, taktik ve teknik tehdit istihbaratı düşük seviyeli istihbarat olarak karşımıza çıkmaktadır. Stratejik ve operasyonel tehdit istihbaratı ise yüksek seviyeli olarak adlandırılmaktadır. Şekil 3.1'de görüldüğü üzere tehdit istihbaratı türleri geçerlilik sürelerine göre incelendiğinde, stratejik ve taktik tehdit istihbaratı uzun

süre geçerliliklerini korumaktadır. Öte yandan operasyonel ve teknik tehdit istihbararatı ise kısa vadeli ve hemen kullanılmaya yönelik bilgi içermektedir.

3.1. Teknik STİ Üzerine Çalışmalar

Teknik STİ temelde GİG'lerden meydana gelmektedir. Bu GİG'ler, zararlı IP adresleri, URL'ler, alan adları, zararlı yazılım dosya imzaları ve oltalama saldırılarına karışmış e-posta adresleri gibi net bilgiler içermektedir. Diğer seviyelerdeki STİ türlerine göre üretilmesi daha kolaydır. Fakat üzerinde çalıştığı bilgi varlıklarının doğası gereği geçerliliğini çabuk kaybetmektedir. Bu sebeple teknik STİ'yi oluşturan GİG'lerin oluşma zamanı ve keşfedilme / fark edilme zamanının bilinmesi ve korunması önemlidir. Teknik STİ, tehditlere karşı hızlı önlem almak için gereklidir. Çizelge 3.1'de teknik STİ'den yararlanan çalışmalara yer verilmektedir.

Çizelge 3.1. Teknik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
A Cause-Based Classification Approach for Malicious DNS Queries Detected Through Blacklists [68]	Zararlı alan adı içeren blok listeleri	Ağda yapılan DNS sorgularının içerdiği alan adlarının blok listelerinde var olma durumu kontrol edilmektedir. DGA ile üretilen alan adlarına yapılan sorgular, alan adının kullanılmadığı durumlarda saldırganın kontrolündeki bir IP adresine çözülmektedir.	DNS sorgularının ağda şifresiz olarak taşınıyor olması bu sorguların manipüle edilmesini kolaylaştırmaktadır.	Teknik STİ kullanılmasına rağmen alan uzmanı desteğine yoğun bir şekilde ihtiyaç duyulmaktadır. Zararlı alan adı içeren blok listelerinin güvenilir olduğu varsayılmaktadır.
A Study on a CTI Analysis Platform for the Proactive Detection of Cyber Attacks Based on Automated Analysis [69]	Geçmiş saldırılara ait IP adresleri, alan adları ve zararlı yazılım dosya imzaları	IP adresleri, alan adları ve zararlı yazılımlara ait dosya imzaları OSINT yöntemleri kullanılarak toplanmaktadır. Elde edilen GİG'ler arasında ilişkiler kurulmaktadır.	GİG'ler arasında ilişkinin kurulmadığı kaynaklardan OSINT toplayarak varlıklar arasında somut ilişkiler kurulmuştur. Elde edillen teknik tehdit istihbaratının zenginleştirilmesi sağlanmıştır.	Toplanan verileri ilişkisel veritabanı yönetim sisteminde tutulmaktadır. Toplanan veri ontolojik olarak NoSQL bir veritabanında tutulmaya daha uygundur.

Çizelge 3.1. (devam) Teknik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
APT Traffic Detection Based on time transform [70]	Zararlı yazılım veri kümesi, DNS sorgu trafiği veri kümesi	Zararlı yazılımlar çalıştırılarak komuta kontrol sunucularına erişim için yaptıkların DNS sorguları izlenmiştir. Üniversite ağındaki DNS trafiği ile karşılaştırılarak davranışsal farklılıkları ortaya çıkarılmaktadır.	Zararları yazılımların komuta kontrol sunucuları ile olan iletişimlerinde düşük paket boyutlarıyla ve yavaş olacak şekilde hız ayarlaması yaptığı belirlenmiştir. DNS sorgu sonuçlarının nasıl teknik STİ olarak kullanılabileceği gösterilmektedir.	Geliştirilen yöntem zararlı yazılımlar ile yapılacan sıfır-gün saldırıları için etkili değildir. Tehdit istihbaratı açısından bakıldığında geliştirilen yöntem DGA tespiti yapmadığı için kolaylıkla aldatılabilir olduğu değerlendirilmek-tedir.
Blacklist-based Malicious IP Traffic Detection [71]	Açık kaynak IP adresi kara listeleri kullanılmak-tadır.	Açık kaynaklı kara listeler günlük olarak toplanmaktadır. Günlük olarak güncellenen ve tekilleştirilen kara liste ile ağda gözlemlenen hedef IP adresleri kıyaslanmakta-dır. Eşleşme olduğu durumda alarm üretilmektedir.	Yaygın olarak saldırılarda kullanılan IP adreslerinin raporlanması sonucunda oluşturulan IP adresi kara listelerinin teknik STİ olarak nasıl kullanılabileceği gösterilmektedir.	Ağdan internete çıkan trafik incelenerek hedef IP adresinin, oluşturulan kara listede var olma kontrolü ortama 270 ms ve 55 ms stadart sapma ile yapılmaktadır. Bu süre zararsız trafiği de bloke edebilecek uzunlukta olduğu için sadece tespit yapılmaktadır. Önerilen model, tehditi önleyici nitelikte değildir. Saldırganlar komuta kontrol sunucuları vekil (proxy) sunucuların arkasına koyabildiği için asıl zararlı IP adresi gizleyebilmekte-dir. Ayrıca bulut tabanlı güvenlik duvarları zararlı alan adının çözüldüğü IP adresini gizleyebilmekte-dir.

Çizelge 3.1. (devam) Teknik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
BotDet - A System for Real Time Botnet Command and Control Traffic Detection [72]	Zararlı IP adresi kara listesi, zararlı SSL sertifikası kara listesi tespiti, zararlı alan adı karalistesi, Tor düğümü listesi	Ağda gerçekleşen trafik üzerinden DNS sorgulamaları, SSL sertifikaları, hedef IP adresleri toplanarak harici teknik STİ kaynaklarından sorgulanmakta-dır. Zararlı olduğu bilgisi edinilen hedeflere giden trafik engellenmekte-dir.	Araştırmada STİ üretiminde SSL sertifikaları, DNS sorgusu, pasif DNS verisi ile Tor düğümlerinden nasıl yararlanılabilece-ği ve trafik verisinin anlamlandırılaca-ğı adreslenmektedir.	Sistemin başarısı karalistelere bağlıdır. Açık kaynaklardan elde edilen kara listelerin doğru ve kapsamlı veri içerdiği varsayılmaktadır. Geçmişte tespit edilmiş saldırıların STİ araştımacıları tarafından raporlanması durumunda sistem etkili olmaktadır. Bilinmeyen GİG'lere karşı herhangi bir çözüm sunmamaktadır.

Bilgisayar ağlarının karşılaştığı en önemli tehditlerden birisi zararlı yazılımlardır. Enfekte olmuş cihazların tespit edilmesi için kullanılan sıkça başvurulan yöntemlerden birisi ağdaki trafiğin monitörlenmesi ve kara listelerde yer alan kurallar ile eşleşme görüldüğü durumlarda iletişimin her iki ucundaki sistemlerin ağ ile olan ilişkisinin kesilmesine dayanmaktadır. Fakat bu yöntemde iki problem bulunmaktadır. Hiçbir kara liste tamamen güvenilir değildir. Ayrıca kara listeler içerdikleri kuralların geçerliliği ve doğruluğu ile ilgili yeterince kanıt içermemektedir. Bu sebeple ağ yöneticileri trafiği analiz etmek için başka yöntemlere başvurmaktadır.

Satoh ve diğerleri [68] kara listeler aracılığıyla tespit edilen kötücül DNS sorgularını nedenlerine göre sınıflandırmayı hedefleyen bir yöntem önermektedir. Yazarların böyle bir yöntem önermelerinin temelinde sahip oldukları bir gözlem bulunmaktadır. Araştırmacılara göre, zararlı bir iletişimi oluşturan zararlı DNS sorgularının, öncesinde ve sonrasında gerçekleşen çevreleyici sorgular, zararlı sorgunun gerekçesini belirlemekte etkili olmaktadır. Önerilen sebep tabanlı sınıflandırıcı ile yapılan deney sonucuna göre 388 adet zararlı sorgu 3 kategori altında kümelenmektedir. Sınıflandırma yapabilmek için Word2Vec ve Gaus karışım modeli (Gaussian mixture model, GMM) makine öğrenmesi algoritmaları kullanılmıştır. Sebeplerine göre sınıflandırılmış DNS sorguları sayesinde ağ yöneticileri sınırlandırılmış bir veri üzerinde çalışabilir hale gelmektedir. Çalışmada ifade edildiği üzere, zararlı yazılımların iletişimi başlarken öncelikle DNS sorgulama işlemi

gerçekleşmektedir. Ayrıca DNS sorgulama işlemi şifresiz bir iletişim ile gerçekleşmektedir. Bu sebeple zararlı yazılım kaynaklı siber tehditlerin tespit edilmesinde DNS sorguları oldukça etkili bir STİ olarak kabul edilmektedir.

Siber saldırıların proaktif olarak tespit edilmesi için başvurulan yöntemlerden birisi de geçmişe yönelik toplanmış STİ verilerinden yararlanmaktadır. Gerçekleşmiş saldırılardan elde edilen STİ, birçok çalışmanın temelini oluşturmaktadır. Bu verilerin oluşturulması, toplanması, paylaşılması ve analiz edilmesi konuları sıkça çalışılan konulardandır. Siber saldırılar 2 temel fazda ele alınmaktadır [73]. Bu fazlar, saldırı öncesi ve saldırı sonrası olarak adlandırılmaktadır. Saldırıların tespit edilebildiği durumlarda, saldırı sonrası olay raporları hazırlanmaktadır. Bu raporların hazırlanması sonucunda saldırıda kullanılan GİG'ler ortaya çıkarılarak STİ üretilebilmektedir. Açık kaynak STİ sağlayıcıların paylaştığı, veriler çoğunlukla bu tür verilerden meydana gelmektedir. Bu tür STİ verileri kullanılarak, aynı GİG'ler ile karşılaşan tarafların önlemler alması hedeflenmektedir. Fakat karmaşık ve etkili saldırılarda, saldırganlar açığa çıkmış GİG'leri kullanma eğiliminde değildir [74]. Bir STİ'nin en değerli olduğu durum önleyici olduğu durumdur. Henüz bir saldırıya karışmamış, daha önce karşılaşılmamış GİG'lerden oluşan bir teknik STİ önceliyici olma noktasında oldukça değerlidir. Aynı şekilde daha önce bir saldırıya karışmış olsa da yeni bir saldırıda kullanılmadan önce önlem alma noktasında geçerliliğini koruyan GİG'ler barındıran STİ'ler de çok değerlidir.

Kim ve diğerleri geçmiş saldırılardan elde edilen verileri toplayarak analiz eden ve bu veriler arasındaki ilişkileri ortaya çıkaran bir araştırma sunmaktadır. Çalışmalarında ele aldıkları kullanım durumu, siber güvenlik analistlerinin geliştirilen platforma sahip oldukları saldırı verilerini girmeleri ve geçmiş saldırı verileri ile karşılaştırılarak saldırının hangi fazda olduğunu ortaya çıkarmak üzerinedir [69]. Araştırmacılar önerdikleri platformda otomatik bir şekilde saldırı bilgilerini toplamayı hedeflemektedir. OSINT kaynaklarından elde edilen GİG'ler; IP adresleri, alan adları ve zararlı yazılımlara ait dosya imzalarıdır. Fakat bu varlıkları birbirinden kopuk bir şekilde toplamak üzerinde analiz yapılmasını mümkün kılmamaktadır. Araştırmacılar IP adresleri ile alan adları, alan adları ve IP adresleriyle de zararlı yazılımlara ait dosya imzaları arasında ilişki kurmaktadır. Bu çalışmada GİG'lerin depolanması için ilişkisel veritabanı kullanılmaktadır. Fakat IP adresleri, alan adları ve dosya imzaları arasındaki ilişkilerin tutulması konusunda ilişkisel veritabanı yerine NoSQL veritabanı kullanılmasının daha uygun olduğu değerlendirilmektedir.

APT'ler günümüzdeki siber saldırılar arasında önemli bir yere sahiptir. APT saldırılarının gerçekleşmeden önce tespit edilmesi oldukça zordur [75]. 2000'li yılların başından itibaren APT saldırıları siber casusluk faaliyetlerinde kullanılan önemli bir saldırı tipi olarak ortaya çıkmıştır [51]. Bu kategorideki siber saldırılar incelendiğinde saldırganların pek çok teknik ve yöntemi bir araya getirerek kullandığı görülmektedir. Bu yöntemler arasında oltalama e-postaları, zararlı yazılımlar, sosyal mühendislik ve botnetlerin kullanılması gibi pek çok saldırı yöntemi bulunmaktadır. APT kaynaklı saldırılar, saldırganların motivasyonu ve bu motivasyon oluşmasına neden olan hedefin sahip olduğu değer sebebiyle oldukça kararlı ve sistematiktir. APT kaynaklı saldırıların tespit edilmesi de bu sebeple görece zordur. APT saldırıları gerçekleştikten sonra yapılan çalışmalar ise, saldırının nasıl gerçekleştiğini anlamaya ve hasarın tespitine yöneliktir. Öte yandan bu saldırıların anlaşılması ve önlenmesine yönelik bir yol haritası oluşturulmasını amaçlayan çeşitli çerçeve modeller bulunmaktadır. Bu modellerden en önemlisi MITRE Att&ck ve Lockheed Martin Cyber Kill Chain modelleridir. Literatürde APT tespiti için yapılan çalışmalar incelendiğinde, bu alanda da makine öğrenmesi yöntemlerine başvurulduğu görülmektedir.

APT kaynaklı saldırıların incelenmesi, STİ üretilmesi ve sahip olunan istihbaratın zenginleştirilmesi için oldukça değerli bir alan olarak karşımıza çıkmaktadır. Saldırı gerçekleştikten sonra elde edilen bulgular ve deliller başka saldırıların önlenmesinde kullanılabilmektedir. APT kaynaklı saldırıların profillenmesi, kullanılan TTP'lerin ortaya çıkarılması, bulgu ve delillerin üzerinden elde edilebilecek henüz haberdar olunmayan potansiyel GİG'lerin keşfedilmesi için bir kapı aralayabilmektedir.

İncelenen çalışmada, Lu ve diğerleri APT kaynaklı saldırıların tespit edilmesi için yüksek başarıya sahip bir yöntem önermektedir. Geliştirilen yöntem oluşturulan test ortamında 2.94% hata oranı ile özellikle zararlı yazılımların kullanıldığı APT saldırılarını tespit edebilmektedir [70]. Fakat çalışmada önerilen yöntem, bilindik zararlı yazılımların yer aldığı test verileri üzerinde etkili olmaktadır. Sıfır-gün saldırısında kullanılmak üzere geliştirilen bir zararlı yazılım çalışmada önerilen yöntem ile tespit edilememektedir. APT kategorisinde saldırı düzenleyebilecek nitelikteki tehdit aktörlerinin zararlı olduğu ortaya çıkmış bir yazılımı tekrar yeni bir saldırıda kullanma ihtimali de oldukça düşük olduğu değerlendirilmektedir.

Araştırmacılar üniversite ağındaki trafiği kayıt altına alarak kendi eğitim ve test veri kümelerini üretmiştir. Yapılan çalışma, APT amaçlı zararlı yazılımların komuta kontrol

(command & control, C&C) sunucuları ile kurduğu haberleşmenin, botnetlerin oluşturduğu trafik ile oldukça benzerlik gösterdiği bilgisine dayandırılmıştır. Araştırmacılara göre, her iki haberleşmenin ortak yönü, periyodik aralıklarla aktif hale gelmek üzere programlanmış olmalarıdır. İncelenen çalışmadaki zararlı yazılım veri kümesi üniversite ağındaki bir aylık trafiğin kayıt altına alınması ile elde edilmiştir. APT saldırılarında kullanılan zararlı yazılımlar ile ilgili olarak, güvenlik duvarları ve saldırı tespit sistemleri tarafından yakalanmamak için düşük ve yavaş seviyede ağ trafiği oluşturdukları bilinmektedir. Bu yaklaşım incelendiğinde, birçok zararlı yazılım küçük boyutlu paketleri uzun sürede kaynak ile hedef sistemler arasında transfer etmektedir. Bu iletişim yöntemi Stuxnet [76], Duqu [77] ve Flame [78] gibi saldırılarda görülmektedir ve zararlı yazılımın ağda oluşturduğu trafik, olağan trafiğin hacimsel olarak yaklaşık 1%'ini oluşturmaktadır.

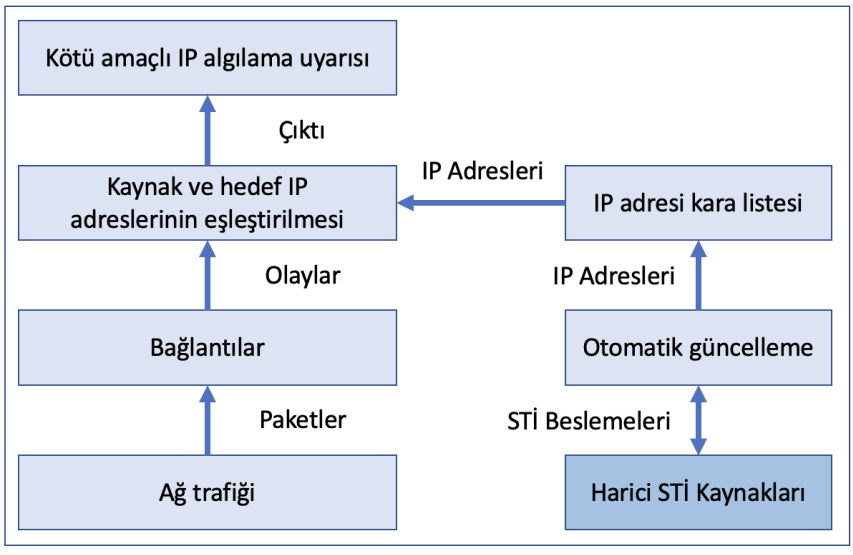
Deney ortamını hazırlamak için yaklaşık 16 çeşit APT zararlı yazılımı ve yaklaşık 2000 adet anormal trafik akışından meydana gelen bir veri kümesi oluşturulmuştur. Zararlı olduğu bilinen bu yazılımlar üniversite ağından izole olarak çalıştırılarak ağ trafiği kayıt altına alınmıştır ve üniversitenin ağ trafiği ile birleştirilmiştir. Üniversite ağında gözlemlenen normal DNS trafiği, zararlı yazılımların oluşturduğu DNS trafiğine göre çok daha hızlı ve rastgele aralıklarla gerçekleşmektedir. Zararlı yazılımların komuta kontrol sunucuları ile olan iletişimi esnasında yapılan DNS sorguları periyodik aralıklarla gerçekleşmektedir.

Bu çalışmanın kapsama alınmasındaki gerekçe; normal bir DNS trafiğinin profillenmesinin ardından, zararlı yazılımların oluşturduğu DNS trafiğinin profillenmesi sayesinde anomali tespiti mümkün hale gelmiş olmasıdır. APT saldırısında kullanılan zararlı yazılımın oluşturduğu trafik ayırt edildiği durumda oldukça değerli bir STİ oluşmaktadır. Zararlı yazılımların iletişime geçtiği komuta kontrol sunucusunun yayın yaptığı alan adının çözüldüğü IP adresi elde edilmiş olmaktadır. Bu bilgi elde edildiği noktada, ağdaki varlıkların ilgili IP adresi ve alan adı ile iletişimi engellenmesi mümkün olmaktadır.
Literatürde Teknik STİ üzerine yapılan çalışmalar incelendiğinde, doğrudan STİ'nin üretilmesine yönelik yöntemlerin oldukça az olduğu görülmektedir. Tehdit istihbaratının üretilmesindeki zorluk; yüksek kaliteli, aksiyon alınabilir, doğrulanabilir ve önleyici istihbaratın oluşmasına imkân sağlayan kaynakların tespit edilmesinin karmaşık bir işlem olmasından ve bu kaynaklardan elde edilen verilerin nasıl yorumlanacağının bilinmemesinden kaynaklanmaktadır. Belirtilen niteliklere sahip STİ, siber güvenlik alanında uzman seviyede alan bilgisi gerektirmektedir. Tecrübeye ve alan bilgisine bağlı keşfedilen yöntemler de ticari

olarak kullanılması sebebiyle örtülü bilgiye dönüşmektedir. Özellikle stratejik ve taktik STİ üretiminde alan uzmanlığı ve disiplinler arası etkileşime olan ihtiyaç artmaktadır. Fakat teknik STİ ve konu aldığı GİG'ler kullanıma hazırdır.

Bu şartlar altında teknik STİ'ye erişebilmek için ücretli ve açık kaynak olmak üzere temelde iki kaynak türü bulunmaktadır. Kurumların güvenlik operasyon merkezlerinin (security operations center, SOC) işlediği STİ, kendi ürettikleri ve yorumlayarak zenginleştirdikleri STİ'ye ek olarak bu iki kaynak türünden gelmektedir. Doğrulamaya muhtaç ve geçerlilik süresinin belirlenmesi gereken bir teknik STİ türü olarak karşımıza kara listeler çıkmaktadır. Siber güvenlik çözümü üreticileri ve açık kaynak STİ sunan organizasyonlar çeşitli kara listeler yayınlamaktadır. Bu kara listeler IP adresleri, alan adları, URL'ler ve zararlı yazılımlara ait dosya imzaları içermektedir.

İncelenen çalışmaların bir diğerinde Ghaifir ve diğerleri kara liste tabanlı zararlı trafik tespiti yapmak için bir yöntem önermektedir [71]. Ortaya çıkarmaya çalıştıkları trafik, zararlı yazılımların haberleştikleri komuta kontrol sunucuları ile zararlı yazılımın enfekte ettiği makineler arasındaki iletişimi oluşturmaktadır. Amaçlarına ulaşmak için kara liste sunan güvenlik bildirimlerini günlük olarak bir araya getirerek güncel ve tekilleştirilmiş kara listeler elde etmekte ve gerçek zamanlı tespit yapmaktadırlar. Şekil 3.2'de görüldüğü üzere ağdaki çıkış ve giriş trafiği, toplanan kara listelerdeki IP adresleri ile karşılaştırılmaktadır. Eşleşme olduğu durumda içe gelen ve dışa çıkan trafik engellenmektedir.



Şekil 3.2. Zararlı IP trafiği tespit yöntemi [71]

Siber uzayda yıkıcılığı çok yüksek olan tehditler arasında fidye zararlı yazılımları ve botnet zararlı yazılımları bulunmaktadır. Botnet yazılımları tarafından enfekte olan makineler, uzak saldırıların gerçekleştirilmesi ve kripto madenciliği gibi pek çok amaçla kullanılabilmektedir [79,80]. Ayrıca botnet ağlarına dahil olan makineler hizmet engelleme (denial of service, DoS) ve dağıtık hizmet engelleme (distributed denial of service, DDoS) saldırılarında kullanılabilmektedir. Kritik altyapılar da dahil olmak üzere saldırganların hedefinde olabilecek kurum, kuruluş ve şirketler bu saldırılardan zarar görebilmektedir [81-83]. Botnetleri yöneten komuta kontrol sunucuları ve botnetlere dahil olan makinelerin tespit edilmesiyle değerli STİ üretilebilir ve sahip olunan STİ zenginleştirilebilir.

İncelenen araştırmalardan birinde Ghafir ve diğerleri botnetlerin komuta ve kontrol sunucuları ile olan trafiğinin gerçek zamanlı tespiti için bir sistem önermektedir [72]. Önerdikleri sistem temelde iki işlevi yerine getirmektedir. Bunlardan birincisi, botnet içindeki komuta kontrol sunucularının tespit edilmesidir. Araştırmanın ikinci önemli çıktısı ise üretilen alarmların doğruluğunu artırmayı hedefleyen bir korelasyon çerçeve modelinin geliştirilmesidir. Bu modelin başarısı 82.3% doğru-pozitif ve 13.6% yanlış-pozitif olarak belirtilmektedir.

Araştırmacıların botnet makineleri ile komuta ve kontrol sunucuları arasındaki trafiği tespit etmek için kullandıkları dört farklı yöntem bulunmaktadır:

- Zararlı IP adresi tespiti,
- Kötücül SSL sertifikası tespiti,
- Alan adı değişikliği tespiti,
- Tor bağlantısı tespitidir.

Zararlı IP adreslerinin tespit edilmesi için kullanılan yöntem güvenlik hizmeti sunan organizasyonlar tarafından dağıtılan kara liste bildirimlerinin kullanılmasına dayanmaktadır. İzlenen ağda trafik oluşturacak yeni bir bağlantı isteği tespit edildiğinde bu trafiğin çıkış tarafındaki IP adresi elde edilmektedir. Elde edilen IP adresinin kara liste bildirimlerinde yer alma durumu kontrol edilmektedir. Kara listede ilgili IP adresi ile karşılaşılırsa, haberleşme engellenmektedir.

Kötücül amaçlı SSL sertifikası tespitindeki motivasyon ise C&C sunucularının botnetteki makineler ile olan iletişimlerini şifreleyerek yerine getirme eğilimine dayanmaktadır. SSL sertifikası kullanılarak botnetteki makineler C&C sunucuları ile şifrelenmiş ve böylece tespit edilmesi daha zor bir iletişim kurmaktadır. Zararlı SSL sertifikalarının tespiti için de kara liste bildirimleri kullanılmaktadır. Elde edilen kara liste bildirimlerinde daha önce karşılaşılmış zararlı SSL sertifikalarının SHA1 parmak izleri ve sertifika seri numaraları bulunmaktadır. Test ortamı olarak kullanılan üniversite ağındaki trafik canlı olarak izlenerek, SSL sertifikaları elde edilmekte ve sahip oldukları SHA1 parmak izleri ve sertifika seri numaraları kara liste bildirimleri ile karşılaştırılmaktadır. Eşleşen kayıt bulunduğu durumda, ilgili sertifika ile karşılaşılan haberleşme engellenmektedir. Araştırmacılar makalenin yazıldığı tarih itibariyle Bro olarak bilinen, fakat bu doktora tezinin kaleme alındığı tarih itibariyle Zeek olarak adlandırılan açık kaynaklı bir ağ trafiği analiz aracını kullanmaktadır.

Alan adı değişikliği tespiti, botnetteki makinelerin C&C sunucuları ile haberleşmek için kullandıkları yazılımların analiz edildiği taktirde, sunucu alan adının doğrudan keşfedilmesini engellemek için kullanılan bir yönteme dayanmaktadır. Saldırganlar tarafından DGA algoritmaları geliştirilerek C&C sunucularının IP adresine çözülen çok sayıda alan adı üretilebilmektedir [84]. Kullanılan alan adlarının yasal yollar ve karşı önlemler alınarak kapatılması veya erişimin engellenmesi durumunda saldırganlar tarafından yeni alan adları devreye alınmaktadır. Böylece C&C sunucularının her zaman erişilebilir olması sağlanmaktadır.

Araştırmacılar ağda çok sayıda başarısız sonuçlanan DNS sorgusu yapan makineleri tespit ederek, onlara odaklanmaktadır. Çok sayıda başarısız sonuçlanan DNS sorgusu trafiği tespit edilirse, ağda olası bir botnet zararlı yazılımı tarafından enfekte olmuş makine olduğu çıkarımı yapılmaktadır. Her ne kadar DNS sorgusu başarısız sonuçlanmış olsa da bu şekilde keşfedilmiş bir alan adına sahip olunduğu taktirde, ilgili alan adı STİ servisleri sorularak pasif DNS bilgisine ulaşılabilmektedir. Pasif DNS bilgisi sayesinde DGA tarafından üretilen alan adlarının hangi IP adresine çözüldüğü ortaya çıkarılmaktadır. Buradan anlaşıldığı üzere DNS ve pasif DNS bilgisi STİ üretimi ve zenginleştirilmesi için oldukça değerli ve kritik bir öneme sahiptir.

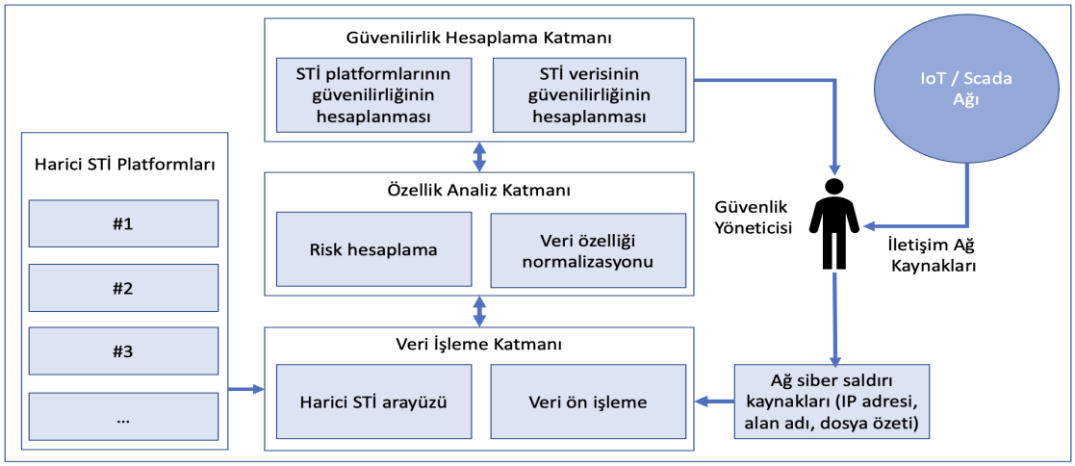
Tor bağlantılarının tespiti ise, olağan bir ağda gözlemlenmesi olası olmayan bir haberleşmenin varlığının ortaya çıkarılması motivasyonuna dayalıdır. Tor, ağdaki haberleşmenin şifrelendiği ve katılımcıların anonim olarak haberleştiği özelleşmiş bir ağdır. İnternet altyapısı üzerine bindirilmiş kendine özgü haberleşme protokolleri ile tasarlanmış bir ağdır. Tor haberleşmesinin takip edilemediğine dair bir algı olsa da bu yaygın bir yanlış düşüncedir. Sağladığı şifreleme gibi avantajlar sebebiyle, saldırganlar tarafından yoğun olarak kullanılmaktadır. Tor sunucuları katılımcıların erişim bilgilerini elde etmesi için açık olarak ilan edilmektedir. Araştırmacılar Zeek kullanarak izledikleri ağ trafiğinden elde ettikleri hedef ve kaynak IP adreslerini, yayınlanan ve bilindik Tor sunucu adresleri karşılaştırarak, olası zararlı haberleşmeleri tespit etmektedir.

Ghafir ve diğerlerinin önerdikleri sistemde yer alan tespit yöntemlerinin birçoğu, her ne kadar hazır kara liste bildirimlerine dayalı olsa da STİ üretimi için oldukça değerli kaynakları adreslemektedir. SSL sertifikaları, DNS ve pasif DNS ile TOR düğümleri STİ üretiminde ve mevcut STİ'nin zenginleştirilmesinde kullanılabilir.

STİ, bilgiye dayalı tehdit yönetim sistemi olarak da tanımlanmaktadır. Teknik STİ'nin temel amaçlarından biri, güvenlik bilgileri ve olay yönetimi (security information and event management, SIEM) sistemleri üzerinden toplanan IP adresleri, URL'ler ve dosya imzaları gibi ağ kaynaklarının sahip oldukları itibarı/repütasyonu ortaya çıkarmaktır. STİ'den yararlanarak bilişim altyapılarının güvenliğinin sağlanması için öncelikli olarak elde edilen bilginin güvenilir kaynaklardan toplanması gerekmektedir. Tehdit istihbaratı sunan farklı kaynaklardan elde edilen bilgiler birbiri ile çelişebilmektedir. Gong ve diğerleri, Virus Total, Threat Crowd, Alien Vault OTX ve Cymon platformlarından topladığı STİ içinden risk değeri, dosya imzaları, alan adları, alt alan adları, DNS kayıtları, Anti-virüs tarama sonuçları, IP adresleri, URL'ler, pasif DNS bilgileri ve eşleşen bilinen zaafiyetler ve açıklar (common vulnerabilities and exposures, CVE) kodlarını belirlemiştir [85].

Araştırmacıların önerdiği güvenilirlik belirleme modeline göre, Şekil 3.3'te görüldüğü üzere öncelikle STİ sunan platformlardan istihbarat toplanmaktadır. Ardından toplanan bilgi ön işlemeden geçmektedir. Ön işleme kapsamında, farklı kaynaklardan elde edilen STİ beslemeleri (feeds) arasındaki özellik farklılıkları ortaya çıkarılmakta ve tüm veriler aynı formata dönüştürülmektedir. Ön işlemeden sonra, risk hesaplama ve veri üzerindeki

özelliklerin normalizasyonu gerçekleştirilmektedir. Normalizasyon kapsamında eldeki veri içindeki özelliklerin sahip olduğu ölçek eşitlenmektedir. Ardından STİ bilgisinin edinildiği platformun ve gelen bilginin güvenilirliği hesaplanmaktadır.



Şekil 3.3. STİ güvenilirliği belirleme modeli [85]

Güvenilirliğin belirlenmesinde farklı kaynaklardan elde edilen verinin çelişkiler içerip içermemesi bir ölçüt olarak kullanılmaktadır. Güvenilirlik bilgisi, güvenlik yöneticisine iletilmektedir. Güvenlik yöneticisi ise yönetmekte olduğu ağda toplanan SIEM verisindeki göstergeleri platformlardan gelen istihbarat içinde sorgulamaktadır. Güvenilirliği yüksek platformlardan birisinden gelen tehdit istihbaratı ile eşleşme görüldüğünde, gerekli aksiyonu almak için fırsat elde edilmiş olmaktadır. Kendi STİ üretme sürecini işletemeyen ve OSINT sunan platformlardan temin eden organizasyonlar için, en önemli konulardan birisi elde ettiği tehdit istihbaratının güvenilirliğidir. Doğru olmayan STİ ile alınan kararlar sonucunda pek çok sorun ile karşılaşılmaktadır. Yanlış STİ kullanımı ağda güvenli bir taraf ile kurulacak iletişim kesilebilir veya güvenli olmayan bir taraf ile iletişim kurulmasına sebep olunabilmektedir. Üretilen STİ'nin güvenilirliğinin sınanması ve çelişen bilgilerin tespit edilerek güvenilirliğin seviyelendirilmesi de bir zenginleştirme faaliyeti olarak görüldüğü için ilgili çalışma tez çalışması kapsamnda alınmıştır.

3.2. Taktik STİ Üzerine Çalışmalar

Taktik STİ, tehditin altında yatan, hedefini maruz bıraktığı sorunla mücadele etmek için daha geniş bir tehdit perspektifi elde etmeyi hedeflemektedir. Organizasyonlar genellikle spesifik bir tehdite odaklandığı için diğer tehditleri de görüş alanlarına almalarını sağlayacak perspektifi

oluşturamamaktadır. Taktik STİ yakın gelecekte saldırganların ne tür yöntemlerle faaliyetlerini yürüteceklerini anlamaya odaklanmaktadır. Elde edilen taktik STİ, teknik STİ'ye görece daha uzun bir ömre sahiptir. Çünkü saldırganların kullandığı TTP'ler kolaylıkla ve hızlı bir şekilde değişmemektedir.

Operasyonel STİ, saldırıların arkasındaki aktörleri daha iyi anlamak için tehdit aktörü profillemeyi amaçlamaktadır. Profilleme süreci zorluklarla doludur, çünkü tehdit aktörleri etkili, fırsatçı ve düşük riskli teknikleri tercih etmektedir. Bu durum da tehdit aktörü profilleme ve operasonel STİ oluşturmayı zorlaştırmaktadır. Profilleme işleminin amacı ise tehdit aktörlerinin sonraki hareketlerini tahmin etmektir. Her saldırının ardında cevaplanması gereken üç temel soru bulunmaktadır. Bunlardan birincisi saldırganın kim veya kimler olduğu, ikincisi saldırının neden gerçekleştirildiği ve üçüncüsü saldırının nasıl yapıldığıdır. Saldırıların ardında tekil bir saldırgan olabildiği gibi organize bir grup ile de karşılaşılabilmektedir. Saldırının etkilerini anlamanin ve hedef alınan varlıkları doğru tespit etmenin, saldırganın motivasyonunu ve niyetini ortaya çıkarmak için önemli olduğu değerlendirilmektedir.

Saldırıların nasıl gerçekleştiğini anlamak tehdit aktörünün sahip olduğu TTP'leri ortaya çıkarmaktadır ve taktik STİ özellikle bu soruya yanıt olabilecek bilgileri edinmeyi hedeflemektedir. Çizelge 3.2'de taktik STİ'den yararlanarak saldırı tespiti kabiliyeti geliştiren çalışmalara yer verilmektedir.

Çizelge 3.2. Taktik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
Adversarial Fingerprinting of Cyber Attacks based on Stateful Honeypots [86]	Saldırganların bal küplerine yaptığı saldırılardan elde edilen GİG ve TTP'ler taktik STİ olarak kullanılmakta-dır.	Bal küpleri oluşturularak saldırganların bu sistemler üzerindeki faaliyetleri günlük kayıtlarına yazılmaktadır. Log dosyaları manuel olarak incelenerek kullanılan TTP'ler keşfedilmeye çalışılmakta-dır.	Araştırmacılar durum bilgisini tutan (stateful) ve saldırganın bir bal küpü ile etkileşime geçtiğini farketmesini zorlaştıracak bir altyapı oluşturalarıdır. Bu sebeple elde ettikeri GİG ve TTP'lerin güvenilirliği yüksektir.	Saldırganın farklı IP adresleri üzerinden bal küpüne erişmesi durumunda yeni oturumlar oluşturması mümkündür. Bu durum saldırganın bal küpünün farkına varmasına sebep olabileceği değerlendiril-mektedir.

Çizelge 3.2. (devam) Taktik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
An Association Rule Mining-Based Framework for Profiling Regularities in Tactics Techniques and Procedures of Cyber Threat Actors [87]	STIX veri formatında ifade edilen hazır STİ dokümanları kullanılmakta-dır. STIX dokümanları içindeki TTP alanına yoğunlaşılmak-tadır.	Bağlantı kuralı madenciliği (association rule mining, ARM) tabanlı makine öğrenmesi algoritması olan FP-Growth kullanılmaktadır. Algoritma toplanan STIX verisini ağaç formuna dönüştürerek sık rastlanan örüntüleri keşfetmekte ve TTP'ler arasında ilişkiler kurmaktadır.	FP-Growth algoritmasının ağaç veri yapıları üzerinde örüntü tespitinde hesaplama maliyeti düşüktür. Bir milyon STIX dokümanı analiz edilerek 28 sık karşılaşına TTP örüntüsü ve TTP'ler arasında 155 ilişki kuralı keşfedilmiştir.	FP-Growth algoritmasının hesaplama maliyetinin düşük olduğu ifade edilmektedir, fakat kullanılan sistem konfigürasyonu ve performans ölçütleri hakkında sayısal veri sunulmamakta-dır.
A Markov Multi-Phase Transferable Belief Model for Cyber Situational Awareness [88]	SIEM veritabanı, IDS alarmları ve Windows işletim sistemi günlük kayıtları korele edilerek veri kümesi oluşturulmuştur.	MM-TBM kullanılarak Siber Ölüm Zinciri üzerinde saldırganların gerçekleştirebileceğ i olası aktivitelerin sırasını gösteren saldırı ağaçları oluşturulması sağlanmıştır. Saldırı adımları olasılıksal olarak ortaya çıkarılmıştır.	Markoz Zinciri modelinden STİ alanında yararlanmasına yönelik örnek bir çalışmadır. Saldırganların günlük kayıtları üzerinde bıraktığı delillerin Siber Ölüm Zincirindeki fazlar ile eşleştirilerek TTP'lerin elde edilmesi için detaylı bir çalışma sunulmaktadır.	Çalışmanın varsayımları arasına almadığı bir unsur olarak, SIEM araçlarını besleyen günlük kayıtlarını üreten sunucuların kesintisiz aynı detay seviyesinde ve aynı zaman sisteminde günlük tutması gerekmektedir. Gerçek hayattaki vaka çalışmalarında günlük kayıtlarının kesintisiz ve tüm sunuculardan beslenecek şekilde konfigürasyon hataları sebebiyle toplanamadığı bilinmektedir.

Kritik bilişim sistemlerinin siber saldırılardan korunması için saldırganların kullandığı yöntem, araç ve eylemlerin tanınması gerekmektedir. Saldırganların tanınması ekosistemdeki potansiyel tehdit aktörlerinin de tanınması için önemli bir arka plan bilgisi oluşturmaktadır. Saldırganlar tarafından hedef alınmış ve saldırılmış birçok başarılı ve başarısız girişim bulunmaktadır. Tehdit istihbaratı oluşturma perspektifinden bakıldığında, geçmiş saldırıların sistemler üzerinde bıraktığı izler, benzer teknoloji ve zafiyetlere sahip diğer sistemlerin hedef olma ve zarar görme potansiyeli hakkında bize fikir verebilmektedir. Siber saldırılar gerçekleştikten ve farkına varıldıktan sonra, saldırganların yöntemlerini, verdikleri zararı ve sistem üzerinde yürüttükleri

faaliyetleri ortaya çıkarmak için sunucuların ürettiği ortak günlük dosyaları (common log format, CLF) kullanılmaktadır. Birçok gerçek hayat senaryosunda görüldüğü üzere, bilişim altyapısını oluşturan sunucuların tamamı detaylı ve sürekliliği koruyacak şekilde günlük tutmamaktadır. Öte yandan sunucu ve son kullanıcı cihazlarının ürettiği günlük kayıtları birçok siber saldırının anlaşılması için kritik öneme sahiptir.

Tehdit istihbaratı üretmenin yöntemlerinden birisi de bal küpü (honeypot) kullanımıdır. Bal küpleri çeşitli zafiyetlere sahip teknoloji, yazılım ve donanım konfigürasyonları barındıran yapay olarak kurgulanmış saldırıya açık sistemlerdir [75]. Balküplerinden maksimum faydayı elde etmek için ise detaylı, sürekli, birleştirilebilir, ilişkilendirilebilir doğru içeriğe sahip günlük kaydı tutulması gerekmektedir. İncelenen çalışmalardan birisinde, Colajanni ve diğerleri bal küpleri kullanarak saldırganların dijital parmak izlerini oluşturmayı hedeflemektedir [86]. Çalışmaya göre saldırganlar bir kritik altyapıya sızmak için keşif fazını gerçekleştirdikten sonra, ana hedeflerine ulaşmak için bir basamak olarak kullanabilecekleri, güvenlik zincirindeki en zayıf halkayı kullanmaktadır. Saldırganlar en zayıf halka üzerinden içeri sızdıktan sonra asıl hedeflerine sessizce ulaşana kadar uzun süre aktivite yürütmeden beklemektedir.

Araştırmacılar, saldırganın sistem ile etkileşime geçtiği esnada oluşturulan oturuma (session) ait bilgileri koruyarak birden çok saldırganın aynı sistem üzerindeki faaliyetlerini birbirinden bağımsız olarak takip etmektedir. Çalışmanın güçlü yönü durum bilgisi tutabilen (stateful) bir bal küpü geliştirilmiş olmasıdır. Durumsuz (stateless) bal küpleri saldırganlar tarafından görece daha kolay anlaşılabilmektedir. Saldırganın yapay bir ortamda olduğunu anlaması ise toplanan GİG'lerin faydasız ve yetersiz olmasına sebep olmaktadır. İncelenen çalışmada durumsuz bal küplerinin sahip olduğu sorunlar aşağıdaki şekilde sıralanmıştır:

- Arka kapılar tarafından açılan herhangi bir yeni port ve arka kapı kodunun kendisi oturum kapandıktan sonra yok olmaktadır. Bu sıfırlama işlemini durumsuz bal küpleri yapmaktadır.
- Saldırganın bal küpü ile her etkileşime geçişinde .ssh/knownhosts dosyasına ssh parmak izini ekleme isteği. Bunun nedeni ise güvenli kabuk hizmeti sunmayı taklit eden bal küplerinin her güvenli soket kabuğu (secure socket shell, SSH) bağlantısında yeni barındırıcı (host) anahtarları kullanmasıdır.
- Saldırıya uğrayan sistemin dosya sisteminde bırakılan herhangi bir yazılım veya veri korunmamaktadır.

- Sistemin konfigürasyonunda güvenliği zayıflatacak yönde yapılan değişiklikler kalıcı değildir ve kaybolmaktadır.

İncelenen çalışmanın tezde yoğunlaşılan problemin çözümüne katkısı, bal küplerinden toplanan verilerin STİ üretimi sürecinde kullanılabilecek nitelikte olmasıdır. Oluşturulacak heterojen tehdit istihbaratı ağını oluşturmak ve zenginleştirmek için bal küplerinden elde edilen GİG'ler ve TTP'ler katkı sağlamaya elverişli ve değerli görülmektedir.

Daha önce belirtildiği üzere STİ; stratejik, operasyonel, taktik ve teknik olmak üzere dört kategori altında sınıflandırılmaktadır. Stratejik istihbarat, karar vericiler tarafından kullanılan, risklerin belirlendiği ve etki analizinin yapıldığı yüksek seviyeli bilgi içeren STİ türüdür. Operasyonel istihbarat, saldırgan forumlardan ve karanlık webden güncel zafiyetlerin ve sıfır-gün (zero-day) saldırılarının toplanmasıyla oluşmaktadır.

Taktik STİ, siber tehdit aktörlerinin kullandıkları TTP'lerden oluşan tehdit istihbaratı türüdür. TTP'ler tehdit aktörünün kurbanın işletim sistemi, ağ gibi sistem kaynakları veya uygulamaları ile etkileşime geçerken sergilediği davranışı temsil etmektedir. Tehdit aktörleri, saldırılarda kullanmak üzere kendi yöntemlerini ve uygulama adımlarını oluşturmaktadır ve bunları bir saldırının ortasında değiştirme eğiliminde değildir. Bu sebeple saldırganlar ile eşleştirilebilen, tanımlanabilen ve zamanında tanınan TTP'ler saldırıların engellenmesinde büyük öneme sahiptir. TTP'lerin zamanında tespit edilmesi için güvenlik analistlerinin bilindik ve yaygın saldırı yöntemleri ile ilgili bilgi birikimini vaka bazlı olarak uyarlaması gerekmektedir. Saldırganların mevcut saldırıları incelenerek ve kullandıkları TTP'ler ve onlar arasındaki ilişkiler ortaya çıkarılarak, tehdit aktörleri hakkında daha derin bilgi sahibi olunmaktadır.

Teknik STİ'yi oluşturan unsurlar, GİG olarak adlandırılan IP adresleri, alan adları, C&C sunucuları, dosya imzaları ve zararlı yazılımlara ait olduğu bilinen dosya özetleridir. Stratejik ve taktik STİ, operasyonel ve teknik STİ ile karşılaştırıldığında görece daha uzun ömürlüdür ve geçerliliğini koruma eğilimindedir [87]. Anwar ve diğerlerine göre siber güvenlik alanında saldırganların kullandıkları TTP'lerin bilinmesi önemli bir STİ kabul edilmektedir. Siber tehdit aktörlerinin kullandıkları TTP'ler ve bunlar arasındaki ilişkilerin bilinmesi saldırıya ait saldırı vektörünün belirlenmesi için etkili olmaktadır. İncelenen çalışmada, araştırmacılar STİ içeren dokümanlar üzerinde geliştirdikleri modeli çalıştırarak çeşitli TTP'ler üretmektedir. ARM tabanlı makine öğrenmesi yöntemleri ile elde edilen TTP'ler arasında bağlantı kuralları

keşfedilmektedir. TTP'ler üzerinden ilişki madenciliği yapmak için FP-Growth algoritması kullanılmaktadır. FP-Growth algoritması büyük veritabanlarında yer alan sık karşılaşılan örüntüleri ortaya çıkarmak ve bu işlemi yaparken gerçekleştirilen hesaplamaları sistem kaynaklarını minimum düzeyde kullanacak şekilde yapmaktadır.

Çalışmada yer alan bir milyon adet dokümandan oluşan veri kümesi üzerinde FP-Growth algoritması kullanılarak iki aşamalı bir yöntem uygulanmaktadır. Birinci aşamada veritabanı iki defa taranmakta ve bir ağaç veri yapısı üretilmektedir. Ağaç veri yapısı kendini tekrar eden yapıları tutmak için uygun olduğu düşünülmektedir. İkinci adımda ise FP ağacı üzerinde gezinerek sık sık görülen örüntüler ortaya çıkarılmaktadır.

Araştırmacılar, açık kaynak STİ içeren dokümanlar üzerinde geliştirdikleri çerçeve modeli uygulayarak 28 adet TTP'nin diğer TTP'lere göre çok daha yaygın olduğunu ifade etmektedir. Ayrıca geliştirilen sistem sayesinde belirlenen 28 adet TTP arasında 155 adet ilişki kuralı keşfedilmiştir.

Siber Ölüm Zinciri saldırı modelleme yöntemine göre saldırganların izleyebileceği olası yollar bulunmaktadır. Bu yolları tarif etmek için kullanılan yöntemler TTP olarak adlandırılmaktadır. Saldırganların kullandıkları TTP'ler de taktik STİ üretim sürecinin çıktılarından biridir. Özellikle APT gruplarının kullandıkları TTP'leri profillemek için geliştirilen Siber Ölüm Zinciri modeli tehdit istihbaratının ifade edilmesi ve transferi için kullanışlı bir yöntem sağlamaktadır. Saldırılarda kullanılan olası yolların tamamının farkında olmak, saldırganların izlediği spesifik yolları anlamak için faydalıdır.

Ioannou ve diğerleri [88] APT'lere karşı siber güvenlik uzmanlarının durumsal farkındalığını geliştirmek için MM-TBM'yi temel alan bir yöntem önermektedir. MM-TBM Siber Ölüm Zinciri üzerinde saldırganın gerçekleştirebileceği olası aktivitelerin sırasını gösteren saldırı ağaçları oluşturmayı sağlamaktadır.
Markov modelleri siber güvenlikte durumsal farkındalığı oluşturmak için kullanılmaktadır. Siber güvenlikte kesinlik içermeyen bilgilerin elde edinilmesi, durum farkındalığını olumsuz etkilemektedir. Bu sebeple Markov Zinciri matematiksel sisteminden yararlanılarak oluşturulacak olasılıklar durumsal farkındalığı artıracak yönde fayda sağlamaktadır. Araştırmacılar yaptıkları deneysel çalışmada kullanmak üzere saldırı vektörü olarak çeşitli CVE'leri seçmiştir. Seçilen saldırı vektörleri kullanılarak örnek ağ topolojisi üzerinden bir saldırı

ağacı oluşturulmuştur. Çalışmada MM-TBM algoritmasının olası saldırı adımlarını tahmin edebildiği belirtilmektedir.

3.3. Operasyonel STİ Üzerine Çalışmalar

Operasyonel STİ'nin oluşturulmasında, taktik isthibarata görece daha fazla alan uzmanı yorumuna ihtiyaç duyulmaktadır. Zafiyet yönetiminde, olay müdahalesinde ve tehdit monitörlemede operasyonel STİ'ye sıkça başvurulmaktadır. Operasyonel STİ, kurum ve kuruluşara yönelik tehditler özelinde bilgi sağlamaktadır. Savunma tarafında kullanılmak üzere potansiyel riskleri ortaya çıkarmaktadır. Saldırganların yöntemleri hakkında daha derin bir içgörü sunmaktadır. İstihbarat üretiminde geçmişteki zararlı aktiviteleri de inceleyerek elde ettiği bilgileri kullanmaktadır.

Kurum ve kuruluşların olası tehdit aktörlerini ve bu tehdit aktörlerinin saldırı niyetlerini, kabiliyetleri ve aradıkları fırsatları, savunmasız bilgi teknolojileri varlıklarını tanımalarını sağlamaktadır. Ayrıca tehdit aktörlerinin başarılı olduğu durumda saldırının etkisinin anlaşılmasına yardımcı olmaktadır. Özellikle devletler seviyesinde operasyonel STİ oluşturulması, kamu ve özel sektördeki kritik altyapıların tehditlere karşı bilgilendirilmesi önemlidir. Çizelge 3.3'te operasyonel STİ'den yararlanarak saldırı tespiti kabiliyeti geliştiren çalışmalara yer verilmektedir.

Çizelge 3.3. Operasyonel STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
An Adaptive Analysis Framework for Correlating Cyber Security Related Data [89]	Ağ trafiği, güvenlik cihazlarının ürettiği alarm günlükleri, harici STİ beslemeleri (feeds)	Veri kaynaklarından toplanan veri üzerinde ilişkiler kurarak tehdit ifade edebilecek örüntüler keşfedilmekte-dir.	Toplanan veri üzerinde veri ilişkilendirme algoritmaları uygulanarak olaylar arasında zaman, neden, hiyerarşi ve anlamsal (semantik) ilişkiler kurulmaktadır.	Önerilen çerçeve modelin saldırı ve anomali tespiti ile ilgili ünversite ağında gerçekleştirilen deneysel çalışmanın sonuçlarına yer verilmemektedir.

Çizelge 3.3. (devam) Operasyonel STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
Collecting Cyber Threat Intelligence from Hacker Forums via a Two-Stage, Hybrid Process using Support Vector Machines and Latent Dirichlet Allocation [90]	Nulled.io forum sitesindeki başlıklar veri kümesi olarak kullanılmaktadır. 16,000 güvenlikle ilişki ve ilişkili olmayan forum başlığı üzerinde çalışılmıştır.	Forumda yer alan başkılar web kazıma yöntemleri ile toplanarak SVM ve LDA ile ikili sınıflandırma yapılmıştır.	Yazarlar 98.82% oranında doğru sınıflandırma yapmıştır. Başlıklar üzerinden sızıntı kimlik bilgileri, zararlı vekil sunucu adresleri, bilinmeyen zararlı yazılımlar ve kurum/ürün odaklı zararlı betik (script) ve hedef odaklı zafiyetlerin çıkarımı yapılabilmektedir.	Sıfır-gün saldırılarına ilişkin tehdit istihbararatı çıkarımı yapılamamıştır. Sıfır-gün etiketli başlıkların veri kümesi içerisinde çok az geçmesi sebebiyle yöntem geliştirilemediği ifade edilmektedir. Sıfır gün saldırına dair başlıklar tespit edilse dahi, zararlı yazılım dosya imzaları elde edilemediği sürece etkili bir tehidt istihbaratı elde edilemeyeceği değerlendirilmek-tedir.
Cyber-Attack Features for Detecting Cyber Threat Incidents from Online News [91]	Siber güvenlik sektöründe internet haberciliği yapılan sitelerden web kazıma ile elde edilen haber içeriği içeren veri kümesi kullanıl-maktadır.	Metin ön işleme yapılarak, haber metinleri temizlenmektedir. Siber saldırı özellikleri belirlenerek, haber içinde geçen anahtar kelimeler sınıflandırılmaktadır.	Operasyonel STİ otonom olarak elde edilmesi zor bir türdür. Yazarlar bu alanda yaptıkları çalışma sınırları içinde bir örnek çıktı sunmaktadır. Semantik olarak zarar gören kurum ile zarar veren saldırganın kimliğinin ayırt edilmesi STİ alanına olumlu bir katkı olarak değerlendirilmektedir.	GİG'lerin kapsam dışında bırakılması ve GİG'ler hakkında semantik sınıflandırma yapılmasına yönelik bir değerlendirme olmayışı sebebiyle kapsamın oldukça dar tutulduğu değerlendirilmektedir. Bir alan adı veya IP adresinin zararlı mı yoksa zarar gören tarafta mı olduğu bilgisinin haberlerden elde edilmesine dair bir değerlendirme sunulmamıştır. Doğruluk değerleri sunulmamaktadır.

Çizelge 3.3. (devam) Operasyonel STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
Discovering hidden relations between Tor Marketplaces users [92]	Tor ağındaki market sitelerinden toplanan PGP açık anahtarları veri kümesi olarak kullanılmaktadır.	Tor ağındaki marketlerden PGP açık anahtarı ve onlara ait PGP kimlikleri toplanarak, PGP açık anahtalarını paylaşarak haberleşen kullacılar (satıcı/alıcı) arasındaki ilişkiler ortaya çıkarılmaktadır.	Market kullanıcılarının kişiye özel üretmiş oldukları PGP açık anahtarlarının, kişinin tekil kimliğini ifade edebileceği fikrinin, STİ alanındaki varlıklar arasında ilişki kurmak için etkili bir yöntem olduğu değerlendirilmektedir.	Önerilen yöntemin otonom çalışması yayın yapan marketlerin sık sık IP ve sunucu değiştirmesi sebebiyle mükün değildir. Ayrıca ilgili siteler de web kazımaya karşı koruma yöntemleri geliştirmektedir. Yönetimin işlemesi için insan müdahalesi gerektiği için üretilen STİ'de kesinkiler ve boşluklar oluşacağı değerlendirilmektedir.
Creation and Management of Social Network Honeypots for Detecting Targeted Cyber Attacks [93]	Linkedin üzerinde kurulan sosyal balküpleri	Sahte Linkedin güvenlik uzmanı profilleri oluşturularak, bu profiller ile eposta ve mesajlaşma üzerinden etkileşime geçen aktörlerin ilettiği dosya ekleri ve metinler.	Sosyal mühendislik yöntemleri kullarak keşif süreci işleten APT'lerin erken tespiti için etkili bir yöntemdir.	Tehdit unsurları tespit edildikten sonra, tehdit aktörüne ait farklı teknik ve taktik STİ'lerin otonom elde edilmesine yönelik bir yöntem sunulmamaktadır.

Siber tehditlerin tespit edilmesi üzerine yapılan çalışmaların ortak olarak karşılaştığı çeşitli durumlar bulunmaktadır. Tehditlerin tespit edilmesi için başvurulan veri kaynaklarına bakıldığında ağ trafiği, harici istihbarat kaynakları, zararlı URL'ler, zararlı alan adları, zararlı dosya imzaları gibi birçok farklı nitelikteki veri üzerinde çalışılması gerekmektedir. STİ alanındaki bu veri hızla büyümektedir. Tehdit istihbaratı oluşumunu destekleyebilecek potansiyel veri kaynaklarını kapsama alma kararı verinin çeşitliğini hızla artırmaktadır. Üzerinde çalışılan verinin tehdit istihbaratı üretilmesine katkı sağlama potansiyeli de keşfedilmesi gereken diğer bir unsurdur. STİ üretimi için üzerinde çalışılan veri, büyük

veri kavramı ile açıklanabilecek niteliktedir. Tez kapsamında, STİ üretimi ve zenginleştirilmesinde önemli görülen, elverişli veriye bağlamsallaştırma kapasitesi yüksek veri denilmektedir.

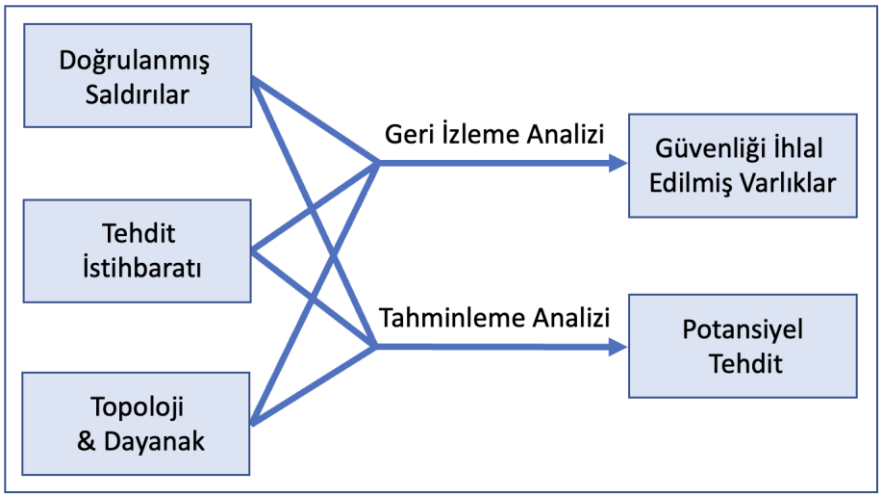
Jin ve diğerleri siber güvenlik ile ilgili veriler üzerinde ilişkiler kurmak üzere bir çerçeve model oluşturmuştur. Araştırmacılar siber güvenlik alanından gelen veri üzerinde ilişkiler kurarak ve tehdit ifade edebilecek örüntüler keşfederek potansiyel siber tehditleri ortaya çıkarmayı hedeflemektedir. Önerilen çerçeve modelde temel olarak 5 ana bileşen bulunmaktadır. Bunlar veri toplama, toplanan veriyi çevrimiçi ve çevrimdışı olarak işleme, veri yönetimi ve veri depolama görevlerini yerine getirmektedir. Veri yönetiminden sorumlu bileşen, verinin kaydedilmesi, indekslenmesi, sorgulanması ve erişim kontrolü yapılmasından sorumludur. Veri depolama bileşeni ise ilişkisel ve NoSQL veri tabanları, Elasticsearch ve Hadoop dağıtılmış dosya sistemi (Hadoop distributed file system, HDFS) veri depolama teknolojilerini yönetilmesini sağlamaktadır [89].

Veri toplama bileşeni, ağdaki güvenlik ile ilişkili donanım ve yazılımların ürettiği trafik verisi ve olay günlüklerini toplamaktadır. Toplanan veri içinden alarm olayları tespit edilerek bir veri kümesi meydana getirilmektedir (Şekil 3.4). Alarm olayları çevrimiçi hesaplamalar için temel girdiyi oluşturmaktadır. Araştırmacılar elde ettikleri veri üzerinde veri ilişkilendirme algoritmaları uygulayarak olaylar arasında zaman, neden, hiyerarşi ve anlamsal (semantik) ilişkiler kurmaktadır.



Şekil 3.4. Çevrim içi hesaplama fazları [89]

Çevrimdışı hesaplama sürecinde ise bilindik saldırı olayları, dış kaynaklardan toplanan tehdit istihbaratı ve ağ topolojisindeki varlıkların bilgileri üzerinden tahminleme analizi (prediction analysis) ve geri izleme analizi (back tracking analysis) gerçekleştirmektedir (Şekil 3.5).



Şekil 3.5. Çevrim dışı hesaplama fazları [89]

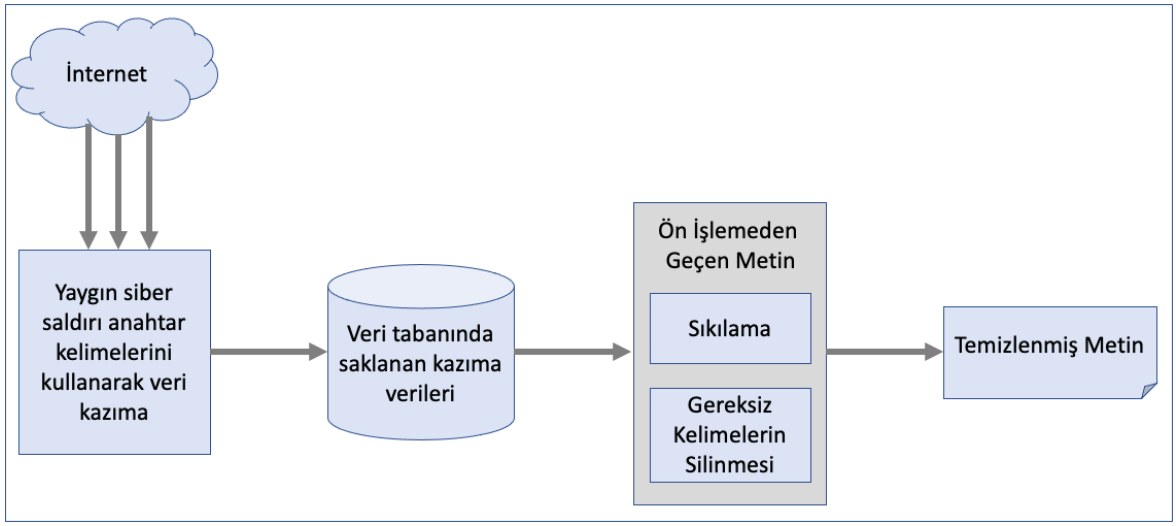
Araştırmacılar geliştirdikleri çerçeve modeli işletmek için üniversite ağındaki trafik verisini, üniversite ağındaki güvenlik cihazlarının ürettiği günlük kayıtlarını ve VirusTotal ile PhishTank gibi açık tehdit istihbaratı kaynaklarından veri toplamaktadır. Çalışmada önerilen çerçeve model, iletim kontrol protokolü (transmission control protocol, TCP) akışı yeniden birleştirme, saldırı tespiti ve trafik anomali analizi yapma kabiliyetlerine sahiptir.

STİ üretimi sürecinin manuel olarak işletilebildiği ve yüksek kalitede istihbaratın elde edildiği durumların başında saldırılar sonrasında yapılan incelemeler ve olay raporu çalışmaları gelmektedir. Bu olay raporlarının sonucunda, teknik, taktik, operasyonel ve stratejik STİ elde edilebilmektedir. Diğer bir etkili manuel süreç ise zararlı yazılımların analiz edilmesi sonucunda elde edilen bulguların STİ'ye dönüştürülmesidir. STİ çözümleri üreten firmaların çalışmaları incelendiğinde bu tür faaliyetlerin de otonom çalışabilen sistemler ile yerine getirilmesi için yürütülen araştırmalar ile karşılaşılmaktadır [94]. Kesintisiz bir STİ üretim süreci elde edilmesi için otonom davranan akıllı sistemlerin geliştirilmesi ve sürece etkili bir şekilde entegre edilmesi gerekmektedir. STİ çıkarımı yapılabilecek değerli bir diğer kaynak ise tehdit aktörleri ve saldırganların yer aldığı forumlardır. Aynı şekilde tehdit aktörlerinin kullandıkları forumların da incelenmesi, analiz edilmesi, yorumlanması ve STİ üretimi yapılması gerekmektedir.

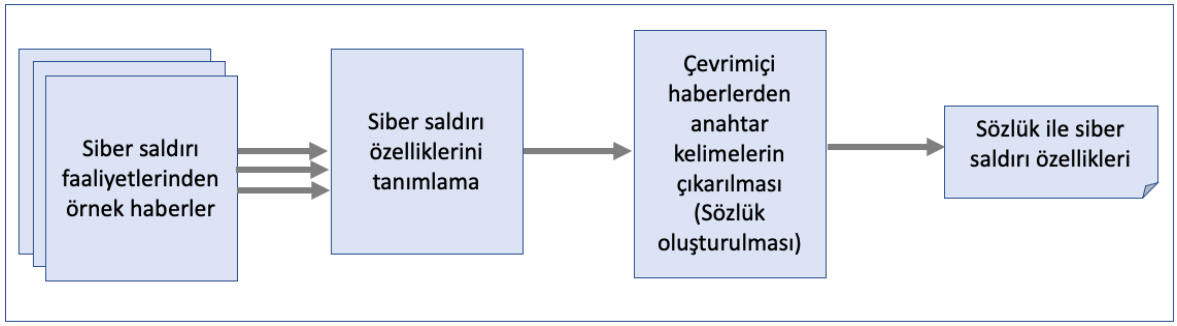
Bu konuda Deliu ve diğerleri destek vektör makineleri (support vector machines, SVM) ve gizli Dirichlet tahsisi (latent Dirichlet allocation, LDA) kullanarak iki aşamalı, hibrit bir süreçle nulled.io forumundan STİ toplamayı hedefleyen bir çalışma yapmıştır [90]. SVM

kullanılarak forumlardaki başlıkların STİ üretimine katkı saylayacak bilgi içerip içermediği belirlenmekte ve LDA sayesinde ise bu başlıkların ilişkili oldukları konulara göre kümelenmesi sağlanmaktadır. Araştırmacılar 10 katlı çapraz doğrulama (k-fold cross-validation) uygulayarak SVM algoritmasını 16,000 adet forum başlığının olduğu veri kümesi üzerinde test etmiştir. Eğitim veri setinde ise forum konu başlıkları, ilgili ve ilgisiz olarak 2 kategoride etiketlenmiştir. 16,000 verinin bulunduğu kümede algoritma 98.82% doğruluk ile sınıflandırma yapmıştır. Operasyonel STİ üretiminde, tehdit aktörünün kimliği, araçları, TTP'leri, hedef aldığı zafiyetler gibi tehdit istihbaratı alanından bilgilerin edinilmesi açısından derin İnternet forumları zengin bir kaynak olarak değerlendirilmektedir.

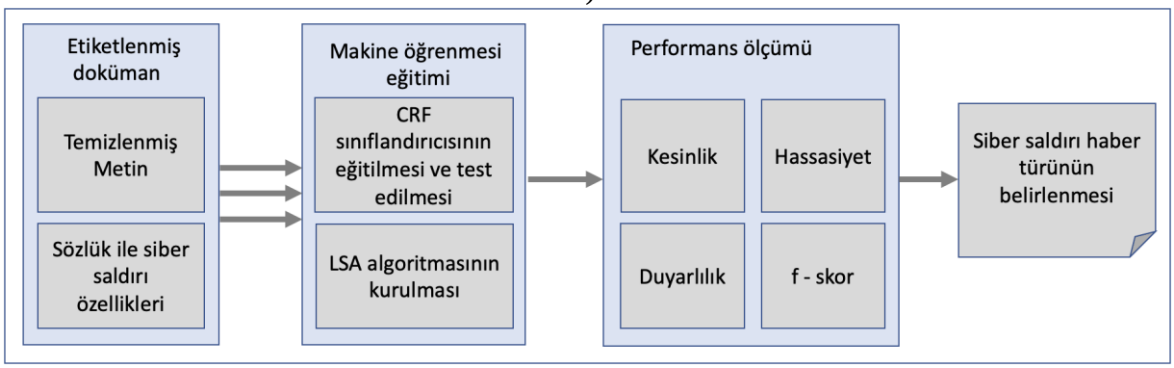
Operasyonel STİ edinmek için önemli bir diğer kaynak da siber saldırılar ile yapılan İnternet haberleridir. Güncel saldırılar ile ilgili güvenlik uzmanlarının ve teknoloji yazarlarının yaptığı haberler tehditlerin farkına varmak için bir fırsat oluşturmaktadır. Siber güvenlik alanında yazılan blog yazıları ve haberler çoğunlukla güncelliğini yitirmiş GİG'ler içerse de TTP'ler ve hedef alınan zafiyetler geçerliliğini korumaktadır. Zainal ve diğerleri internet haberlerinden STİ çıkarımı yapmak için bir model önermektedir [91]. Önerdikleri yöntemde koşullu rastgele alan (conditional random field, CRF) sınıflandırıcı ve gizli semantik analiz (latent semantic analysis, LSA) kullanarak siber saldırıları ifade eden haberler ve saldırıya dair detay bilgiler belirlenmeye çalışılmaktadır. Geliştirilen sistemin operasyonel STİ üretimini gösteren işlem basamakları Şekil 3.6(a), Şekil 3.6(b) ve Şekil 3.6(c)'de sunulmaktadır.



a)



b)



c)

Şekil 3.6. a) Metin ön işleme b) Siber saldırı özelliklerinin tanımlanması c) CRF ve LSA kullanarak internet haberi sınıflandırma [91]

Yazarlar önerdikleri model ile belirledikleri özelliklere karşılık Çizelge 3.4'teki değerleri tespit etmişlerdir.

Çizelge 3.4. İnternet haberlerinden elde edilen özellik ve değer örnekleri

Siber saldırı özelliği	Terimler
Tehdit türü	Zararlı yazılım, Botnet, Fidye yazılımı, Oltalama, Trojan, Arka kapı, Casus yazılım, Ayrıcalık yükseltme saldırısı, Solucan
Tehdit adı	Reaper, Satori, Wannacry, Loapi, Notpetya, Andromeda
Tehdit aktörü	The Shadow Brokers, Anonymous, Nexus Zeta, North Korea, Carbanak Gang, Lazarus Group
Etkilenen Organizasyonlar	Google, Microsoft, Apple
Etkilenen Platformlar	Router, iOS, Windows, Play Store, Cryptocurency
Etkilenen Ülkeler	South Korea, Brazil

APT saldırıları saldırganın motivasyonu sebebiyle, planlı, karmaşık yöntemler ile kurgulanmış ve hedef odaklı saldırılardır. Tehdit aktörü, saldırı öncesinde keşif aşaması adı verilen araştırma süreci geçirmektedir. Keşif aşaması dahilinde hedefin bilişim varlıkları ve altyapısı hakkında bilgiler de toplanabilir. Aynı zamanda sosyal mühendislik adı verilen yöntemler kullanılarak kurum çalışanları, kurumun iş ortakları ve müşterilerinin çalışanları hakkında OSINT ve insan istihbaratı (human intelligence, HUMINT) toplanabilmektedir. Sosyal mühendislik yöntemlerinin ne kadar geniş bir sosyal ağa yayılacağı belirsizdir ve saldırgan açısından etik bir sınır yoktur. Burada önemli olan hedefin tanınması, zafiyetlerinin belirlenmesi bunların sömürülmesi ve bunları yaparken güvensiz hissetmesini engelleyici seviyede inandırıcılık sergilenmesidir. Örneğin, e-posta, sahte alan adı ve mobil uygulama ile oltalama (phising) saldırısı yapmak, sahte telefon aramaları, gibi yöntemler ile hedefe yaklaşmak için fırsatlar oluşturulabilmektedir.

Paradise ve diğerleri yaptıkları çalışmada hedefli siber saldırıların tespiti için sosyal ağ bal küplerinin (honeypots) oluşturulması ve yönetimini konu alan bir çalışma yapmıştır [93]. Çalışmada incelendiği üzere, bal küpü profilleri, sosyal ağa başarılı bir şekilde asimile edilmekte ve olası bir saldırının temel belirtilerini ortaya çıkaran şüpheli arkadaşlık istekleri ve e-posta mesajları almaktadır. Ayrıca, sosyal ağlarda bulunan çalışanların davranışları ve sosyal ağlar üzerinden gelen sızma girişimlerine karşı dayanıklılıkları araştırılmıştır. İncelen çalışma üç önemli katkısından dolayı tez kapsamına alınmıştır. Bunlardan birincisi, APT saldırılarının keşif gibi erken bir fazda farkına varılması için bir sosyal bal küpü önerilmektedir. Sosyal bal küplerinden toplanan veriler, STİ zenginleştirmesi için değerli görülmektedir. Özellikle tehdit aktörlerinin tespit edilmesi için sosyal bal küpü ziyaretçilerinin kayıt alınması dikkate değerdir. Çalışmada, bal küpü

olarak oluşturulan profil, kurumsal organizasyonun siber güvenlik uzmanıymış gibi ağda asimile edilmektedir.

Çalışmanın sunduğu ikinci katkı, bal küplerinin verimli bir şekilde oluşturulması ve bakımı için bir çerçeve model sunulmaktadır. Bu modelin içeriğinde sosyal ağa erişimin elde edilmesi, mevcut profillerin toplanması, yapay profillerin oluşturulması ve yönetilmesi, oluşturulan bu profillerin ilgili sosyal ağa aktarılması, son olarak oluşturulan profillerin ve ilişkili oldukları e-posta hesaplarının izlenmesi bulunmaktadır. Çalışmanın üçüncü katkısı ise önerilen modelin, geçici bir süreyle gerçek hayatta uygulanmış olmasıdır. Bu süre zarfı içinde toplanan veriler yorumlanmıştır. Çalışmada kullanılan sosyal ağ Linkedin olarak belirlenmiştir.

Linkedin'deki profillere ait bilgiler toplandıkça çeşitli yorumlar yapılmıştır. Bunlardan birincisi, bir kullanıcının ne kadar çok arkadaşı varsa, arkadaşlık isteklerini onaylama eğiliminin o kadar yüksek olduğudur. İkincisi ise bir kişinin, kendisine bağlantı isteği gönderen kişi ile ne kadar çok ortak arkadaşı varsa, bu isteği onaylama ihtimale o kadar artmaktadır. Çeşitli kurumlarda siber güvenlik uzmanı olarak çalışıyor gözüken sahte profillere gelen e-posta sayıları dikkat çekici bulunmuştur. Çalışmada 6 adet profil oluşturulmuştur ve bunlardan üçü 6 ay boyunca, birisi 4 ay boyunca, ikisi ise 2 ay boyunca aktif olarak yönetilmiştir. Oluşturulan profiller sırasıyla, 430, 4718, 250, 2819, 270 ve 414 adet e-posta almıştır. Profillerin aldıkları e-postalar, VirusTotal platformu üzerinde sorgulandığında sırasıyla, 2, 3, 1, 1, 8 ve 6 tanesi şüpheli olarak gözlemlenmiştir.

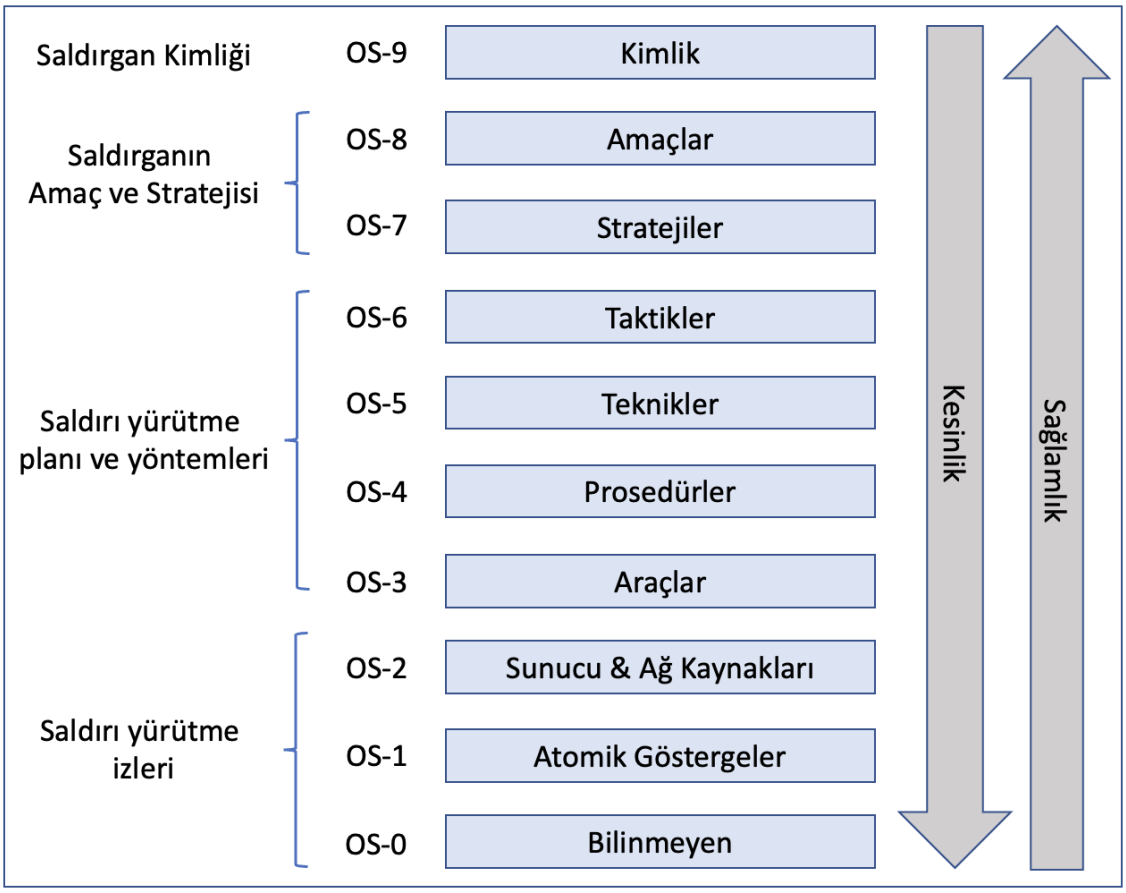
3.4. Stratejik Siber Tehdit İstihbaratı Üzerine Çalışmalar

Stratejik STİ, iş kararlarını ve bunların arkasındaki süreçleri bilgilendirmektedir. Tehdit aktörleri yanlış anlaşıldığı ve yorumlandığı zaman iş kararlarının alınmasında hatalar yapılmaktadır. Stratejik STİ, küresel olayların, dış politikanın ve uluslararası etkileşimin organizasyonların siber güvenliğini potansiyel olarak nasıl etkileyebileceğini göstermektedir. Bu sayede siber güvenlik alanındaki yatırımlar şekillenmektedir. Bu seviyedeki istihbaratın üretilmesi diğer seviyelere görece daha zordur ve dünyadaki jeopolitik gelişmelerin de yorumlanmasını gerektirmektedir. Çizelge 3.5'te stratejik STİ'den yararlanarak saldırı tespiti kabiliyeti geliştiren çalışmalara yer verilmektedir.

Çizelge 3.5. Stratejik STİ alanındaki çalışmalar

Çalışma	Tehdit İstihbaratı Kaynağı	Yöntem	Güçlü Yönler	Zayıf Yönler
Analysis of Malware Communities Using Multi-Modal Features [95]	Zararlı yazılım veri kümesi	Dukes, APT-1 ve Deep Panda APT gruplarının ürettiği düşünülen zararlı yazılımlar üzerinden multimodel özellik çıkarımı yaparak, denetimsiz öğrenme ile zararlı yazılımların ilgili APT gruplarına ait olma durumu değerlendi-rilmektedir.	Zararlı yazılımların içinden IP adresi, alan adı, URL gibi bilindik GİG'lerin çıkarımını yapmak yerine ve makine öğrenmesine girdi sağlayabilecek multimodal özellik gruplarının tespiti yapılmıştır. Bu da kural tabanlı çalışmalardan ayrışarak, byte entropisi, zararlıda kullanılan kütüphaneler, string ifadelerin imza değerleri (hash) üzerinden zararlılar arasında benzerlik hesaplamayı mümkün kılmıştır.	Veri seti olarak kullanılan zararlı yazılım ve geliştiricisi olduğu tahmin edilen APT grubu eşleşmesi hatalı olabileceği değerlendirilmek-tedir. Temel alınan veri kümesini oluşturulmasında ki kıstaslar da veri kümesini oluşturan diğer araştımacıların varsayımları üzerine kuruludur.

STİ üretim sürecinin esas amacı siber tehditlerin ortaya çıkarılmasıdır. Tehditlerin ortaya çıkarılmasında farklı derinliklerde bilgi elde edilmektedir. Tehdit unsurlarına ilişkin ipuçları elde edilerek olabildiğince net bir tehdit tanımlaması yapılması hedeflenmektedir. Bu konuda literatürde karşılaşılan en olgun siber tehdit tespiti olgunluğu seviyelendirme çalışması Bromander ve diğerleri tarafından yapılmıştır [96]. Önerdikleri olgunluk seviyelendirme modeline Şekil 3.7'de yer verilmektedir.



Şekil 3.7. Siber tehdit olgunluk seviyesi modeli [96]

Şekil 3.7'de görüldüğü üzere giriş Olgunluk Seviyesindeki (OS) tehdite dair hiçbir bilgiye sahip olunmamaktadır. OS arttıkça, tehdite dair atomik göstergeler elde edilmiş, sunucular ve ağdaki diğer varlıklar hakkında bilgi sahibi olunmuş, tehditin kullandığı araçlar belirlenmiş, saldırganın TTP'leri hakkında bilgi edinmiş, en üst seviyelerde ise saldırganın stratejisi ve amacına yönelik çıkarımlar yapılmış, nihai olarak ise tehdit aktörünün kimliği belirlenmiş olmaktadır.

STİ üretim sürecindeki her bir basamağın hizmet ettiği ortak amaç siber tehditin belirlenmesidir. Tehditlerin belirlenmesi için çok önemli kaynaklardan birisi ise zararlı yazılımlardır. Zararlı yazılımlar bir saldırı vektörünün kendisi olabileceği gibi saldırı vektörünün bir parçası olarak da ortaya çıkabilmektedir. Zararlı yazılımlar doğrudan saldırgan veya saldırganın amacına hizmet etme motivasyonu kazanmış taraflarca geliştirildiği için, tehditi ve tehdit aktörünü belirlemek için oldukça somut bir fırsat sunmaktadır. Bir programın zararlı yazılım olarak kabul edilmesi için gerçekleştirilmesi gereken oldukça karmaşık analiz yöntemleri bulunmaktadır. Siber güvenliğin pek çok

alanında olduğu gibi zararlı yazalım analizi alanında da deterministik (kural tabanlı) [97-101] ve gerekçeye dayalı yöntemler olduğu gibi makine öğrenmesine dayalı yöntemler de bulunmaktadır [102-106]. Zararlı yazılımların analizine ilişkin yöntemler tez kapsamı dışında bırakılarak, STİ'nin oluşumu ve zenginleştirilmesinde zararlı yazılımların analiz edilmesi ile elde edilebilecek verilerin neler olduğu ve bunlardan nasıl yararlanılabileceği üzerinde durulmaktadır.

Cruickshank ve diğerlerinin yaptığı çalışmada multimodal özellikler kullanılarak zararlı yazılım analizi yapılmaktadır. Multimodal, iki veya daha fazla modun kombinasyonları yoluyla, iki veya ikiden fazla iletişim moduna sahip olunduğu anlamına gelmektedir. Makalede ele alınan modlar arasında yazılı dil, konuşma dili, görsel, işitsel, jest, dokunsal ve uzamsal anlam kalıpları bulunmaktadır [107]. Araştırmacılar Dukes, APT-1 ve Deep Panda olarak bilinen üç ayrı APT grubuna ilişkin zararlı yazılımlardan oluşan bir veri kümesi üzerinde çalışmaktadır. Çalışmada, zararlılar denetimsiz öğrenme ile kümelenerek hangi tehdit grubuna ait olduğu ortaya çıkarılmaktadır. Ayrıca multimodal yaklaşımın orta veya geç entegrasyonunun, erken entegrasyonuna göre daha iyi doğruluk verdiği sonucuna ulaşılmaktadır.

Araştırmada üç aşamalı bir yöntem kullanılmaktadır. Birinci adımda statik analiz yapılarak zararlı yazılımlara ait farklı özellikleri ortaya çıkarılması sağlanmaktadır. İkinci adımda, veri modlarının her biri için küme atamaları ve en yakın komşu çizgelerinin oluşturulması bulunmaktadır. Üçüncü aşamada ise, her zararlı yazılım için küme atamalarını güncellemek için bir çapraz yayılma ve etkileme süreci kullanılarak nihai küme atamaları yapılmaktadır [95]. Araştırmacılar zararlı yazılımlardan elde ettiği özellikler; byte entropisi (byte entropy), programa dahil edilen kütüphaneler (import address table), string imzaları (string hashes) ve taşınabilir programların başlıklarıdır (portable executable headers). Bu araştırmayı farklı kılan yönlerden biri zararlı yazılımların içinden IP adresi, alan adı, URL gibi bilindik GİG'lerin çıkarımını yapmamaları ve makine öğrenmesine girdi sağlayabilecek multimodal özellik gruplarını elde etmeleridir.

Siber uzayda gerçekleşen saldırılar incelendiğinde fidye amaçlı ve botnet oluşturmaya yönelik zararlı yazılımlar önemli bir kısmı oluşturmaktadır. Özellikle 2016 yılında başlamış olan bir fidye zararlı yazılımı trendi meydana gelmiştir. Etki alanı ve oluşturduğu maddi kayıpları yüksek fidye zararlılarına bakıldığında karşımıza WannaCry (2017) [108],

Locky (2016) [109], Cerber (2016) [110], Jigsaw (2016) [111], CrySiS (2016) [112], Torrent Locker (2014) [113], CryptoLocker (2013) [114], MBRBlock (2016), Petya (2015) [115] ve Bad Rabbit (2017) [116] çıkmaktadır. Fidye yazılımlarına karşı hızlı ve otomatik tespit ile engelleyici önlemler almak için teknik STİ edinilmesi gerekmektedir. Fakat bu zararlılara karşı nasıl ve neden önlem alınması gerektiğine dair bilgi, stratejik STİ kaynaklarından elde edilen trend analizleri ve iş katmanında oluşabilecek maddi ve itibari kayıplara yönelik yapılan risk analizleri sonucunda elde edilmektedir [117,118].

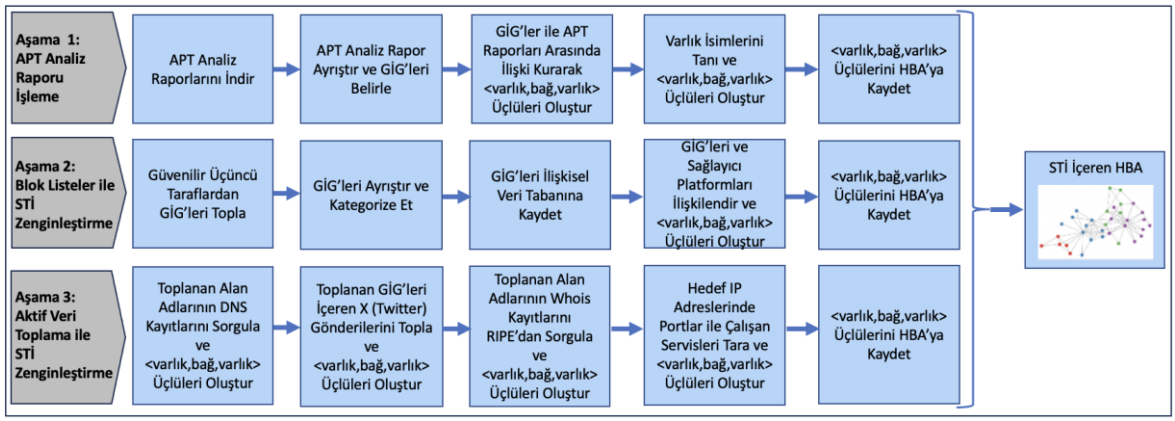
4. ZENGİNLEŞTİRİLMİŞ STİ İÇEREN HBA ÜRETİMİNE YÖNELİK ÇERÇEVE MODELİ ÖNERİSİ

Bu bölümde, tez çalışmasında önerilen zenginleştirilmiş STİ içeren HBA üretimine yönelik çok aşamalı çerçeve modeli anlatılmaktadır. Önerilen çerçeve model üç aşamadan meydana gelmektedir. Her bir aşamada gerçekleştirelen işlemlerin dayandığı gereksinimler aşağıda maddeler halinde sunulmaktadır. İlgili aşamalar, bu bölümün alt başlıklarında detaylı bir şekilde anlatılmaktadır. Şekil 4.1'de HBA üretimi süreci akış diyagramı halinde sunulmaktadır.

Aşama 1: Temel HBA'nın oluşması için APT analiz raporlarının elde edilmesi gerekmektedir. APT analiz raporlarının içinden bağlamsallığı yüksek olduğu düşünülen, literatür taramasında karşılaştığımız ve tez çalışmasında yararlı görülen GİG'lerin elde edilmesi sağlanmaktadır.

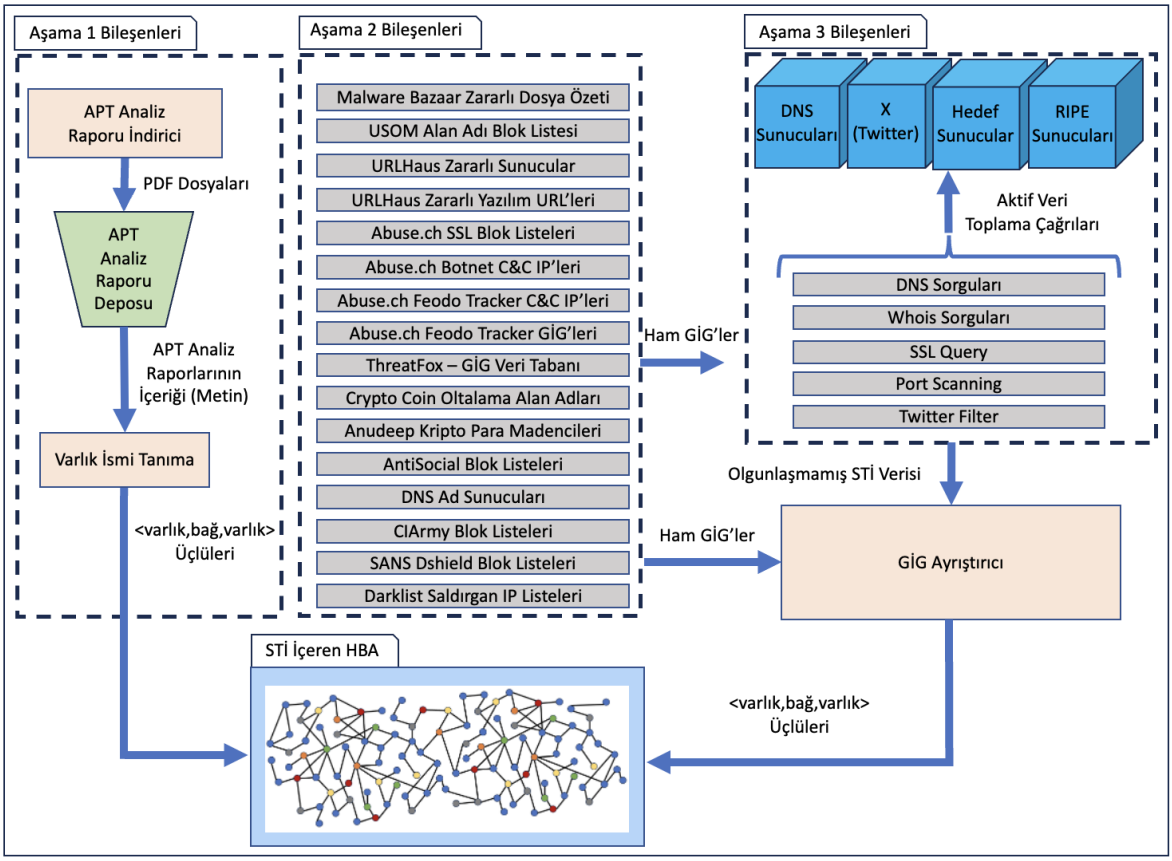
Aşama 2: Oluşturulan HBA'nın zengileştirilmesi için GİG'lerden oluşan blok listelerin açık kaynaklardan toplanması, ayrıştırılması ve mevcut HBA'ya entegre edilmesi sağlanmaktadır.

Aşama 3: Elde edilen HBA'nın daha yüksek bağlamsallığa ulaşması için aktif veri toplama yöntemleri uygulanmalıdır. Aktif veri toplama işlemlerinden elde edilen verinin HBA'ya entegre edilmesi sağlanmaktadır.



Şekil 4.1. Toplanan STİ'den HBA üretilmesi ve zenginleştirilmesi

Bu bölümde tarif edilen işlem basamakları ve yöntemler uygulandığında nihai olarak çizge veri yapısıyla ifade edilebilen zenginleştirilmiş STİ'den meydana gelen bir HBA elde edilmektedir. HBA oluşturmak için oluşturulan sistemin mimari tasarımı Şekil 4.2'de sunulmaktadır.



Şekil 4.2 STİ toplama, zenginleştirme, HBA üretimi ve bilişenler arasındaki veri akışına ilişkin üst düzey sistem mimarisi

Aşama 1 bileşenleri, APT analiz raporlarının indirilmesini, raporların dosya servisine kaydedilmesini, raporlar üzerinde NER faaliyetlerinin yürütülmesinden ve ortaya çıkarılan GİG'leri <varlık, ilişki, varlık> üçlülerine dönüştürerek HBA'ya kaydetmekten sorumlu uygulamalar ve yazılım bileşenlerinden meydana gelmektedir.

Aşama 2 bileşenleri, APT analiz raporlarından elde edilen STİ'nin zenginleştirilmesinden sorumludur. Blok listeleri, web kazıma ve dosya halinde indirme yöntemleriyle elde edilmektedir. Elde edilen listeler ayrıştırılarak, GİG'ler kategorize edilmektedir. Kategorize edilen ve tanınan GİG'ler <varlık, ilişki, varlık> üçlülerine dönüştürerek HBA'ya kaydetmekten sorumlu programlardan meydana gelmektedir.

Aşama 3 bileşenleri, ilk iki aşamadan elde edilen GİG'leri kullanarak aktif veri toplama işlemleri yapan, mevcut STİ'yi zenginleştirmekten sorumlu uygulamalardan meydana gelmektedir. Alan adları (domain) ve alt alan adlarının çözüldüğü IP adresleri, alan adları ve IP adreslerinin Whois sorgu sonuçları, yayın yapan adreslerin (domain, subdomain, URL) varsa SSL sertifikaları, aktif olan uzak makinelerin varsa hizmet veren açık portlarından toplanan veri ve sahip olunan GİG'lere ilişkin X (Twitter) gönderilerini toplamaktan sorumlu uygulamalar bu aşamada kullanılmaktadır. Bu uygulamalar da <varlık, ilişki, varlık> üçlüleri oluşturarak HBA'ya entegre etmekten sorumludur. Şekil 4.2'de önerilen mimaride yer alan yazılım bileşenleri, kullanılan APT analiz raporları, ham veri kümeleri ve oluşturulan çizge veri tabanı https://github.com/burakgulbay/apt-scope adresinde sunulmuştur.

4.1. APT Analiz Raporlarının İşlenmesi

Birinci aşamada APT analiz raporlarının işlenmesine dair işlemler yerine getirilmektedir. Bu aşamada, öncelikli olarak APT raporları elde edilmektedir. https://github.com/aptnotes/data/blob/master/APTnotes.json adresinde aktif olarak sürdürülen geniş bir APT saldırı raporu veri kümesine ait meta bilgiler yer almaktadır. Her bir rapor için dosyanın yüklendiği dosya paylaşım sitesini gösteren bağlantı adresi bulunmaktadır. Dosyaların otomatik olarak indirilmesi için bir dosya indirme uygulaması geliştirilmiştir. Geliştirilen uygulama, ilgili adrese erişerek dosyaları dosya sistemine kaydetmektedir.

Kaydedilen dosyalar bir GİG ayrıştırma uygulaması ile işlenerek sahip oldukları GİG'ler elde edilmektedir. Literatürde bahsedilen ve açık kaynaklarda karşılaşılan birçok GİG ayrıştırma uygulaması incelenmiştir. Bunlardan, Caballero ve diğerlerinin geliştirdiği uygulama alternatiflere göre yüksek başarı ile GİG'leri doğru tespit etmektedir [119]. Kullanılan GİG ayrıştırıcının elde ettiği veri türleri aşağıda sunulmaktadır:

- URL
- Alan adı (domain)
- Alt alan adı (subdomain)
- Web adresi (url)
- IP adresi (ip)

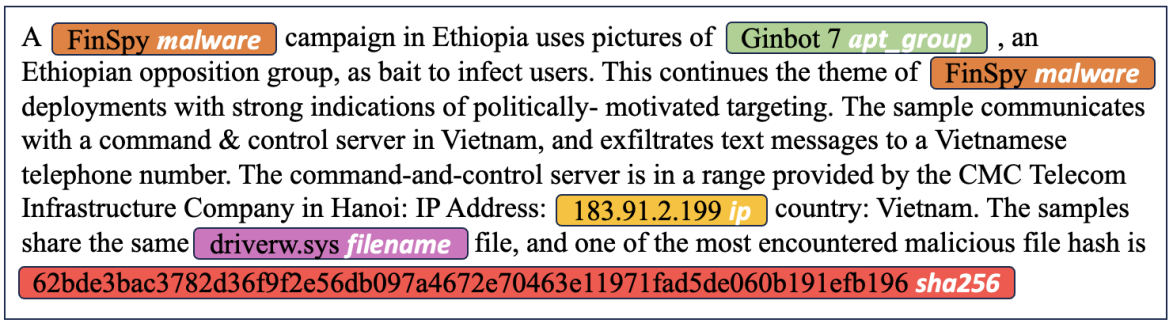
- IP alt ağı (ip4net)
- Dosya özeti (md5, sha1, sha256)
- E-mail adresi (email)
- UUID (uuid)
- Telefon numarası (phone)
- Sosyal medya hesapları (twitter_handle, facebook_handle, linkedin_handle, youtube_channel, instagram_handle)
- Dosya adı (filename)

Caballero ve diğerlerinin geliştirdiği GİG ayrıştırıcının (extractor) bulamadığı fakat, APT analiz raporlarında sıkça rastlanan diğer GİG'ler ise zararlı yazılım adları (malware), MITRE saldırı tekniği (mitre_technique), CVE kodu (cve) ve APT grubu adıdır (apt_group). Bu dört GİG türünden MITRE saldırı tekniği (mitre_technique) ve CVE (cve) kodu kurallı ifadeler (regular expressions, RegEx) eşleştirmesi yaparak tespit edilmiştir. Bu iki türe ait metin değerlerini eşleştirmek için özel kurallı ifadeler yazılarak kullanılan açık kaynaklı ayrıştırıcıya eklenmiştir. Yukarıda listelenen GİG türlerine özgü yazılmış kurallı ifadeler kullanılarak <varlık, ilişki, varlık> üçlüleri oluşturulmuş ve çizge veritabanına kaydedilmiştir. Bu bölümde anlatılan APT analiz raporlarının işlenmesi basamağında aşağıda listelenen üçlüler oluşmaktadır.

- <apt_report,MENTIONS,md5>
- <apt_report,MENTIONS,phone>
- <apt_report,MENTIONS,url>
- <apt_report,MENTIONS,uuid>
- <apt_report,MENTIONS,filename>
- <apt_report,MENTIONS,subdomain>
- <apt_report,MENTIONS,domain>
- <apt_report,MENTIONS,twitter_handle>
- <apt_report,MENTIONS,country>
- <apt_report,MENTIONS,ip>
- <apt_report,MENTIONS,sha1>
- <apt_report,MENTIONS,industry>
- <apt_report,MENTIONS,email>
- <apt_report,MENTIONS,sha256>

- <apt_report,MENTIONS,cve>
- <apt_report,MENTIONS,ip4net>
- <apt_report,MENTIONS,mitre_technique>
- <apt_report,MENTIONS,facebook_handle>
- <apt_report,MENTIONS,linkedin_handle>
- <apt_report,MENTIONS,youtube_channel>
- <apt_report,MENTIONS,malware>
- <apt_report,MENTIONS,youtube_handle>
- <apt_report,MENTIONS,package_name>
- <apt_report,HAS_APT_GROUP,apt_group>
- <apt_group,ALSO_KNOWN_AS,apt_group>

APT analiz raporlarından, zararlı yazılım adları ve APT grup adlarını elde etmek için bir varlık ismi tanıma (named entity recognition, NER) bileşeni kullanılmaktadır. Adlandırılmış varlık tanıma hedefine ulaşmak için, doğal dil işleme (natural language processing, NLP) modellerinin, manuel olarak etiketlenmiş metinlerden oluşan veri kümeleri üzerinde eğitilmesi gerekmektedir. Konum, ülke, hukuk ve tarih gibi birçok varlık kategorisine ilişkin çeşitli alandan bağımsız metin veri kümeleri üzerinde eğitilmiş genel amaçlı varlık ismi tanıyıcılar mevcut olsa da STİ ve APT konularında yetersiz oldukları görülmektedir [120]. Bunun nedeni, zararlı yazılım adları ve APT grubu adları gibi siber tehdit istihbaratına özgü varlıklar ile eğitilmemiş olmalarıdır. Bu iki varlık, tezdeki hedeflere ulaşmak için ihtiyaç duyulan, yüksek bağlamsallığa sahip HBA'nın oluşturulmasında çok önemli bir rol oynamaktadır.



Şekil 4.3. Uygulanan NER işleminin sözde çıktısı

Şekil 4.3’te, uygulanan NER işleminin sözde çıktısı (pseudo output) sunulmaktadır. 734 adet APT analiz raporu PDF formatında toplanmıştır. Tanınması hedeflenen “apt_group” ve “malware” isimleri uzman görüşüyle etiketlenmiştir. AhnLab, Checkpoint, CrowdStrike, Dragos, ESET gibi siber güvenlik şirketleri tarafından yayınlanan raporların %30'u kesin referans (ground truth) olarak seçilmiştir. Etiketlenmiş varlık türleri, manuel etiketlenenlerin sayıları ve NER uygulanarak raporlardan tespit edilen varlık sayılar ile manuel etiketlenenlerin toplamı Çizelge 4.1'de verilmiştir.

Çalışmamızda, varlık ismi tanımada SpaCy adlı önceden eğitilmiş ve uyarlanabilir bir NLP modeli kullanılmıştır [121]. Alam ve diğerleri siber güvenlik alanındaki adlandırılmış varlıkları tanımak için tasarlanmış bir Python kütüphanesi olan CyNER'i önermektedir. Siber güvenlik alanına özgü varlıklar için siber güvenlik külliyatı üzerinde eğitilmiş transformatör modellerini, GİG’leri tanımlamak için kurallı ifade eşleştirmeyi ve genel varlık türlerini tanımak için Flair ve SpaCy'den alınan NER modellerini entegre etmektedir. NER işlemi, siber güvenlik varlıklarının tanınmasının kesinliğini artırmak için önceliğe dayalı bir birleştirme tekniğini içermektedir [122]. apt_group ve malware isimleri hem SpaCy'nin en_core_web_md hem de CyNER modelleri kullanılarak tanınmaktadır. Tez çalışmasında NER’in başarısı en_core_web_md ve CyNER'in kabiliyetleri ile sınırlıdır.

Çizelge 4.1. APT analiz raporlarında etiketlenmiş ve toplam varlık sayıları

Etiketlenmiş Varlık	Etiketlenmiş Varlık Sayısı	Toplam Varlık Sayısı
apt_group	105	557
malware	196	972

NER işlemleri sonucunda da <apt_report, HAS_APT_GROUP, apt_group> ve <apt_group, ALSO_KNOWN_AS, apt_group> üçlüleri oluşturularak çizge veritabanına kaydedilmektedir. Çizelge 4.2’de APT analiz raporlarının RegEx ve NER yöntemleriyle ayrıştırılması sonucunda ortaya çıkarılan düğüm türleri ve sayıları sunulmaktadır.

Çizelge 4.2. APT analiz raporlarından elde edilen düğüm türleri ve sayıları

Düğüm Türü	Düğüm Sayısı
apt_group	557
apt_report	734
country	164
cve	271
domain	8992
email	991
facebook_handle	27
filename	2592
industry	124
ip	6431
ip4net	123
linkedin_handle	8
malware	972
md5	11159
mitre_technique	183
package_name	3
phone	142
sha1	3355
sha256	4626
subdomain	7758
twitter_handle	76
url	8031
uuid	170
youtube_channel	2
youtube_handle	5

Oluşturulan HBA'da toplamda 57,496 adet düğüm ve 68,873 adet ilişki elde edilmiştir. APT analiz raporlarının incelenmesi sonucunda elde edilen ilişki türleri ve sayıları ise Çizelge 4.3'te sunulmaktadır.

Çizelge 4.3. APT analiz raporlarından elde edilen ilişki türleri ve sayıları

İlişki Türü	İlişki Sayısı
ALSO_KNOWN_AS	479
HAS_APT_GROUP	1108
MENTIONS	67286

APT analiz raporları incelendiğinde zararlı bağlantıların bazı raporlarda etkin tehlikeli (fanged) bazı raporlarda ise etkisizleştirilmiş (defanged) olarak belirtildiği gözlemlenmiştir. Fanged olarak belirtilen bir adres, onun aktif ve potansiyel olarak zararlı olduğunu ifade etmektedir. Yani fanged halde yazılan terimler GİG olarak kabul

edilmektedir. Fanged adreslere yanlışlıkla tıklanırsa veya herhangi bir yöntem ile erişilmeye çalışılırsa zararlı faaliyetler gösteren kaynak ile iletişim kurulabilir. Bu durumun engellenmesi için ilgili metin anlam kaybına uğramaksız geçersiz hale getirilerek etkisizleştirilmektedir. RegEx eşleştirmesi yapılarak ayrıştırılan alan adları, alt alan adları, URL ve e-posta adreslerine ilişkin fanged ve defanged örnekler Çizelge 4.4'te sunulmaktadır.

Çizelge 4.4. Etkin tehlikeli ve etkisizleştirilmiş adres örnekleri

GİG Türü	Etkin Tehlikeli	Etkisizleştirilmiş
IP adresi	104.225.129.103	104.225.129[.]103
Alan adı	exoticlol.com	exoticlol[.]com
Alt alan adı	secret.whoami.la	secret.whoami[.]la
E-posta adresi	haveip2015@gmail.com	haveip2015@gmail[.]com
URL	https://ctmnews.kr/member/process/success.php	hxxps://ctmnews[.]kr/member/process/success.php

Tez çalışmasında APT analiz raporlarından RegEx eşleştirme yöntemleri ile elde edilen tüm GİG'ler fanged hale getirilerek HBA'ya dahil edilmiştir. Bunun sebebi, bir GİG'i defanged hale getirmek için birçok yöntem ve varyasyon olması ve HBA'da aynı varlığı ifade etmeye yönelik kayıtların tekil olmasının hedeflenmesidir.

4.2. Blok Listelerinin İşlenmesi ve STİ Zenginleştirme

Toplanan STİ'den HBA üretilmesi ve zenginleştirilmesi işlem basamaklarından ikincisinde, blok listeler üzerinde çalışılmaktadır. Elde edilmesi hedeflenen HBA'nın yüksek bağlamsallığa ulaşmasına hizmet edecek blok listeler araştırılmıştır. Literatür taramasında karşılaşılan blok listelerin bazıları ve tez çalışması kapsamında dikkate alınan diğer blok listeler bulunmaktadır. Bu blok listesi kaynakları, içerdikleri STİ'nin türü ve STİ'nin içeriği Çizelge 4.5'te sunulmaktadır.

Çizelge 4.5. Blok listesi kaynakları, içerdikleri STİ'nin türü ve içeriği

STİ Türü	STİ Varlığı	Kaynak
Teknik	Alan Adı Blok Listesi	Crypto coin phishing domains [123], Anudeep crypto coin miner blocklist [124], Anudeep ad server blocklist [125], USOM blocklist [126], URLhaus malware domains list [127], AntiSocial BlockList [128]

Çizelge 4.5. (devam) Blok listesi kaynakları, içerdikleri STİ'nin türü ve içeriği

STİ Türü	STİ Varlığı	Kaynak
Teknik	IP Adres Blok Listesi	Proofpoint Compromised IPs [129], SANS Dshield Blocklist [130], CIArmy blocklist [131], Abuse.ch Botnet C&C Server Blocklist [132], Feodo Tracker IP Recommended Blocklist [133], Feodo Tracker IP Blocklist [134], Darklist.de Network Attackers IP List [135], Tor IPs [136]
Teknik	URL Blok Listesi	Crypto Coin Phishing URL Addresses [137], URLhaus Recent Malicious URL List [138], URLhaus Full Malicious URL List [139], ThreatFox Recent Malicious URL List [140], ThreatFox Full Malicious URL List [141]
Teknik	Zararlı Yazılım Dosya Özeti Listesi	Malware Bazaar Recent Malicious File Hashes [142], Malware Bazaar Full Malicious File Hashes [143]
Teknik	DNS Sunucu Listesi	DNS Nameservers [144], DNS Valid Nameservers [143]
Teknik	SSL Sertifika Blok Listesi	Abuse.ch SSL Blocklist [146]

Yukarıda sunulan blok listeler ilgili adreslerden indirilmiştir. Her bir dosya satır bazlı okunarak <varlık, ilişki, varlık> üçlüleri oluşturulmuştur. Bu üçlüler oluşturulurken karşılaşılan varlıkların içinde veriyi bozan idafeler, ön ek veya son ekler varsa temizlenmiştir. Blok listelerden oluşturulan üçlüler <reporter_platform, REPORT, ip>, <reporter_platform, REPORT, domain>, <reporter_platform, REPORT, url>, <reporter_platform, REPORT, sha1> ve <reporter_platform, REPORT, hash> gibidir. Oluşturulan üçlüler çizge veri tabanında yoksa eklenmektedir. Üçlüyü oluşturan varlıklardan birisi çizge veritabanında zaten varsa, mevcut varlık ile yeni gelen varlık çizge veri tabanında ilişkilendirilmektedir. STİ zenginleştime işlemlerinden sonra oluşan HBA'da yer alan GİG'ler ve sayıları Çizelge 4.6'te, ilişkiler ve sayıları Çizelge 4.7'de sunulmuştur.

Çizelge 4.6. Blok listelerin entegre edildiği HBA'daki düğüm türleri ve sayıları

Düğüm Türü	Düğüm Sayısı
apt_group	557
apt_report	734
asn	2669
country	164
cve	271

Çizelge 4.6. (devam) Blok listelerin entegre edildiği HBA'daki düğüm türleri ve sayıları

Düğüm Türü	Düğüm Sayısı
dns	209
domain	30496
domain_registrar_name	1134
email	991
facebook_handle	27
filename	2592
industry	124
ip	291907
ip4net	123
issuer_country	606
issuer_name	607
issuer_organization	606
linkedin_handle	8
malware	972
md5	11161
mitre_technique	183
nameserver	9014
package_name	3
phone	142
port	98
port_scan_domain	6107
port_scan_ip	12700
protocol	1
registrant_company	2590
reporter_platform	27
service_name	124
service_os_type	9
service_product	555
service_version	899
sha1	9489
sha256	4627
ssl_certificate	416
ssl_port	607
subdomain	9015
tor	52
tweet	762
twitter_handle	494
url	8402
uuid	170
whois_domain	13386
whois_ip	58408
youtube_channel	2
youtube_handle	5

Çizelge 4.7. Blok listelerin entegre edildiği HBA'daki ilişki türleri ve sayıları

İlişki Türü	İlişki Sayısı
ALSO_KNOWN_AS	479
BELONGS	74
COVER	16597
HAS	399
HAS_APT_GROUP	1108
HAS_NAMESERVER	25488
HAS_OS_TYPE	8281
HAS_PRODUCT	33932
HAS_SCAN_RESULT	18807
HAS_SERVICE_VERSION	16786
ISSUE	606
ISSUED_BY	606
ISSUED_FROM	606
IS_OPEN_ON	65723
IS_SERVED	49281
LOOKUP	71781
MENTIONS	67286
POSTED	762
POSTED_IN	792
PROTECTED	1523
REGISTERS	26772
REPORT	249521
RESOLVES	948
RUNS_ON	16930
SERVE	607

Blok listelerin ayrıştırılması ve <varlık, ilişki, varlık> üçlülerine dönüştürülerek HBA'ya entegre edilmesi sayesinde tezde önerilen çok katmanlı çerçeve modelin ilk STİ zenginleştirme faaliyeti gerçekleştirilmiş olmaktadır. Bazı GİG'ler ile birden çok blok listesinde karşılaşılmaktadır. Bu durum, STİ'ye dönüşen GİG'in güvenilirliğini artırması gerektiği şeklinde yorumlanmıştır. HBA'daki her düğümün ağırlık (weight) adında ortak bir özelliği bulunmaktadır. Bir GİG ile ilk defa karşlılaşıldığında weight = 1 olarak ilk değer ataması yapılmaktadır. Aynı GİG ile farklı blok listelerde her karşılaşıldığından weight = weight + 1 formulüne göre güncelleme yapılmaktadır. Weight değeri yükselen GİG'lerin tezin ilerleyen bölümlerinde bağlantı tahminlemede daha baskın rol alması hedeflenmiştir.

4.3. Aktif Veri Toplama ve STİ Zenginleştirme

Toplanan STİ'den HBA üretilmesi ve zenginleştirilmesi işlem basamaklarından üçüncüsünde, aktif veri toplama yöntemleri kullanılmaktadır. Elde edilmesi hedeflenen HBA'nın yüksek bağlamsallığa ulaşmasına hizmet edecek harici aktif veri toplama yöntemleri oluşturulmuştur. Literatür taramasında karşılaşılan aktif veri kaynaklarından bazıları ve tez çalışmasına özgü aktif veri toplama yöntemleri belirlenmiştir. Bu STİ zenginleştirme basamağında kullanılan yöntemler, yöntemin sonucunda elde edilen STİ'nin türü ve elde edilen STİ varlığı ve aktif veri toplama işleminin gerçekleştirildiği kaynak Çizelge 4.8'te sunulmuştur.

Çizelge 4.8. Aktif veri toplama kaynakları, içerdikleri STİ'nin türü ve içeriği

STİ Türü	STİ Varlığı	Kaynak
Teknik	DNS sorgu sonuçları	DNS sunucuları
Teknik	SSL sertifikaları	Blok listelerden ve APT analiz raporlarından toplanan alan adları, alt alan adları ve URL'ler
Teknik	Whois sorgu sonuçları	Blok listelerden ve APT analiz raporlarından toplanan alan adları ve IP adresleri
Taktik	Port tarama sonuçları	Blok listelerden ve APT analiz raporlarından toplanan IP adresleri ile alan adları ve alt alan adlarının çözüldüğü IP adresleri
Operasyonel	X (Twitter) gönderileri	APT analiz raporlarından ve blok listelerden elde edilen GİG'ler

İkinci zenginleştirme basamağında, çizge veritabanında bulunan alan adları ve alt alan adları, birinci zenginleştime basamağında elde edilen DNS sunucuları üzerinden sorgulanarak çözüldükleri IP adresleri belirlenmektedir. Bu sorgulama sonucunda <domain, RESOLVES, ip> ve <subdomain, RESOLVES, ip> üçlüleri elde edilerek çizge veritabanına kaydedilmektedir. Tez çalışmasında, alan adlarının ve alt alan adlarının DNS sorguları zaman aralıklarıyla birden çok defa gerçekleştirilmiştir. Farklı cevaplar elde edildiğinde, ilgili alan adı (domain) ve/vaya alt alan adına (subdomain) ait düğüme yeni bir IP adresi düğümü ilişkilendirilmiştir. Bunun sebebi aktif olarak faaliyetlerine devam eden zararlı adreslerin, çözüldükleri IP adreslerini değiştirme eğiliminde olmalarıdır. Bu tür durumların ortaya çıkarılması için rastgele olarak farklı lokasyonlardaki DNS sunucuları tercih edilmiştir. Belirli bir liste içinde rastgele DNS sunucularının seçilmesinin temelde bir sebebi bulunmaktadır. Geliştirilen uygulamanın kısa süre içerisinde eş zamanlı olarak çok sayıda IP çözümleme isteğinde bulunması sebebiyle, DNS sunucuları tarafından bloklanmamak amaçlanmıştır. Tercih edilen DNS sunucularının listesi Çizelge 4.9'da sunulmaktadır.

STİ zenginleştirme işlemleri arasına, alan adları ve alt alan adlarının çözüldükleri IP adreslerinin belirlenmesi prosedürünün eklenme sebeplerinden bir diğeri ise aynı IP adresine yönlenen çok sayıda zararlı adresin birbirleri ile ilişkilendirilmesini mümkün kılmaktır. Ayrıca çok sayıda zararlı faaliyetin ilişkilendiği IP adresleri üzerinden sunulan diğer hizmetlerin de zararlı olma potansiyeli ortaya çıkarılmış olmaktadır. Bu tür IP adreslerine çözülen ama henüz zararlı olduğuna dair bir kanıta sahip olunmayan alan adları ve alt alan adlarına şüphe ile yaklaşılması için bir imkân elde edilmektedir.

Çizelge 4.9. IP çözümlemede kullanılan DNS sunucuları

Alan Adı	IP Adresi	Lokasyon
ns2.seolizer.de	170.64.147.31	Avustralya, Sidney
ns2.directontv.com	41.204.224.38	Nijerya, Lagos
ns.ramboll.com	195.137.178.35	Danimarka, Kopenhag
ns.terakata.co.jp	210.238.242.249	Japonya, Kuradani
6cb322d6.cst.lightpath.net	108.179.34.214	ABD, New York
host81-138-71-238.in-addr.btopenworld.com	81.138.71.238	İngiltere, Lisburn
host-173.9.16.81.ucom.am	81.16.9.173	Ermenistan, Vanadzor
srv77201-206152.vps.etecsa.cu	152.206.201.77	Küba, Havana
vmi497757.contaboserver.net	161.97.150.198	Almanya, Düsseldorf
dns2.bau.edu.jo	87.236.233.117	Ürdün, Amman
mailgw.polyram-group.com	199.203.131.151	İsrail, Tel Aviv
ppp170-23.bakinter.net	217.64.23.170	Azerbaycan, Bakü
gns.paltel.net	213.244.72.31	Filistin, Nablus
85.105.81.252.static.ttnet.com.tr	85.105.81.252	Türkiye, İstanbul
93-87-74-150.static.isp.telekom.rs	93.87.74.150	Sırbistan, Novi Sad

Öte yandan güvenilir hizmetler sunan yüksek repütasyona sahip servislerin de çok sayıda farklı IP adresi ile karşılabilmektedir. Örneğin youtube.com için karşılaşılan IP adresleri 172.217.169.206, 142.251.46.23 ve 208.65.153.253 şeklindedir. Google, Netflix, Youtube, Instagram gibi çok sayıda fiziksel lokasyona dağılmış uygulamalara sahip sistemler döngüsel (round robin) DNS hizmetleri kullanmaktadır. Döngüsel DNS, yük dengelemenin özel bir yük dengeleme donanımı/yazılımı kullanmak yerine yetkili bir ad sunucusu (name server) olarak adlandırılan bir tür DNS sunucusu tarafından yapıldığı bir yük dengeleme tekniğidir. Döngüsel DNS, bir web sitesi veya hizmetin içeriklerinin birden fazla yedekli web sunucusunda barındırıldığı durumlarda kullanılabilir. DNS yetkili ad sunucusu bir IP adresi için sorgulandığında, sunucu her seferinde farklı bir adres vermektedir ve bu işlem bir döngü halinde gerçekleşir.

Bu yöntem, özellikle yedekli web sunucuları coğrafi olarak ayrılmış olduğunda ve geleneksel yük dengelemenin zor olduğu durumlarda kullanışlıdır. Zararsız ve yüksek repütasyona sahip alan adlarının, APT analiz raporlarına ve blok listelere dahil olma ihtimaline karşı 1 milyon kayıttan oluşan bir beyaz liste (white list) kullanılmıştır [147]. Pochat ve diğerleri, Tranco adını verdikleri beyaz listeyi Cloudflare Radar, Farsight, Majestic ve Cisco Umbrella listelerinin birleştirilmesi ve yeniden derecelendirilmesiyle elde etmiştir. Beyaz listede bulunan alan adları için DNS sorguları yapılmamış ve HBA'ya dahil edilmemiştir.

Alan adları ve alt alan adlarının IP adresine çözülmesi sürecinde veri kalitesini olumsuz etkileyecek bir diğer unsur ise DDoS saldırılarına karşı kullanılan koruma teknolojileridir. Web uygulamaların hizmet verdiği sunucuların IP adreslerinin, saldırganlar tarafından bilinmesini engellemek ve IP adreslerine doğrudan erişimi engellemek amacıyla Cloudflare, Google Cloud Armor, Radware, Akamai gibi DDoS koruma sistemleri, web uygulama güvenlik duvarları (web application framework, WAF) gibi ürün ve hizmetler kullanılabilmektedir.

Tranco haricinde kullanılan bir diğer beyaz liste ise bu güvenlik duvarları ve DDoS koruma sistemlerinin ilan edilen IP adreslerinden meydana gelmektedir. Kullanılan sınıfsız alan adları arası yönlendirme (classless inter-domain routing, CIDR) adreslerini içeren 15 kayıttan oluşan liste Çizelge 4.10'da sunulmaktadır. Eğer HBA üzerindeki bir zararlı alan adının çözüldüğü IP adresi listedeki CIDR aralıklarına denk gelmekteyse, bu IP adresi HBA'ya tekrarlı halde eklenmemiştir, tamamını temsilen tek kayıt halinde eklenmiştir.

Çizelge 4.10. DDoS koruma ve WAF hizmetlerinin IP adresleri

Sağlayıcı	CIDR	IP Aralığı	Ondalık Formattaki IP Aralığı
Cloudflare	103.21.244.0/22	103.21.244.0 - 103.21.247.255	1729491968 - 1729492991
	103.22.200.0/22	103.22.200.0 - 103.22.203.255	1729546240 - 1729547263
	103.31.4.0/22	103.31.4.0 - 103.31.7.255	1730085888 - 1730086911
	2400:cb00::/32	2400:CB00:0000:0000:0000:0000:0000:0000 - 2400:CB00:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF	4785632517740651271363311 5462103465984 - 4785632525663467522789745 3055647416319

Çizelge 4.10. (devam) DDoS koruma ve WAF hizmetlerinin IP adresleri

Sağlayıcı	CIDR	IP Aralığı	Ondalık Formattaki IP Aralığı
	2606:4700::/32	2606:4700:0000:0000:0000:0000:0000:0000 - 2606:4700:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF	50543257672059871404715951523469721600 - 50543257751288033918980289117013671935
Akamai	2.17.128.0/19	2.17.128.0 - 2.17.159.255	34701312 - 34709503
	2.16.224.0/19	2.16.224.0 - 2.16.255.255	34660352 - 34668543
	2.22.192.0/19	2.22.192.0 - 2.22.223.255	35045376 - 35053567
	2.23.224.0/19	2.23.224.0 - 2.23.255.255	35119104 - 35127295
	95.100.0.0/20	95.100.0.0 - 95.100.15.255	1600389120 - 1600393215
Radware	141.226.101.0/24	141.226.101.0 - 141.226.101.255	2380424448 - 2380424703
	141.226.102.0/24	141.226.102.0 - 141.226.102.255	2380424704 - 2380424959
	141.226.254.0/24	141.226.254.0 - 141.226.254.255	2380463616 - 2380463871
	141.226.96.0/24	141.226.96.0 - 141.226.96.255	2380423168 - 2380423423
	141.226.97.0/24	141.226.97.0 - 141.226.97.255	2380423424 - 2380423679

Birinci zenginleştirme basamağında elde edilen blok listelerde geçen alan adları ile IP adresleri ve APT analiz raporlarından ayrıştırılan alan adları ve IP adresleri için Réseaux IP Européens (RIPE) sunucularından Whois sorguları yapılmaktadır. Whois sorgusu sonucunda, alan adı kayıt şirketi adı, alan adı sahibinin adı, alan adı sahibinin ülkesi, alan adı sahibinin telefonu, alan adı sunucuları gibi birçok veri elde edilmektedir. Bu veri, HBA'da kötü amaçlı alan adlarının, URL'lerin, alt alan adlarının ve IP adreslerinin ortak özelliklerini bulma fırsatı sunmaktadır. Zararlı faaliyetler yürüttüğü tespit edilen bir alan adının sahibine dair gerçek bilgiler elde edilebilirse, bu saldırgana ait varsa diğer alan adlarının keşfedilmesi mümkündür.

Whois sorguları, çok sayıda zararlı alan adının kiralandığı aracı kayıt şirketine ulaşmak için de bir fırsat sunmaktadır. Çok sayıda zararlı alan adını kaydetme noktasında merkezileşen şirketler, tekdit aktörleri tarafından sömürülen bir karaktere sahip olabilir. Bu

bakış açısına bağlı olarak whois sorgu sonuçlarının HBA'ya dahil edilmesi stratejik öneme sahiptir.

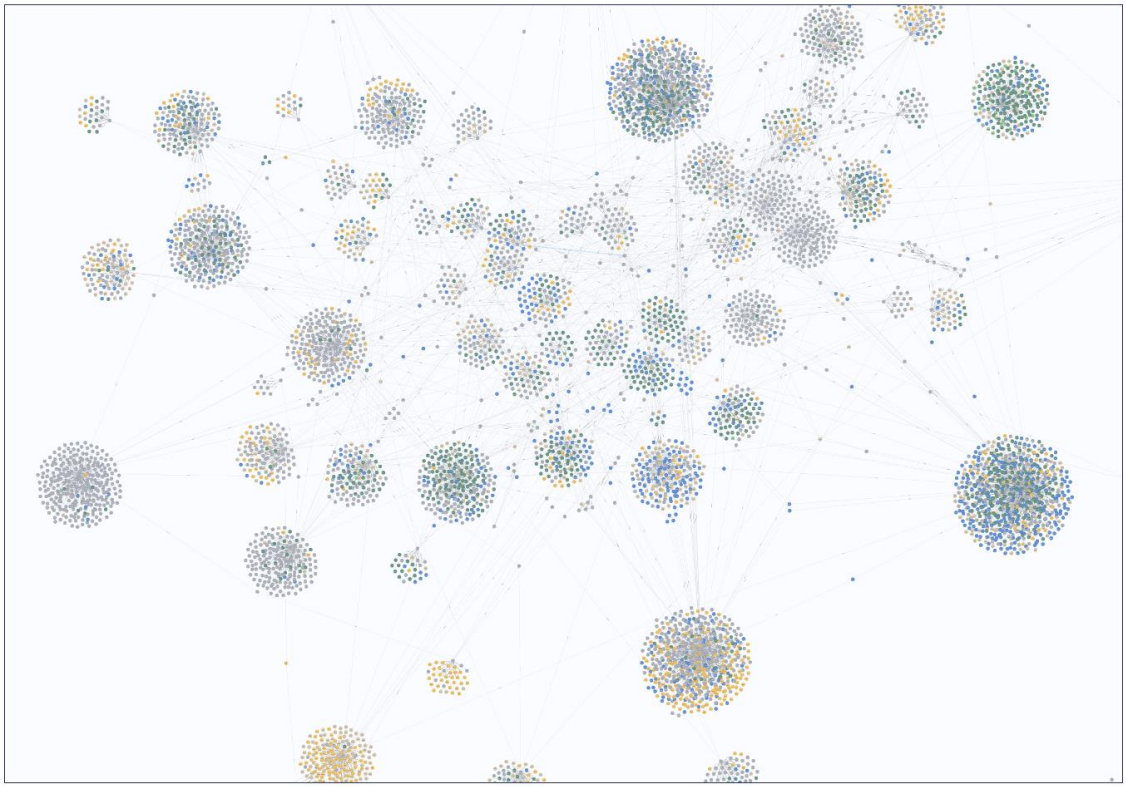
Whois sorgu sonuçlarında elde edilen tüm bilgiler <domain, LOOKUP, whois_domain> ve <ip, LOOKUP, whois_ip> üçlüleri oluşturularak çizge veritabanına kaydedilmektedir. Bu üçlülere bağlı olarak alt üçlüler oluşturmakta ve ana düğümler ile ilişkilendirilmektedir. Oluşturulan alt üçlüler <domain_registrar_name, REGISTERS, whois_domain>, <nameserver, HAS_NAMESERVER, whois_domain>, <registrant_company, REGISTERS, whois_domain> ve <ip, COVER, asn> şeklindedir. HBA üzerinde çok sayıda zararlı alan adının sahibi veya yöneticisi durumundaki kişilere ait ortak veriler tespit edilerek, STİ araştırmasında yeni hedeflere yönelmenin önü açılmaktadır.

Üçüncü zenginleştirme basamağında, çerçeve modelin birinci ve ikinci basamaklarda toplanan IP adresleri üzerinde port taraması işlemleri yapılmaktadır. Özellikle APT analiz raporlarında geçen ve blok listelerde adı geçen zararlı alan adlarının çözüldüğü IP adresleri hedef alınarak port tarama, hedef portta servis keşfi, keşfedilen servise ait varsa bilinen zafiyetlerin tespiti yapılmaktadır. Bu işlemler sonucunda <ip, HAS_SCAN_RESULT, port_scan_ip> üçlüleri veritabanına kaydedilmektedir. Port tarama işlemleri Nmap aracı kullanılarak gerçekleştirilmiştir. Tarama sonucunda elde edilen veriler alt detaylarına bölünerek <protocol, RUNS_ON, port_scan_ip>, <port, IS_OPEN_ON, port_scan_ip>, <service_name, IS_SERVED, port_scan_ip>, <service_product, HAS_PRODUCT, port_scan_ip>, <service_version, HAS_SERVICE_VERSION, port_scan_ip>, <service_os_type, HAS_OS_TYPE, port_scan_ip> üçlüleri oluşturularak HBA'ya entegre edilmektedir.

Port tarama faaliyetlerinin yürütülme sebebi, tez kapsamında STİ üretme yöntemlerine geniş bir perspektiften bakılmasıdır. Uzak bir bilgisayarın doğrudan zararlı servisler sunuyor olması ve zararlı faaliyetleri destekleyen ağ işlemleri yapıyor olması haricinde, bu bilgisayarın sahip olduğu zafiyetler sebebiyle saldırganlar tarafından ele geçirilmeye aday olması da değerli bir STİ'dir. Sunduğu hizmetlerden kaynaklı güvenlik zafiyetlerine sahip bilgisayarlar, gelecekte gerçekleşebilecek saldırılarda aparat olarak kullanılabilmektedir.

Son olarak ise birinci ve ikinci basamaklarda toplanan alan adları, alt alan adları, IP adresleri ve URL'ler X (Twitter) gönderileri üzerinden etiket (hashtag) olarak

sorgulanmaktadır. X üzerinde aranan GİG'ler ile karşılaşıldığı durumda, ifadenin geçtiği gönderi ve gönderinin sahibi elde edilmektedir. Bunun sonucunda <ip, POSTED_IN, tweet>, <domain, POSTED_IN, tweet>, <subdomain, POSTED_IN, tweet> ve <url, POSTED_IN, tweet> üçlüleri oluşturularak HBA'ya kaydedilmektedir. HBA'daki GİG'ler ve X gönderileri arasındaki bağlamsallığı artırmak için <twitter_handle, POSTED, tweet> üçlüleri de oluşturularak HBA'ya entegre edilmektedir. Böylece GİG'ler hakkında gönderiler yazan X kullanıcıları da HBA'ya dahil edilmiş olmaktadır. X gönderileri içinde saldırılara dair ipuçları aranması operasyonel STİ kapsamında yapılan bir işlemdir. HBA üzerindeki düğümler arasında ilişkilerin çoğaltılması, grafın bağlamsallığını artırmak için imkân sağlamaktadır.

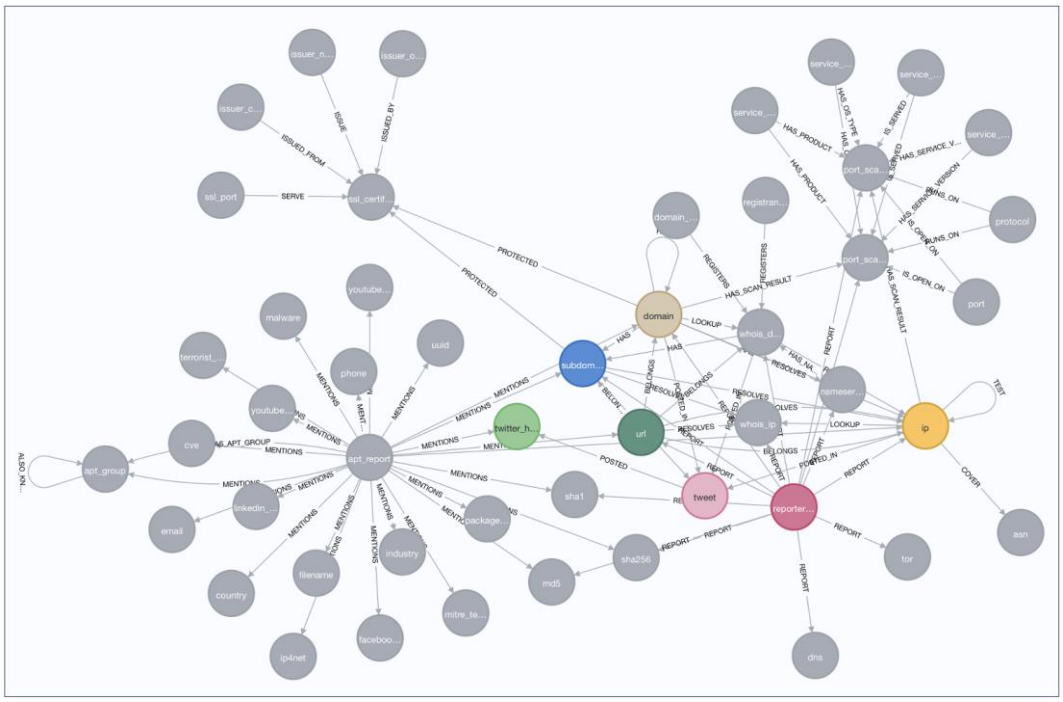


Şekil 4.4. HBA'nin kısmi görünümü

Tezin bu bölümünde yapılan çalışmalar sonucunda elde edilen HBA'nın kısmı bir görünümü Sekil 4.4'te sunulmaktadır. Görselde farklı türlerdeki düğümler ve aralarındaki ilişkiler görüntülenmektedir.

5. ÇİZGE TABANLI TAHMİNLEME MODELİ GELİŞTİRİLMESİ

Zenginleştirilmiş STİ'den meydana gelen HBA üzerinde ilk olarak meta-yol ve meta-çizge analizleri yapılmaktadır. Meta yol, varlıklar arasındaki ilişkiyi tanımlayan bir dizi varlık türü ve ilişki türüdür. Meta-yollar, zengin anlamsal bilgileri ortaya çıkarmak için HBA'larda yaygın olarak kullanılmaktadır [148]. Meta yollar bir araya gelerek meta-çizgeyi oluşturmaktadır. Şekil 5.1'de HBA üzerinde keşfedilen meta-yollar ve meta-çizge sunulmaktadır.



Şekil 5.1. HBA üzerinde keşfedilen meta-yollar ve meta-çizge

Meta-yolların farkında olmak, HBA üzerindeki ağ analizi işlemleri için önemlidir. Çünkü meta-yollar ve meta-çizge, varlıklar arasındaki karmaşık anlamsal ilişkileri ortaya çıkarmada geniş bir açı ve yorumlanabilirlik sağlamaktadır. Elde edilen meta-çizge üzerinde 48 adet düğüm türü ve 25 adet ilişki türü yer almaktadır. Nihai HBA'da ise 484.249 adet düğüm ve 675.696 adet ilişki bulunmaktadır.

Tezin bu bölümünde çizge tabanlı bir tahminleme modeli geliştirilmesine yönelik çalışmalara yer verilmektedir. Tahminleme modelinin eğitilmesi sürecine geçmeden önce elde edilen nihai HBA'nın yakından tanınması hedeflenmiştir. Milyonlarca düğüm ve

ilişkiden meydana gelen bir HBA elde edildiği için görselleştirme yöntemleri ile çizge üzerinde yorumlarda bulunulması ve tahminleme modeli eğitimine yönelik kararlar verilmesi mümkün değildir. Çizge algoritmaları kullanılarak, mevcut HBA hakkında metrikler elde edilmesi, düğümler ve aralarındaki ilişkilerin birçok açıdan yorumlanması ihtiyacı meydana gelmiştir.

5.1. Nihai HBA'nın Çizge Algoritmaları ile Analizi

Tahminleme modelinin eğitimi öncesinde HBA'yı tanımaya yönelik yapılan çalışmalar bu başlık altında ele alınmıştır. Yapılan çalışmalar; düğümlerin merkeziliğinin keşfedilmesi, HBA'daki toplulukların belirlenmesi, düğümler arasındaki yolların keşfi ve HBA'daki önemli düğümlerin benzerliklerinin belirlenmesi üzerinedir.

5.1.1. Düğümlerin merkezilik analizi

Merkezilik, bir ağdaki farklı düğümlerin önemini belirlemek için kullanılmaktadır [149]. Merkezilik algoritmaları, bir ağdaki herhangi bir düğümün önemini hesaplamak için çizge teorisini kullanmaktadır. Bu alandaki algoritmalar, gürültü (noise) etkisi yapan düğüm ve ilişkileri arka plana iterek, odaklanarak fayda sağlanabilecek düğüm ve ilişkileri ortaya çıkarılmasını sağlarlar. Zenginleştirilmiş ve bağlamsallığı artırılmış STİ'den oluşturulan HBA milyonlarca düğüm ve ilişkiye sahip olduktan sonra, bu denli büyük bir çizge üzerinde merkezilik ve öneme dair yorumlarda bulunmak için sistematik yöntemlere ihtiyaç duyulmaktadır.

Aradalık merkeziliği

Aradalık merkeziliği, bir düğümün HBA üzerindeki bilgi akışına etkisini tespit etmenin bir yoludur. HBA'nın bir kısmından diğerine köprü görevi gören düğümleri bulmak için kullanılmaktadır. Aradalık merkeziliği, HBA'daki tüm düğüm çiftleri arasındaki en kısa yolları hesaplamaktadır. Her düğüm, üzerinden geçen en kısa yol sayısına dayalı olarak bir skor elde etmektedir. Diğer düğümler arasındaki en kısa yollarda daha sık yer alan düğümler, daha yüksek aradalık merkeziliği skoruna sahip olmaktadır [150-152]. Yüksek aradalık skoru, bir düğümün ağdaki farklı kümeler üzerinde otoriteye sahip olduğunu veya bu düğümün her iki kümenin de ortak alanında yer aldığını göstermektedir. Tez

çalışmasında oluşturulan HBA ağırlıklandırılmış düğüm ve ilişkilerken oluşmaktadır. Bir düğümün içerdiği bilgi ile birden çok STİ kaynağında karşılaşıldıysa zaman içinde ilgili düğüme artan ağırlık atanmaktadır. Bir ilişki, birden çok STİ kaynağından dolayı meydana geldiyse, zaman içinde ilgili ilişkiye artan ağırlık atanmaktadır. Bir çizge yapısındaki düğümler arasındaki en kısa yolu bulmak için literatürdeki çeşitli algortimalar incelenmiştir. Bunlar arasında heterojen düğümler ve ağırlıklandırılmış ilişkiler üzerinde çalışmak üzere uygun olanlar tespit edilmiştir. Bu kapsamda Bellman-Ford, Dijkstra, Floyd–Warshall, A-Star, Yen algoritmaları kıyaslanarak bir seçim yapılmıştır [153,154]. En kısa yolu bulmak üzerine incelenen algoritmalar Çizelge 5.1'de kıyaslanmaktadır.

Çizelge 5.1. Çizgeler üzerinde en kısa yolu bulma algoritmaları

Algoritma	Özellikleri	Zaman Karmaşıklığı
Bellman-Ford Algoritması	1- Negatif ilişki ağırlıklarını işleyebilmektedir. 2- Ağdaki negatif döngüleri tespit edebilmektedir. 3- Negatif ilişki ağırlıklarının bulunmadığı senaryolarda alternatif algoritmalara görece az verimlidir.	$O(V \cdot E)$
Dijkstra Algoritması	1- Pozitif kenar ağırlıklarına sahip çizgelerle çalışırken etkilidir. 2- Çizgedeki bir kaynak düğümden hedef veya diğer tüm düğümlere giden en kısa yolu bulmak için tasarlanmıştır.	$O((V+E) \cdot \log(V))$
Floyd–Warshall Algoritması	1- Tüm düğüm çiftleri arasındaki en kısa yolları hesaplamaktadır. 2- Çizgenin negatif ilişki ağırlıklarına sahip olduğu ancak negatif ağırlık döngülerinin olmadığı durumlarda kullanışlıdır. 3- Tek seferde tüm düğüm çiftleri arasındaki en kısa yolları hesaplamaktadır.	$O(V^3)$
A-Star Algoritması	1- Ağırlıklı bir çizgede başlangıç düğümü ile hedef düğüm arasındaki en kısa yolu bulmak için kullanılmaktadır. 2- Hedefe götüren aramayı yönlendirmek için sezgisel (heuristic) yöntemler kullanmaktadır. 3- Coğrafi navigasyon, rota planlama, arazi analizi gibi alanlarda kullanılarak maliyet etkin yol oluşturulmasında kullanılmaktadır.	En kötü durumda: $O(b^d)$ En iyi durumda: $O(d)$ b: dallanma faktörüdür. (düğüm başına ortalama çocuk sayısı) d: en sığ hedef düğümün derinliğidir.
Yen Algoritması	1- Düğümler arasında ağ esnekliği ve optimizasyonu açısından yararlı olan birden fazla alternatif yol sunmaktadır. 2- Her alternatif en kısa yol hesaplaması sebebiyle hesaplama karmaşıklığı artmaktadır. 3- Her yineleme, Dijkstra algoritmasının birden çok kez çalıştırılmasını gerektirmektedir.	$O((E+V) \cdot \log(V))$

Mevcut HBA'da ilişkiler ağırlıklı ve pozitif olarak oluşturulmuştur. Dijkstra algoritması seyrek çizgelerde, yani ilişki sayısının düğüm sayısının karesinden çok daha az olduğu ($E < V^2$) durumlarda Floyd-Warshall ve Yen algoritmasına kıyasla daha verimlidir. Dijkstra algoritmasının zaman karmaşıklığı pozitif ağırlıklı bağlantılara sahip senaryolarda da diğer algoritmalara döre daha düşüktür. Düğümler arasındaki en kısa yolların belirlenmesi ve buna bağlı olarak aradalık merkeziliği hesaplamak için Dijkstra algoritmasından yararlanılmıştır [155].

Dijkstra Algoritması bir kaynak düğümden başlayarak, o düğüm ile çizgedeki diğer tüm düğümler arasındaki en kısa yolu bulmak için çizgeyi analiz etmektedir. Algoritma, her düğümden kaynak düğüme olan mevcut bilinen en kısa mesafeyi takip etmektedir ve daha kısa bir yol bulduğunda bu değerleri güncellemektedir. Kaynak düğüm ile başka bir düğüm arasındaki en kısa yolu bulduğunda, o düğümü "ziyaret edildi" olarak işaretlemektedir ve yola eklemektedir. İşlem, çizgedeki tüm düğümler yola eklenene kadar devam etmektedir. Böylece kaynak düğümü, her düğüme ulaşmak için mümkün olan en kısa yolu takip ederek diğer tüm düğümlere bağlayan bir yola sahip olunmaktadır. Şekil 5.2'de Dijkstra Algoritmasının kaynak bir düğümden diğer tüm düğümlere giden en kısa yolu bulan işlem adımları gösterilmektedir.

Fonksiyonda ilk olarak bir döngü oluşturulmakta ve çizgenin tüm düğümleri iterasyonla dolaşılmaktadır (satır 2). Her düğümün başlangıçtaki mesafesi sonsuz olarak atanmaktadır. Bu işlem, başlangıçta tüm düğümlere olan mesafenin bilinmediğini belirtmektedir (satır 3). Her düğümün önceki düğümü belirsiz olarak atanmaktadır. Bu işlem, başlangıçta her düğümün yolunun bilinmediğini belirtmektedir (satır 4). Her düğüm öncelik sırasıyla tutulan *Q* adlı veri yapısına eklenmektedir (satır 5). Kaynak düğümün kendisine olan mesafe sıfır olarak atanmaktadır, çünkü bir düğümün kendisine olan mesafesi her zaman sıfırdır (satır 6).

```
1   fonksiyon dijkstraEnKisaYolunuBul(Graf, kaynak):
2       Graf.Dugumler icindeki her bir dugum v icin:
3           uzaklik[v] ← SONSUZ
4           onceki[v] ← BELİRSİZ
5           v'yi Q'ya ekle
6       uzaklik[kaynak] ← 0
7       Q bos değilken:
8           u ← Q icindeki uzaklik[u] degeri en kucuk olan dugum
9           u'yu Q'dan cikar
10          hala Q'da olan u'nun her komsusu v icin:
11              alternatifUzaklik ← uzaklik[u] + Graf.Baglantilar(u, v)
12              eger alternatifUzaklik < uzaklik[v] ise:
13                  uzaklik[v] ← alternatifUzaklik
14                  onceki[v] ← u
15      uzaklik[] ve onceki[]'i dondur
```

Şekil 5.2. Dijkstra algoritması ile en kısa yolu bulma adımları

Çizgedeki tüm düğümler üzerinde dönen döngü tamamlandıktan sonra, Q boş olmadığı sürece devam edecek bir döngü başlatılmaktadır. Bu döngü, tüm düğümler işlenene kadar devam etmektedir (satır 7). *Q* içinde mesafesi en küçük olan düğüm seçilmektedir ve bu düğüm *u* değişkenine atanmaktadır (satır 8). Seçilen *u* düğümü *Q*'dan çıkarılmaktadır (satır 9). *u* düğümünün hala *Q*'da bulunan her komşu düğümü *v* için bir döngü başlatılmaktadır (satır 10). *u* düğümünden *v* düğümüne olan kenarın ağırlığı *uzaklik[u]* ile toplanmaktadır ve bu *alternatifUzaklik*n olarak adlandırılmaktadır (satır 11). Eğer *alternatif Uzaklik* değeri *v* düğümüne mevcut bilinen mesafeden daha küçükse, takip eden satırlardaki güncellemeler yapılmaktadır (satır 12). *v* düğümüne olan en kısa mesafe güncellenmektedir (satır 13). *v* düğümünün önceki düğümü *u* olarak güncellenmektedir, yani en kısa yol *u* düğümünden geçmektedir (satır 14). Tüm düğümler işlendikten sonra, her düğüme olan en kısa mesafeleri içeren *uzaklik[]* ve her düğümün önceki düğümünü içeren *onceki[]* dizileri döndürülmektedir.

Yakınlık merkeziliği

Yakınlık merkeziliği, organizasyon ağlarını araştırmak için kullanılmaktadır. Yüksek yakınlık merkeziliğine sahip düğümler, organizasyon içinde önemli bilgi ve kaynakları kontrol etmek ve elde etmek için elverişli bir konumdadır [156, 157].

Yakınlık merkeziliği, bir çizgede bilgiyi verimli bir şekilde yayabilen düğümleri tespit etmenin bir yoludur. Bir düğümün yakınlık merkeziliği, diğer tüm düğümlere olan ortalama uzaklığını ölçmektedir. Yüksek yakınlık skoruna sahip düğümler, diğer tüm düğümlere en kısa mesafelere sahiptir. Her *u* düğümü için, yakınlık merkeziliği algoritması, tüm diğer düğümlere olan mesafelerinin toplamını hesaplamaktadır; bu hesaplama, tüm düğüm çiftleri arasındaki en kısa yolları temel alarak yapılmaktadır. Elde edilen toplam, daha sonra düğümün yakınlık merkeziliği puanını belirlemek için ters çevrilmektedir. Herhangi bir *u* düğümünün ham yakınlık merkeziliği 1 / (*u*'dan diğer tüm düğümlere olan mesafelerin toplamı) şeklinde hesaplanmaktadır.

```
fonksiyon yakinlikMerkeziligiHesapla(Graph, kaynak):
    uzaklik[] dizisini SONSUZ ile baslat
    yakinlikMerkeziligi[] dizisini 0 ile baslat
    kuyruk Q oluştur
    uzaklik[kaynak] = 0
    kaynak'ı Q'ya ekle
    Q bos olana kadar:
        su anki = Q'dan uzaklik[] degeri en kucuk dugumu cikar
        her bir su anki komsusu icin:
            eger komsu ziyaret edilmemisse:
                komsuyu Q'ya ekle
                komsunun mesafesini guncelle:
                    uzaklik[komsu] = uzaklik[su anki] + mesafe(su anki, komsu)
        toplamMesafe = uzaklik[] degerlerinin toplami
        eger toplamMesafe > 0 ise:
            yakinlikMerkeziligi[su anki] = (ulasilabilir dugum sayisi - 1) / toplamMesafe
        degilse:
            yakinlikMerkeziligi[su anki] = 0
    return yakinlikMerkeziligi[]
```

Şekil 5.3. Yakınlık merkeziliği hesaplama adımları

Elde edilen skoru normalize etmek, en kısa yolların toplamı yerine ortalama uzunluğu temsil etmesi açısından daha yaygındır. Bu değişiklik, farklı boyutlardaki çizgelerdeki düğümlerin yakınlık merkeziliklerinin karşılaştırılmasına olanak tanımaktadır. Herhangi bir *u* düğümünün normalize edilmiş yakınlık merkeziliği (düğüm sayısı - 1) / (*u*'dan diğer tüm düğümlere olan mesafelerin toplamı) şeklinde hesaplanmaktadır. Wasserman ve Faust, bağlı olmayan çizgelerde kullanmak üzere geliştirilmiş bir formül önermiştir. Formüldeki *n*'nin, *u*'dan erişilebilir düğüm sayısı olduğu durumda, herhangi bir *u* düğümü için Wasserman-Faust'a göre normalize edilmiş yakınlık merkeziliği, $(n\text{-}1)^2$ / ((düğüm sayısı - 1) x (*u*'dan diğer tüm düğümlere olan mesafelerin toplamı)) olarak hesaplanmaktadır.

Yakınlık merkeziliği hesaplamak için uygulanan işlem adımları ise Şekil 5.3'te sunulmuştur.

Uzaklık adlı bir dizi oluşturulmakta ve tüm elemanlarına *SONSUZ* ilk değer ataması yapılmaktadır. Bu dizi, işlem adımları boyunca her bir düğümün, kaynak düğümünden olan uzaklığını saklamaktadır (satır 2). *yakinlikMerkeziligi* adında bir dizi oluşturulmakta ve tüm elemanlarına başlangıçta *0* değer ataması yapılmaktadır. Bu dizi, işlem adımları boyunca her bir düğümün yakınlık merkeziliği değerlerini saklamaktadır (satır 3). Boş bir *Q* kuyruğu oluşturulmaktadır. Bu kuyruk, ziyaret edilecek düğümleri tutmak için kullanılmaktadır (satır 4). *kaynak* düğümünden başlanacağı için bu düğümün uzaklık değeri *0* olarak atanmaktadır (satır 5). *kaynak* düğümü kuyruğa eklenmektedir (satır 6). Kuyruk boşalana kadar sürecek bir döngü başlatılmaktadır (satır 7). *uzaklik* dizisinde en küçük uzaklık değerine sahip olan düğüm *su anki* olarak belirlenmektedir ve kuyruktan çıkarılmaktadır (satır 8). *su anki* düğümün her bir komşusu için tekrarlanmak üzere bir döngü başlatılmaktadır (satır 9). Komşu düğümüm daha önce ziyaret edilme durumu kontrol edimektedir (satır 10). Komşu düğüm kuyruğa eklenmektedir (satır 11). Komşu düğümün uzaklığı güncellenmektedir (satır 12 - 13). Tüm *uzaklik* değerlerinin toplamı hesaplanmaktadır (satır 14). Eğer *toplamMesafe* sıfırdan büyükse, *su anki* düğümün yakınlık merkeziliği hesaplanmaktadır (satır 15 - 16). Aksi takdirde, *yakinlik Merkeziligi[su anki]* değeri *0* olarak atanmaktadır (satır 17 - 18). Son olarak, *yakinlikMerkeziligi* dizisi geri döndürülmektedir.

Derece merkeziliği

Bağlantı merkeziyeti, her düğüm tarafından sahip olunan bağlantıların sayısına bağlı olarak elde ettiği bir önem puanıdır. Ağdaki her düğümün diğer düğümlerle kaç doğrudan bağlantıya sahip olduğunu anlamak için kullanılmaktadır.

```
1  fonksiyon dereceMerkeziligiHesapla(Graph):
2      dereceMerkeziyetiSozlugu = {}
3      Graph icindeki her dugum icin :
4          derece = komsulukListesiUzunlugu(Graph[dugum])
5          dereceMerkeziyetiSozlugu[dugum] = derece
6      return dereceMerkeziyetiSozlugu
```

Şekil 5.4. Derece merkeziliği hesaplama adımları

Ağdaki çok bağlantılı düğümleri, popüler düğümleri, muhtemelen en fazla bilgiye sahip düğümleri veya daha geniş ağlara hızla bağlanabilecek düğümleri bulmak için kullanılmaktadır [158, 159]. Şekil 5.4'te derece merkeziliği hesaplama adımları sunulmaktadır.

Öncelikle *dereceMerkeziyetiSozlugu* adında boş bir sözlük oluşturulmaktadır. Bu sözlük, her düğüm ve hesaplanan derece merkeziyetini anahtar-değer (key-value) saklamak için işlem adımları boyunca kullanılmaktadır (satır 2). *Graph* içindeki her düğüm üzerinde dönmek üzere bir döngü başlatılmaktadır (satır 3). *derece* adında bir değişken oluşturulmaktadır. Bu değişken, o anki düğümün komşuluk listesinin uzunluğuna eşittir. Yani, düğümün kaç tane komşusu olduğunu temsil etmektedir (satır 4). *dereceMerkeziyetiSozlugu* sözlüğüne ilgili düğümün derecesi kaydedilmektedir. Burada düğüm, sözlüğün anahtar değeri olarak kullanılmaktadır (satır 5). Son olarak, hesaplanan *dereceMerkeziyetiSozlugu* döndürülmektedir (satır 6).

5.1.2. Topluluk tespiti analizi

Topluluk tespit algoritmaları, çizge veri yapılarında düğüm gruplarının nasıl kümelendiğini veya bölündüğünü, ayrıca bu grupların güçlenme veya dağılma eğilimlerini değerlendirmek için kullanılmaktadır. Tez çalışmasında gerçek dünya verisinden oluşan bir HBA oluşturulmuştur. Gerçek dünya verisinden oluşan HBA'larda düğüm türleri ve bağlantı türleri yüksek seviyede çeşitlilik göstermektedir. Aynı zamanda düğümlerin ve sahip oldukları bağlantıların nicelik olarak bölgesel yoğunluğu da yüksek seviyede çeşitlilik gösterebilmektedir [160]. Bu sebeple heterojenlik kavramı HBA'larda tür çeşitliliği dışında niceliksel anlamda da geçerlidir. Topluluk tespiti, bir ağın yapısal özelliklerini ve dinamiklerini anlamaya yardımcı olmaktadır. Bu sayede ağdaki düğümler arasındaki ilişkiler ve etkileşimler daha iyi anlaşılabilmektedir.

Sosyal ağlar gibi yapılarda bilgi ve fikirler belirli topluluklar içinde hızla yayılabilmektedir. Topluluk tespiti, bu yayılımın nasıl gerçekleştiğini ve hangi düğümlerin bu süreçte kritik rol oynadığını anlamamıza yardımcı olmaktadır [161]. Aynı şekilde topluluk tespiti STİ alanında, tehdit aktörlerinin gerçekleşen saldırılara dahiliyetlerini anlamak ve çeşitli GİG'lerin farklı saldırılarda nasıl ortak olarak kullanıldığını ortaya çıkarmakta kullanılabileceği değerlendirilmektedir [162].

K-Çekirdek ayrışımı

K-çekirdek ayrışımı, bir çizgedeki düğümleri, derece dizisi ve çizgenin topolojisine dayalı olarak gruplara ayırma sürecini oluşturmaktadır. Bu algoritmada, *i*-çekirdek (*i*-core) terimi, bir alt çizgedeki her düğümün en az *i* dereceye sahip olduğu, orijinal çizgenin en büyük alt çizgesine atıfta bulunmaktadır. En büyük alt çizge oluşu, derece özelliğinin geçerli olduğu daha fazla düğüme sahip başka bir alt çizge bulmanın mümkün olmadığını garanti etmektedir.

i-çekirdek ile gösterilen alt çizgedeki düğümler, $j<i$ için *j*-çekirdek ile gösterilen alt çizgeye de aittir. Ancak bunun tersi doğru değildir. Her *u* düğümü, *u*'nun i-çekirdeğe ait olduğu en büyük *i* değerini gösteren bir çekirdek değeri ile ilişkilidir. En büyük çekirdek değeri, çizgenin dejeneresi (degeneracy) olarak adlandırılmaktadır. K-Çekirdek Ayrışımı, çizgedeki en düşük dereceye sahip düğümü iteratif olarak kaldırarak çizge boş kalana kadar devam etmektedir. Bir düğüm çizgeden çıkarıldığında, tüm ilişkileri de ortadan kaldırılmaktadır ve komşularının derecesi bir azalmaktadır. Bu yaklaşımla, farklı çekirdek grupları birer birer keşfedilmektedir [163-165]. Şekil 5.5'te K-Çekirdek Ayrışımının hesaplama adımları sunulmuştur.

```
1   fonksiyon kCekirdekAyristirmaHesapla(graph):
2       dereceler = {dugum: uzunluk(graph[dugum]) graph icindeki her dugum icin}
3       cekirdekDegerleri = {}
4       siraliDugumler = sirala(dereceler, artan=True)
5       her dugum icin siraliDugumler:
6           dugum = siradaki(siraliDugumler)
7           cekirdekDegerleri[dugum] = dereceler[dugum]
8           her komsu icin graph[dugum]:
9               dereceler[komsu] -= 1
10              graph[komsu].sil(dugum)
11          graph.sil(dugum)
12          siraliDugumler.sil(dugum)
13      return cekirdekDegerleri
```

Şekil 5.5. K-çekirdek ayrışımı hesaplama adımları

Önelikli olarak her düğümün derecesi hesaplanmaktadır ve bu değerler bir sözlükte saklanmaktadır. Bu sözlükte anahtar olarak düğümler, değer olarak ise düğümlerin dereceleri tutulmaktadır (satır 2). Her düğümün çekirdek değerini saklamak için boş bir

sözlük oluşturulmaktadır (satır 3). Düğümler derecelerine göre artan sırada sıralanmaktadır (satır 4). Sıralı düğümler listesinde her düğüm için döngü başlatılmaktadır (satır 5). En düşük dereceli düğüm seçilmektedir (satır 6). Seçilen düğümün çekirdek değeri güncellenmektedir (satır 7). Seçilen düğümün komşuları üzerinde dönmek üzere döngü oluşturulmaktadır (satır 8). Seçilen düğümün komşularının derecesi bir azaltılmaktadır (satır 9). Seçilen düğüm, komşularının komşuluk listesinden silinmektedir (satır 10). Seçilen düğüm hem çizgeden hem de sıralı düğümler listesinden silinmektedir (satır 11 - 12). Son olarak, fonksiyon her düğümün çekirdek değerini içeren sözlüğü döndürmektedir (satır 13).

Etiket yayılımı

Etiket Yayılımı algoritması, bir çizgede toplulukları bulmak için kullanılan hızlı bir algoritmadır. Toplulukları yalnızca ağ yapısını rehber olarak kullanarak tespit edebilmektedir ve önceden tanımlanmış bir amaç fonksiyonu veya topluluklar hakkında önceden bilgi sahibi olmayı gerektirmemektedir.

```
1  fonksiyon etiketYayilimiHesapla(graph, maksIterasyon):
2      etiketler = {dugum: graph icindeki her dugum icin}
3      her iterasyon icin 1'den maksIterasyon'a:
4          dugumSirasi = karistir(anahtarlar(etiketler))
5          her dugum icin dugumSirasi:
6              komsuEtiketleri = [etiketler[komsu]: graph[dugum] içindeki her komsu icin]
7              yeniEtiket = enCokGorulen(komsuEtiketleri)
8              etiketler[dugum] = yeniEtiket
9          eger tum dugumler icin etiketler[dugum] == yeniEtiket:
10             dur
11     return etiketler
12
13 fonksiyon enCokGorulen(liste):
14     sayac = {}
15     her oge icin liste:
16         eger oge sayac'da yoksa:
17             sayac[oge] = 0
18         sayac[oge] += 1
19     maksDeger = maksimum(sayac)
20     maksEtiketler = [etiket: sayac icindeki her etiket icin if sayac[etiket] == maksDeger]
21     return rastgeleSec(maksEtiketler)
```

Şekil 5.6. Etiket yayılımı hesaplama adımları

Etiket Yayılımı, etiketleri çizge boyunca yayarak ve bu etiket yayılım sürecine dayanarak topluluklar oluşturmaktadır. Algoritmanın arkasındaki fikir, tek bir etiketin yoğun bir şekilde bağlı düğüm grubunda hızla baskın hale gelebileceği, ancak seyrek bir şekilde bağlı

bir bölgeyi geçmekte zorlanacağıdır. Etiketler, yoğun bir şekilde bağlı düğüm grubunun içinde sıkışır ve algoritma tamamlandığında aynı etikete sahip olan düğümler aynı topluluğun parçası olarak kabul edilmektedir [166].

Algoritmanın çalışma şekli incelendiğinde; başlangıçta her düğüm benzersiz bir topluluk etiketi ile başlatılmaktadır. Bu etiketler ağ boyunca yayılmaktadır. Her yayılma iterasyonunda, her düğüm etiketi, komşularının en fazla sayısının ait olduğu etiketle güncellenmektedir. Etiket Yayılımı, her düğümün komşularının çoğunluk etiketine sahip olduğunda yakınsama sağlamaktadır. Etiket Yayılımı, ya yakınsama sağlandığında ya da kullanıcı tarafından tanımlanan maksimum iterasyon sayısına ulaşıldığında durmaktadır. Etiketler yayıldıkça, yoğun bir şekilde bağlı düğüm grupları hızla benzersiz bir etiket üzerinde uzlaşmaktadır. Yayılma sürecinin sonunda sadece birkaç etiket kalmaktadır ve çoğu yok olmaktadır. Süreç sonunda aynı topluluk etiketine sahip olan düğümlerin aynı topluluğa ait olduğuna karar verilmektedir [167, 168]. Şekil 5.6'da Etiket Yayılım algoritması hesaplama adımları sunulmaktadır.

Her düğüm benzersiz bir topluluk etiketi ile başlatılmaktadır. Başlangıçta her düğüm kendi özgün etiketine sahiptir (satır 2). Fonksiyonda girdi olarak belirtilen maksimum iterasyon sayısı kadar döngü başlatılmaktadır (satır 3). Düğümler rastgele bir sırayla işlenmektedir (satır 4). Rastgele sırayla her düğüm üzerinde döngü başlatılmaktadır (satır 5). Her düğümün komşularının etiketleri bir araya getirilmektedir (satır 6). En çok görülen komşu etiketi bulunmaktadır (satır 7). Bulunan etiket kullanılarak düğümün etiketi güncellenmektedir (satır 8). Bu noktada yakınsama kontrol edilmektedir yani, eğer tüm düğümler çoğunluk etiketine sahip ise döngü sonlandırılmaktadır (satır 9 - 10). Ayrıca bir listedeki en çok görülen öğeyi bulmak için yardımcı bir fonksiyona ihtiyaç duyulmaktadır (satır 13 - 21).

Leiden algoritması

Leiden algoritması, büyük ağlarda toplulukları tespit etmek için kullanılan bir algoritmadır. Algoritma, her topluluk için bir modülerlik skorunu en üst düzeye çıkarmak amacıyla düğümleri ayrık topluluklara ayırmaktadır. Modülerlik, düğümlerin topluluklara atanmasının kalitesini, yani bir topluluktaki düğümlerin ne kadar yoğun bir şekilde bağlantılı olduğunu, rastgele bir ağdaki bağlantılarına kıyasla ölçmektedir.

```
fonksiyon leiden_algoritmasi(graph, maksIterasyon):
    etiketler = {dugum: graph icindeki her dugum icin}
    oncekiEtiketler = {}
    her iterasyon icin 1'den maksIterasyon'a:
        her dugum icin graph:
            komsuEtiketleri = [etiketler[komsu] icin her komsu graph[dugum]]
            yeniEtiket = enYuksekmodulerlikArtisiHesapla(komsuEtiketleri, graph, dugum)
            etiketler[dugum] = yeniEtiket
        etiketler = toplulukIyilestir(etiketler, graph)
        yogunlasmisGraph = yogunlastirilmisGrafOlustur(etiketler, graph)
        graph = yogunlasmisGraph
        eger yakinsamaSaglamasiYap(oncekiEtiketler, etiketler):
            dur
        oncekiEtiketler = etiketler.copy()
    return etiketler
fonksiyon enYuksekmodulerlikArtisiHesapla(komsuEtiketleri, graph, dugum):
    enYuksekArtis = -sonsuz
    enIyiEtiket = komsuEtiketleri[0]
    her etiket icin benzersiz(komsuEtiketleri):
        modülerlikArtisi = modulerlikArtisiHesapla(graph, dugum, etiket)
        eger modülerlikArtisi > enYuksekArtis:
            enYuksekArtis = modülerlikArtisi
            enIyiEtiket = etiket
    return enIyiEtiket
fonksiyon toplulukIyilestir(etiketler, graph):
    yeniEtiketler = {}
    her topluluk icin benzersiz(etiketler):
        eger topluluk buyukse:
            kucukTopluluklar = toplulukRastgeleBol(topluluk)
            her kucukTopluluk icin:
                yeniEtiketler[kucukTopluluk] = kucukTopluluk
        aksi halde:
            yeniEtiketler[topluluk] = topluluk
    return yeniEtiketler
fonksiyon yogunlastirilmisGrafOlustur(etiketler, graph):
    yeniGraph = {}
    her etiket icin benzersiz(etiketler):
        yeniGraph[etiket] = komsulariBirlestir(etiket, graph, etiketler)
    return yeniGraph
fonksiyon yakinsamaSaglamasiYap(oncekiEtiketler, etiketler):
    her dugum icin etiketler:
        eger etiketler[dugum] != oncekiEtiketler[dugum]:
            return false
    return true
fonksiyon modulerlikArtisiHesapla(graph, dugum, etiket):
    m = toplamKenarSayisiHesapla(graph)
    mevcutEtiket = etiketler[dugum]
    mevcutKomsuSayisi = 0
    yeniKomsuSayisi = 0
    her komsu için graph[dugum]:
        eger etiketler[komsu] == mevcutEtiket:
            mevcutKomsuSayisi += 1
        eger etiketler[komsu] == etiket:
            yeniKomsuSayisi += 1
    delta_Q = (yeniKomsuSayisi - mevcutKomsuSayisi) / (2 * m)
    return delta_Q
```

Şekil 5.7. Leiden algoritması hesaplama adımları

Leiden algoritması, modülerliği optimize ederek toplulukları tek kümeye birleştiren ve bu işlemi yoğunlaştırılmış çizgede tekrar eden hiyerarşik bir kümeleme algoritmasıdır. Louvain algoritmasının bazı eksikliklerini gidermek için geliştirilmiştir. Bu eksikliklerin

başında, Louvain algoritması tarafından bulunan bazı toplulukların iyi bağlantılı olmaması gelmektedir. Bu sorun Leiden algoritmasında, toplulukları periyodik olarak rastgele daha küçük ve iyi bağlantılı olanlara ayırarak çözülmektedir. Leiden algoritmasının hesaplama basamakları Şekil 5.7'de sunulmaktadır.

İlk olarak *etiketler* adlı bir sözlük oluşturulmaktadır. Her düğüm başlangıçta kendine özgü bir topluluk etiketi ile başlatılmaktadır (satır 2). Adı *oncekiEtiketler* olan boş bir sözlük oluşturulmaktadır. Bu sözlük, önceki iterasyondaki etiketleri saklamak için kullanılmaktadır (satır 3). Belirtilen maksimum iterasyon sayısına kadar bir döngü başlatılmaktadır (satır 4 - 5). Mevcut düğümün komşularının etiketleri toplanmaktadır. *komsuEtiketleri* adında bir liste oluşturulmaktadır ve bu liste, mevcut düğümün komşularının etiketlerini içermektedir (satır 6). *enYuksekmodulerlikArtisiHesapla* fonksiyonu çağrılarak, mevcut düğüm için komşularının etiketlerinden en yüksek modülerlik artışını sağlayan etiket belirlenmektedir (satır 7). Belirlenen yeni etiket, mevcut düğümün etiketi olarak güncellenmektedir (satır 8). *toplulukIyilestir* fonksiyonu çağrılarak, büyük toplulukları daha küçük ve iyi bağlantılı topluluklara ayırma işlemi gerçekleştirilmektedir ve etiketler güncellenmektedir (satır 9). *yogunlastirilmisGrafOlustur* fonksiyonu çağrılarak, yoğunlaştırılmış çizge oluşturulur ve yogunlasmisGraph değişkenine atanır (satır 10). Yeni yoğunlaştırılmış çizge, mevcut çizge olarak güncellenmektedir (satır 11). *yakinsamaSaglamasiYap* fonksiyonu çağrılarak, önceki iterasyondaki etiketlerle mevcut etiketler karşılaştırılmaktadır. Eğer yakınsama sağlanmışsa, döngü durdurulmaktadır (satır 12 - 13). Mevcut etiketler, *oncekiEtiketler* olarak kopyalanmakta ve saklanmaktadır (satır 14). Fonksiyon, her düğümün topluluk etiketini içeren sözlüğü döndürmektedir (satır 15). *enYuksekmodulerlikArtisiHesapla* fonksiyonu bir düğümün komşularının etiketlerini dikkate alarak, o düğüm için modülerlik artışını en üst düzeye çıkaracak etiketi belirlemektedir (satır 16 - 24). *toplulukIyilestir* fonksiyonu mevcut toplulukları iyileştirerek daha küçük ve iyi bağlantılı alt topluluklara ayırmayı amaçlamaktadır (satır 25 - 34). *yogunlastirilmisGrafOlustur* fonksiyonu mevcut çizgeden yoğunlaştırılmış bir çizge oluşturmayı amaçlamaktadır (satır 35 - 39). *yakinsamaSaglamasiYap* fonksiyonu önceki iterasyondaki etiketlerle mevcut iterasyondaki etiketlerin aynı olup olmadığını kontrol etmektedir, yani algoritmanın yakınsama sağlayıp sağlamadığını belirlemektedir (satır 40 – 44). *modulerlikArtisiHesapla* fonksiyonu bir düğümün belirli bir etikete atanmasının modülerlikte ne kadar bir artış sağlayacağını hesaplamaktadır (satır 45 - 56).

5.2. Tahminleme Modelinin Eğitilmesi

Meta-yolların farkında olmak, varlıklar arasındaki karmaşık anlamsal ilişkileri ortaya çıkarmak amacıyla HBA'daki veri madenciliği ve ağ analizi işlemleri için önemlidir. Şekil 5.8'de, tahminleme modeli oluşturmak için önerilen çizge tabanlı makine öğrenimi hattı gösterilmektedir. Eğitilen model yardımıyla HBA üzerindeki düğümler arasında çeşitli ilişkiler tahmin edilmesi hedeflenmektedir.

Tahminleme modeli eğitimine başlanılan noktada, tezin dördüncü bölümü sonucunda elde edilen HBA, (A) adımınında çizge veri tabanında hazır bulunmaktadır. HBA üzerinde 48 farklı düğüm türü ve 25 farklı bağlantı türü bulunmaktadır. (B) adımında mevcuttaki HBA'nın tamamı belleğe aktarılmaktadır. Bu aktarım esnasında tüm düğümler ve bağlantılar korunmakta ve herhangi bir bilgi kaybı oluşmaması hedeflenmektedir. HBA'yı dosya sisteminden belleğe aktarmak, sonraki adımlarda yapılacak hesaplamaların performanslı bir şekilde yapılmasına olanak tanımaktadır. Bu aşamada HBA üzerindeki tüm düğümlere düğüm özellikleri eklenmektedir. Tahminlerin kalitesini artırmak için modelin doğru kararlar almasını sağlayacak bilgilendirici düğüm özelliklerinin kullanılması gerekmektedir. (B) adımında tüm düğüm özelliklerinin bellekteki çizgeye taşınmasıyla birlikte yeni düğüm özelliklerinin türetilmesi mümkün kılınmaktadır.

(C) adımında uygulanan işlemler, düğümler arasındaki ilişkilerin tahminlenmesi için oluşturulacak makine öğrenmesi modellerinin eğitim, test ve ilişki tahminleme süreçlerinde kullanılacak özellik kümesinin genişletilmesi içindir. Tez çalışmasında oluşturulan HBA'da düğümlerin sahip oldukları özellikler düğüm türleri arasında farklılık göstemektedir. En önemlisi, tez çalışmasında düğüm ve ilişki çeşitliliğinin artırılması için düğümlerin içindeki özellikler de birer düğüm türü haline getirilerek, düğümlerin içerisinden çıkarılmıştır. Bu sebeple düğümlerin içindeki özelliklerin sayısı oldukça azalmıştır. Düğüm özelliklerinin azalması, geliştirilecek tahminleyici makine öğrenmesi modellerinin tahminleme sürecine girdi parametresi olarak alabileceği özelliklerin kısıtlanmasına sebep olmuştur. Düğümlere ait özelliklerin artırılabilmesi için mevcut düğüm özelliklerinden anlamlı yeni özelliklerin türetilmesi hedeflenmiştir. Böylece her düğümün özellik kümesinin genişletilmesi sağlanmıştır.

Mevcut düğüm özelliklerinden yeni özellikler türetmek için literatürdeki düğüm yerleştirme yöntemleri incelenmiştir. Hızlı rastgele projeksiyon (fast random projection, FastRP) [169], graf örnekleme ve birleştirme (graph sample and aggregate, GraphSAGE) [175], Node2Vec [176] ve graf sinir ağları özetleme (hash graph neural networks, HashGNN) [177] algoritmaları incelenmiştir. Tez çalışmasında çeşitli sebeplerden dolayı FastRP algoritması kullanılarak düğüm yerleştirme işleminde kullanılmış ve tahminleme modeli için yeni düğüm özelliği üretilmiştir.

FastRP algoritmasının seçilme sebeplerine gelindiğinde, öncelikli olarak Node2Vec ve GraphSAGE algoritmalarından çok daha hızlı sonuca ulaşmaktadır. Büyük çizgeler için geliştirilmiştir ve çizge büyüdükçe algoritmanın zaman karmaşıklığı çizgisel (linear) olarak büyümektedir. Node2Vec'in, düğümlerin komşuluklarını keşfetmek için hesaplama açısından pahalı olan rastgele yürüyüşlere ihtiyaç duymasının aksine, FastRP doğrudan düğüm yerleştirmeleri hesaplamak için rastgele projeksiyonlar kullanmaktadır. Bu durum, düğüm yerleştirme sürecini basitleştirmekte ve hesaplama süresini azaltmaktadır.

GraphSAGE, Node2Vec ve HashGNN algoritmaları, devamlı güncellenen, gelişen, büyüyen, düğüm ve ilişki tür çeşitliliği artan çizgeler için, tekrarlı olarak modelleri eğitimi yapmak gerektiğinden yüksek zaman ve hesaplama kaynağı gerektirmektedir. Öte yandan FastRP, diğerlerine göre çok daha hızlı şekilde eğitim sürecinde düğüm yerleştirmeleri üretmektedir. Tez çalışmasında önerilen çerçeve model düzenli olaran HBA'ya yeni veriler entegre etmeyi gerektirdiği için düğümlerin özelliklerin zaman için çeşitlenebileceğine yönelik bir esneklik gözetilmiştir. FastRP algoritması bu noktada, düşük boyutlu düğüm yerleştirmeleri üretme kabiliyetiyle diğer algoritmaların önüne geçmiştir.

(C) adımında FastRP algoritması kullanılarak yeni bir düğüm özelliği üretilmektedir. Rastgele projeksiyonun teorik temeli, düğümler arasındaki ikili mesafeleri yaklaşık olarak korurken herhangi bir boyuttaki n vektörün, $O(log(n))$ boyutlarına yansıtılmasına dayanmaktadır. Bu yaklaşım, rastgele seçilen bir doğrusal projeksiyon tarafından karşılanmaktadır. FastRP algoritması, iki düğüm arasındaki mesafeyle ilgili bilgileri korurken boyutsallığın azaltılmasına olanak tanımaktadır. FastRP algoritması, düğümler ve komşuları arasındaki benzerliğin korunmasına odaklanan çizge tabanlı işlemler için kullanılmaktadır. Algoritma, rastgele oluşturulmuş vektörlerle başlamakta ve düğüm komşulukları üzerinden yinelemeli ortalama alma yoluyla, denklemdeki her bir düğüm için

bir dizi ara düğüm yerleştirme e_n^i (node embedding) üretmektedir (Eş. 5.1). Çizgeye ilişkin olarak, *m* değişkeni *n* düğümünün komşuları üzerinde yinelenir ve e_n^0 ise düğümün başlangıçtaki rastgele vektörünü temsil etmektedir. *ort* fonsiyonu ortalamayı hesaplamaktadır.

$$e_n^i = ort(e_m^{i-1}) \tag{5.1}$$

Herhangi bir n düğümüne ait e_n düğüm yerleştirme, daha önce tanımlanan düğüm vektör formlarının bir bileşimidir (Eş.5.2).

$$e_n = w_0 \cdot normalizeEt(r_n) + \sum_{i=1}^{i=k} w_i \cdot normalizeEt(e_n^i) \tag{5.2}$$

Bu bağlamda *normalizeEt* fonksiyonu bir vektörü L2 normuna bölmektedir [170]. Bir düğümün kendi kendine etkisi w_0 olarak gösterilmekte ve iterasyon ağırlıkları [w_1, w_2,..., w_k] ile temsil edilmektedir. Her düğümün düğüm yerleştirmesi, yineleme sayısına eşdeğer olan yarıçapa sahip bir komşuluktan etkilenmektedir. Bu yaklaşım sayesinde FastRP, yüksek ölçeklenebilirliği koruyarak çizge içindeki daha üst düzey ilişkilerden yararlanmaktadır.

Tezde, FastRP algoritması bir düğüm yerleştirme tekniği olarak kullanılmaktadır. FastRP algoritması, ağdaki düğümler ve komşuları arasındaki benzerliğin korunmasının önemli olduğu HBA'lar için elverişlidir. FastRP algoritması çalıştırıldıktan sonra, ağda benzer komşuluklara sahip düğümler benzer düğüm yerleştirme vektörlerine sahip olmaktadır. Tam tersi şekilde düşünüldüğünde, özellikleri ve komşuları tamamen farklı düğümlere benzer düğüm yerleştirme atanmamaktadır [169].

(C) adımında, FastRP algoritması kullanılarak, her düğüm için o düğümün özelliklerini ve komşuları ile olan ilişkilerini yansıtılan bir vektör içeren düğüm yerleştirme adlı ek bir özellik oluşturulmuştur. Yeni oluşturulan düğüm özelliği, bir sonraki adım (D)'de düğüm çiftleri arasında bağlantı vektörü oluşturmada kullanılan özelliklerden biridir. Tez çalışması kapsamında yapılan deneylerde, FastRP'nin düğüm yerleştirme adımı olarak kullanılmasının, tahminleme modelinin başarısına olan etkisi ölçülmüştür. FastRP ile düğüm yerleştirmenin uygulandığı ve uygulanmadığı iki senaryo işletilmiş ve elde edilen

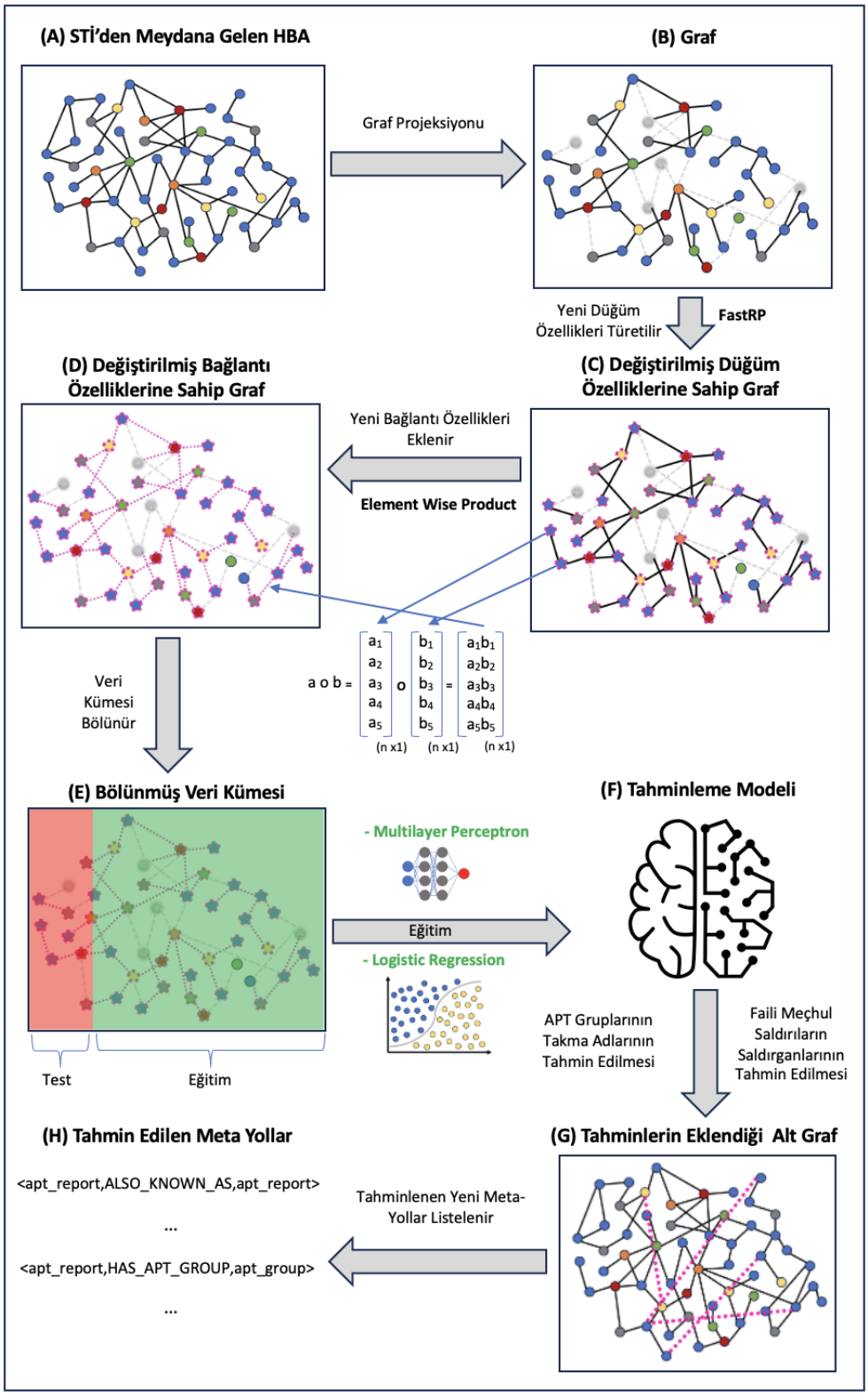
sonuçların karşılaştırması 6. Önerilen Çerçeve Modelin ve Tahminleme Modelinin Değerlendirilmesi başlığı altında sunulmuştur.

(D) adımında, bağlantı özellik vektörü oluşturma işlemi, geleneksel makine öğrenmesi yöntemlerinden farklı olarak uygulanan bir işlemdir. Bu noktada amaç, tekil düğümleri sınıflandırmak yerine düğüm çiftlerini sınıflandırmaktır. Her düğüm çifti için bir bağlantı özellik vektörü oluşturmak, iki düğümün özellik vektörlerinin kullanılmasını ve bunların bileşke bir vektör oluşturmak üzere bir araya getirilmesinden oluşmaktadır.

Bağlantı özelliği vektörü oluşturma işlemi, bireysel özelliklerine dayalı olarak bir çift düğüm için bir bağlantı vektörü oluşturma sürecidir. Öncelikle bu sürece hangi düğüm özelliklerinin dahil edileceğinin belirlenmesi gerekmektedir. Kullanılacak düğüm özellikleri olarak ‘embedding’, ‘time’, ‘weight’ ve ‘name’ seçilmektedir. ‘time’, dış kaynaklarda ilgili STİ’nin karşılaşıldığı zamanı temsil eder, düğüm yerleştirme düğümün vektör formundaki temsilini ifade eder ve ‘weight’ bir düğümün taşıdığı bilgi ile kaç kere karşılaşıldığını ifade eder, ‘name’ düğümün içerdiği GİG’in adını ifade etmektedir.

Düğümler arasındaki ilişkilerin tahmin edilmesinde yeni bağlantı özellikleri türetebilmek, bu bağlantı özelliklerini model eğitimi ve ilişki tahminlemede kullanmak için düğümler üzerinde üretilen düğüm yerleştirme vektörlerini kombine etmekte kullanılan yöntemler araştırılmıştır. İncelenen fonksiyonlar, eleman bazlı çarpım (element wise product, EWP), seviye 2 normu (level 2 norm, L2) ve kosinüs (cosine) uzaklığıdır. EWP diğer fonksiyonlardan farklı olarak iki vektörün çarpımında vektör boyutunu korumaktadır. Matematiksel olarak basit bir hesaplama yapıldığı için performans olarak diğer yöntemlerden hızlıdır. Düğüm yerleştirmeleri daha da geliştirmek için toplama veya normalizasyon gibi işlemlerle birleştirilebilmektedir.

Tez çalışmasında, birleştirici fonksiyon olarak EWP kullanılmıştır. [171]. Bir düğüm çifti olarak *(s,t)* ele alındığında, kaynak düğüm $s = [s_1, s_2, s_3, ..., s_n]$ ve hedef düğüm $t = [t_1, t_2, t_3 ..., t_n]$ şeklinde ifade edilmektedir. EWP fonksiyonu aşağıdaki şekilde ifade edilmektedir (Eş. 5.3).



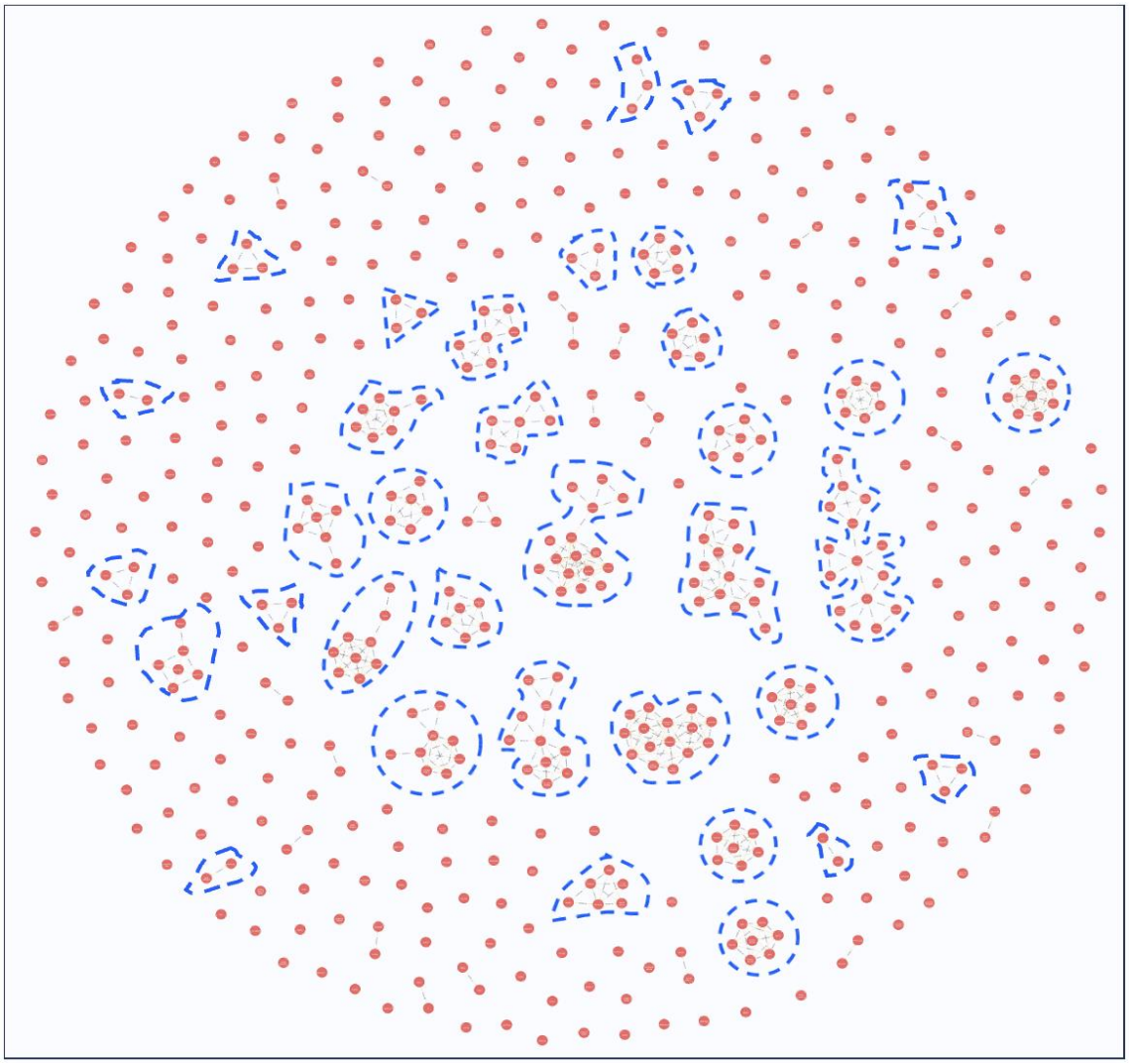
Şekil 5.8. Tahminleme modelini eğitmek için çizge tabanlı makine öğrenmesi süreci

$$f = [s_1 * t_1, s_2 * t_2, s_3 * t_3, \ldots, s_n * t_n] \quad (5.3)$$

Tez çalışmasında *s* ve *t* özellik vektörlerinin uzunluğu dörttür; [$s_{embedding}, s_{time}, s_{weight}, s_{name}$] ve t = [$t_{embedding}, t_{time}, t_{weight}, t_{name}$] şeklinde ifade edilmektedir. 'time' ve 'embedding'in de EWP'ye girdi olması gerektiği değerlendirilmiştir. Daha yüksek bir 'time' değerine sahip yeni bir STİ, tahminin yapılmasında daha yüksek bir etkiye sahip olmalıdır. Zaman geçtikçe her türlü STİ verisi, özellikle de teknik STİ, geçerliliğini kaybetmektedir. 'weight' HBA'daki eşsiz bir GİG ile veri toplama sürecinde kaç kere karşılaşıldığı bilgisini tutmaktadır. Düğümlerdeki 'name' değerinin EWP'ye dahil edilmesinin sebebi ise GİG'lerin sahip oldukları isimler arasında birbirlerini içerme ilişkisi bulunmasıdır. Örneğin bir URL başka bir alan adını içerebilir. Bir alt alan adı, başka bir alan adını içerebilir. Bir APT grup adı (Dragonok), başka bir APT grup adıyla (Hidden Dragon, Ghost Dragon, Night Dragon) kesişebilir. Bu benzerliklerin vektör uzayındaki pozisyonlarını etkilemesi gerektiği değerlendirilmiştir. EWP kullanarak düğümler arasındaki ilişkilere yeni özellik türetme işleminin uygulandığı ve uygulanmadığı iki senaryo işletilmiş ve elde edilen sonuçların karşılaştırması 6. Önerilen Çerçeve Modelin ve Tahminleme Modelinin Değerlendirilmesi başlığı altında sunulmuştur.

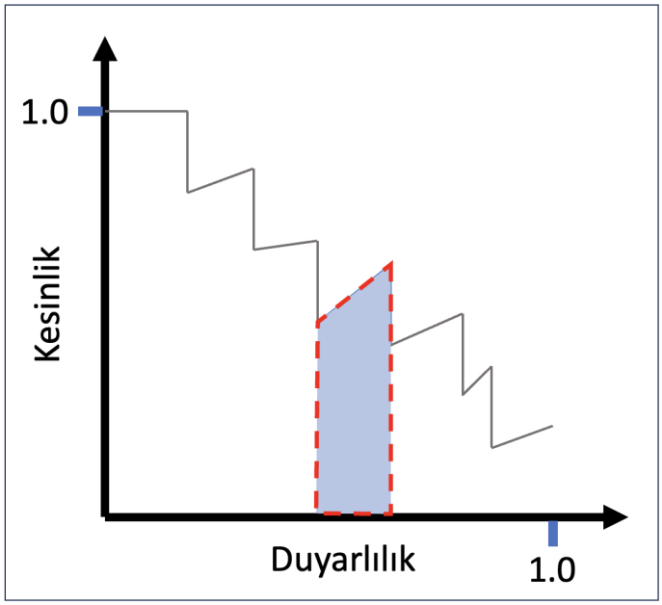
(E) adımında bellekteki çizge; test (%20) ve eğitim (%80) olarak iki kümeye bölünmektedir. Tekil düğümler yerine düğüm çiftlerinin sınıflandırıldığı göz önüne alındığında, çizgedeki her düğüm için çok sayıda kombinasyon *(|V| - 1)* ortaya çıkmaktadır. Tamamen bağlantılı bir çizge üzerinde çalışılmadığı sürece, her düğüm için herhangi bir kenarın olmadığı çok sayıda kombinasyonla *(|V| - 1 - derece(n))* karşılaşılmaktadır.

Verilerin bölünmesinin ardından (F) adımında Rastgele Orman (Random Forest), Çok Katmanlı Algılayıcı (Multilayer Perceptron) ve Lojistik Regresyon (Logistic Regression) algoritmaları uygulanarak çok sayıda aday model elde edilmektedir. Eğitim esnasında her aday model, en iyi model konfigürasyonunu bulmak için k-katlı çapraz doğrulamaya tabi tutulmaktadır. Aday modellerin karşılaştırılmasında değerlendirme ölçütü olarak kesinlik-duyarlılık eğrisi altındaki alan (area under the precision-recall curve, AUCPR), kullanılmakta ve kazanan model belirlenmektedir [172]. AUCPR, özellikle sınıf dağılımının dengesiz olduğu durumlarda, ikili sınıflandırma modelinin etkinliğini değerlendirmek için kullanılmaktadır. Tez çalışmasında oluşan HBA'da, komşuları olan APT grubu düğümleri (pozitifler), komşuları olmayan APT grubu düğümlerinden (negatifler) daha azdır. APT gruplarının denge durumu Şekil 5.9'de sunulmaktadır.



Şekil 5.9. HBA'daki APT grubu düğümlerinin pozitif ve negatif çift dağılımları

AUCPR, kesinlik (precision) – duyarlılık (recall) eğrisinden türetilen bir ölçüttür. Hesaplama yöntemi incelendiğinde; üretilen tahminlerin tümü, tahmin ile ilişkili bir tahmin olasılık puanına sahip olmaktadır. Oluşturulan tahminler olasılık puanlarına göre azalan şekilde sıralanmaktadır. Modellere ait tahmin olasılık puanları hesaplandıkça, bu olasılık eşik değer olarak belirlenmektedir. Bu olasılığa ve daha fazlasına sahip olan tüm düğüm çiftleri pozitif olarak etiketlenmektedir. Bu noktada kesinlik ve duyarlılık hesaplanmakta ve bu değerler kesinlik-duyarlılık eğrisi üzerinde işaretlenmektedir. Son olarak eğri doğrusal enterpolasyona tabi tutulmakta ve noktaların birleştirilmesiyle oluşturulan yamuk (trapezoid) şeklin içindeki alan hesaplanmaktadır. Şekil 5.10'da örnek bir trapezoid taranarak gösterilmiştir.



Şekil 5.10. Kesinlik ve duyarlılık eğrisi altına oluşan bir trapezoidin alanı

AUCPR, çeşitli sınıflandırma eşiklerindeki ortalama kesinliği (precision) temsil etmektedir. Birçok çizge, bağlantılı olanlara kıyasla daha fazla sayıda komşu olmayan düğüm çifti içermektedir ve bu durum seyreklik (sparsity) olarak bilinmektedir. Sonuç olarak çizge veri yapılarında sınıf dengesizliği, yani pozitif düğüm sayısı ile negatif düğüm sayısı arasında orantısızlık, ortak bir sorun haline gelmektedir. Farklı veri senaryoları için AUCPR ve alıcı işletim karakteristiği - eğri altındaki alan (receiver operating characteristic - area under the curve, ROC-AUC) ölçüm yöntemleri mevcuttur. Tez çalışmasında azalan sırada olasılık puanına sahip bağlantı tahminlerine odaklanılmaktadır. AUCPR, ROC-AUC'den farklı olarak dengesiz dağılımlı veri kümelerinde daha yüksek başarım sağlamakta ve daha yüksek performansla çalışmaktadır [173].

6. ÖNERİLEN ÇERÇEVE MODELİN VE TAHMİNLEME MODELİNİN DEĞERLENDİRİLMESİ

Farklı siber güvenlik organizasyonları benzer profillere sahip saldırı gruplarını farklı takma adlar kullanarak etiketlemektedir. Örneğin İsrail orijinli olduğu tahmin edilen "SunFlower" saldırı grubu, "MoonFlower" ve "Cheshire Cat" isimleriyle de bilinmektedir. APT gruplarının tekilleştirilmesi, tehdit aktörlerinin takma adlarının tespiti, APT gruplarının profillenmesi karmaşık ve zorlu bir süreçtir [174] ve bu tez çalışmasında önemli görülmektedir.

APT niteliğindeki saldırılara ilişkin APT analiz raporlarının ayrıştırılmasıyla elde edilen GİG'lerin bir çizge veritabanına kaydedilmesi ve GİG'ler arasındaki ilişkilerin ontolojik olarak incelenerek ortaya çıkarılması sonucunda, ilişkilerin de çizge veri tabanına işlenmesi birçok önemli avantaj sağlamaktadır. Tezin bu bölümünde ilk olarak HBA üretilimden kaynaklı elde edilen faydalar değerlendirilmektedir. Ardından önerilen modeller neticesinde ulaşılan sonuçlar değerlendirilmektedir.

Gelişmiş ilişki analizi: Çizge, APT saldırılarıyla ilgili GİG'ler arasındaki karmaşık ilişkileri ortaya çıkarmak ve analiz etmek için güçlü bir veri yapısıdır. Tez çalışmasında farklı GİG'ler arasında bağlantı kurma becerisinden yararlanarak APT saldırılarına ilişkin kapsamlı bir perspektif sunulmuştur. HBA üzerinde çizge tabanlı merkezilik analizleri ve topluluk tespiti analizleri yapılmıştır. APT grupları ile faili meçhul APT saldırıları arasındaki bağlantılar tahmin edilmiştir. Ayrıca APT grupları arasında takma ad ilişkileri tahmin edilmiştir.

Sorgu performansı: İlişkisel veritabanlarının aksine, çizge veri yapıları birbiri ile bağlantılı veriler üzerinde çalışmak üzere özelleşmiştir. Düğümler ve bağlantılardan oluşan bu tür veriler için ilişkisel veritabanlarına görece üstün sorgu performansı sunmaktadır. Bu da tez çalışması kapsamında hızlı tehdit algılama ve aksiyon planı geliştirme imkânı elde etmek anlamına gelmektedir.

Esneklik: Çizge veri yapıları, ilişkisel veritabanları gibi katı bir şema kurallarına uyum sağlamayı gerektirmemektedir. Böylece önerilen modellerin değişen ve gelişen siber tehditlere uyum sağlamasına olanak tanımaktadır. Ortaya çıkan HBA, şema

gerektirmemesi sebebiyle yeni GİG'lerin ağa dahil edilmesi konusunda yüksek esnekliğe sahiptir.

Proaktif tehdit avcılığı: Tez çalışması kapsamında önerilen çerçeve model sayesinde GİG'ler arasındaki ilişkileri ortaya çıkararak, planlı bir saldırının göstergesi olan örüntüleri tanımak mümkündür. Organizasyonların ağlarına sızmış tehdit aktörlerinin bıraktığı izler, üretilen HBA üzerinde sorgulanarak tehdit avcılığına dair ip uçlarına dönüştürmek mümkündür. Henüz saldırılar başarıya ulaşmadan, tehdit avcılığında proaktif bir yaklaşım sergilemek mümkün kılınmaktadır.

Tehdit aktörü ilişkilendirme: Tez çalışmasında, GİG'ler arasındaki bağlantıları tanımlamak ve analiz etmek için çizge veri yapısı kullanılması, tehdit aktörlerinin TTP'leri hakkında geleneksel yöntemlere göre daha bağlamsal içgörü geliştirilmesini sağlamaktadır. Bu, saldırıların belirli tehdit gruplarına atfedilmesine yardımcı olmakta ve daha geniş kapsamlı STİ analizlerine katkıda bulunmaktadır.

Beşinci bölümdeki üğümlerin Merkezilik Analizi ve Topluluk Tespiti Analizi başlıklarında detaylı tarif edilen, merkezilik ve topluluk oluşturma algoritmaları bu bölümde uygulanarak sonuçlar incelenmektedir. Mevcut HBA üzerindeki düğümler ve ilişkiler üzerinde analizler yapılarak, elde edilen metrikler sunulmaktadır.

Aradalık merkeziliği: Aradalık Merkeziliği, STİ'den meydana gelen HBA'da siber saldırıların organize edilmesinde, ağa hükmeden, saldırılar arasında köprü görevi gören, saldırılar arasında bilgi akışı sağlayacak potansiyele sahip APT gruplarının ortaya çıkarılması için bir ölçüt olarak ele alınmıştır. Ağ üzerindeki apt_group etiketli düğümlerin tamamı üzerinde Aradalık Merkeziliği hesaplandığında ağın tamamına dair elde edilen skorlar Çizelge 6.1'de sunulmuştur.

Çizelge 6.1. HBA üzerindeki tüm düğümlerin Aradalık Merkeziliği istatistiği

Minimum Skor	Maksimum Skor	Ortalama Skor
0.0	49.4999	0.8545

APT Grubu bazlı olarak hesaplanan en yüksek 20 düğüme ait Aradalık Merkeziliği skorları ise azalan sırada Çizelge 6.2'de sunulmuştur.

Çizelge 6.2. HBA üzerindeki düğümlerin Aradalık Merkeziliği skorları

APT Grubu Adları	Aradalık Merkeziliği Skoru
phosphorus	49.4999
tsar team	42.0
charming kitten	35.5833
apt28	32.25
32.25	28.3333
strontium	24.6666
apt29	23.5
fancy bear	23.1666
sofacy	22.75
apt37	18.5
lazarus	18.0
apt42	15.0
apt1	12.0
cozyduke	11.5
sandworm	11.0
itg18	10.25
shuckworm	7.0
sednit	6.7499
cozy bear, rocket kitten, armageddon, gamaredon	6.0
apt17	5.0

Yakınlık merkeziliği: Yakınlık Merkeziliği, STİ'den meydana gelen nihai HBA üzerinde APT grupları arasındaki etkileşimi artırma eğiliminde olan aracı APT gruplarını saptamak için bir yöntem olarak kullanılmıştır. Yakınlık Merkeziliği hesaplandığında elde edilen skorlar Çizelge 6.3'te sunulmuştur.

Çizelge 6.3. HBA üzerindeki tüm düğümlerin Yakınlık Merkeziliği istatistiği

Minimum Skor	Maksimum Skor	Ortalama Skor
0.0	1.0000	0.3139

APT grubu bazlı olarak hesaplanan en yüksek 20 değere ait Yakınlık Merkeziliği skorları ve ilgili APT grubu adları ise azalan sırada Çizelge 6.4'de sunulmuştur.

Çizelge 6.4. HBA üzerindeki APT gruplarının Yakınlık Merkeziliği skorları

APT Grubu Adı	Yakınlık Merkeziliği Skoru
apt1, kimsuky, transparent tribe, apt36, projectm,…	1.0000
winnti group	0.8571
shuckworm	0.8333
phosphorus	0.8181
mythic leopard, dnscalc, ta471, konni group	0.8
sofacy, dustman, dyncalc, pupa, unc2589, …	0.75

Çizelge 6.4. (devam) HBA üzerindeki APT gruplarının Yakınlık Merkeziliği skorları

APT Grubu Adı	Yakınlık Merkeziliği Skoru
superhard	0.7272
apt29, apt37, bluenoroff, turla, newscaster, …	0.7143
dota, cozybear	0.7
cobalt illusion	0.6923
uglygorilla, voodoo bear, uxeshe, apt-t-c-oo, apt19,…	0.6666
mint sandstor, clearsky	0.6470
bronze atlas	0.6363
unit 61398, yellow garuda, certfa, cozy bear, cozyduke,…	0.625
Yttrium	0.6154
strontium, apt10, cryptomimic, hidden cobra,…	0.6
stellarparticle	0.5833
armageddon	0.5555
the dukes, stardust chollima, aurora panda	0.5454
office monkeys	0.5333

Derece merkeziliği: Derece Merkeziliği, STİ'den meydana gelen HBA üzerindeki en çok bağlantıya sahip APT gruplarının ortaya çıkarılması, popüler ve muhtemelen en çok bilgiye sahip tehdit aktörlerinin belirlenmesi için kullanılmıştır. Yüksek Derece Merkeziliğine sahip APT grupları daha hızlı şekilde ağdaki bağlamsallığı artırabilecek potansiyele sahiptir. Çünkü diğerlerinden çok daha fazla ALSO_KNOWN_AS ilişkisine sahip olan bir APT Grubunun ağdaki diğer düğümler ile ilişki kurma potansiyeli ve imkânı daha yüksektir. Çok sayıda ALSO_KNOWN_AS ilişkisine sahip bir APT grubu, aynı anda birçok koldan ağ üzerinde yayılması, tekil bir APT grubuna göre daha elverişlidir. Derece Merkeziliği hesaplandığında elde edilen skorlar Çizelge 6.5'te sunulmuştur.

Çizelge 6.5. HBA üzerindeki APT gruplarının Derece Merkeziliği istatistiği

Minimum Skor	Maksimum Skor	Ortalama Skor
0.0	10.0	0.8599

APT grubu bazlı olarak hesaplanan en yüksek 10 skora ait Derece Merkeziliği skorları ve ilgili APT grubu adları ise azalan sırada Çizelge 6.6'da sunulmuştur.

Çizelge 6.6. HBA üzerindeki düğümlerin Derece Merkeziliği skorları

APT Grubu Adı	Derece Merkeziliği Skoru
charming kitten	10.0
apt28	9.0
apt42	8.0
phosphorus, bluecharlie, apt29	7.0

Çizelge 6.6. (devam) HBA üzerindeki düğümlerin Derece Merkeziliği skorları

APT Grubu Adı	Derece Merkeziliği Skoru
tag-53, snake, shell_crew, fin7, deep panda, chessmaster, apt35	6.0
unc788, transparent tribe, russia-nexus,muddywater, …	5.0
trident ursa, stonefly, sednit, pawn storm, fancy bear, …	4.0
wicked panda, webmasters, venomous bear, uac-0056, …	3.0
zerocleare, winnti, wild neutron, uxeshe, uroburos,…	2.0
yellow garuda, winnti group, websiic, waterbug, vovan, …	1.0

K-çekirdek ayrışımı: Mevcut HBA üzerindeki APT grupları arasında ne derece bağlantılılık olduğunu ortaya çıkarmak için K-Çekirdek algoritmasından yararlanılmaktadır. Mevcut HBA'da APT grupları arasındaki tek ve doğrudan ilişki türü ALSO_KNOWN_AS'dir. Bunun haricinde APT grupları arasında dolaylı bağlantılar bulunmaktadır. K-Çekirdek kullanılarak kendi aralarında bağlantılı küçük APT grubu topluluklarının belirlenmesinin mümkün olduğu değerlendirilmektedir.

APT grubu düğümleri arasındaki bağlara ait ilişki ağırlık değeri olan rel_weight özelliği kullanılarak yapılan hesaplamalar sonucunda çizgedeki en büyük çekirdek değeri 7 olarak bulunmuştur. Bu kapsamda elde edilen topluluklara atanan APT grupları Çizelge 6.7'de sunulmuştur.

Çizelge 6.7. HBA üzerindeki APT Gruplarının k-çekirdek kullanılarak kümelenmesi

APT Grubu Adları	Çekirdek Değeri
bluecharlie, calisto, callisto, coldriver, russia-nexus, seaborgium, …	7
apt 10, apt28, apt35, apt42, certfa, charming kitten, chessmaster, clearsky, cobalt illusion, …	6
anunak, apt-c-6, apt1, apt17, apt36, apt87, byzantine candor,…	5
apt3, blackmine, buckeye, darkseoul, snake, stonefly, tg-0110, …	4
ajax, ajax security team, apt-c-00, apt12, apt27, apt29, apt2, apt8, …	3
ajaxtm, apt-c-3, apt-t-c-oo, apt, apt34, apt37, apt43, arid viper,…	2
apt 30, apt-c-08, apt-c-36, apt-c-37, apt0, apt9, apt30, aurora panda,...	1

Etiket Yayılımı: HBA üzerinde kendi aralarında ilişkili APT grubu kümeleri arasında meydana gelecek daha geniş kümelerin tespiti için Etiket Yayılımı algoritmasından yararlanılmaktadır. Yüksek derecede ortak komşuluklara sahip kümelerin belirlenmesi, birlikte hareket ettiği düşünülen APT gruplarının ortaya çıkarılmasında kullanılmıştır. Etiket Yayılımı hesaplandığında elde edilen istatistik Çizelge 6.8'de sunulmuştur.

Çizelge 6.8. HBA üzerinde Etiket Yayılımı hesaplamasına ait istatistikler

İterasyon Sayısı	Yakınsama Durumu	Topluluk Sayısı
10	Olumsuz	364

En çok sayıda APT grubunun dahil olduğu yüksek sayıda üyesi olan 10 topluluğa ait bilgiler Çizelge 6.9'da sunulmuştur.

Çizelge 6.9. HBA üzerindeki APT gruplarının Etiket Yayılımı kümelenmesi

APT Grubu Adları	Topluluk
apt1, byzantinecandor, commentcrew, dota, launit61398, plaunit61398, superhard, uglygorilla	1
apt28, black energy, fancy bear, forest blizzard, group74, pawn storm, sandworm, sednit, sofacy, sofacy bear, tsar team, …	2
ajax, apt35, apt42, charming kitten, cobalt illusion, itg18, …	3
apt-c-56, apt36, apt87, c-major, projectm, transparent tribe, …	4
apt29, cozy bear, cozybear, cozycar, cozyduke, dukes, nobelium,…	5
apt37, hermit, hidden cobra, inkysquid, lazarus, redeyes, scarcruft, …	6
bluecharlie, calisto, callisto, coldriver, russia-nexus, seaborgium, …	7
blackmine, dark seoul, darkseoul, operation troy, silent chollima, …	8
earth vetala, mercury, muddywater, seedworm, static kitten, …	9
apt17, aurora panda, barium, gref, lead, passcv, wicked panda, …	10

Leiden Algoritması

Leiden algoritması kullanılarak, HBA üzerinde aralarında güçlü bağlar bulunan APT gruplarından meydana gelen toplulukların belirlenmesi amaçlanmıştır. Leiden algoritmasına ait hesaplamalardan elde edilen değerler Çizelge 6.10'da sunulmuştur.

Çizelge 6.10. HBA üzerinde Leiden algoritmasına ait istatistikler

İterasyon Sayısı	Yakınsama Durumu	Topluluk Sayısı
2	Olumlu	336

En çok sayıda APT grubunun dahil olduğu 10 topluluğa ait bilgiler Çizelge 6.11'da sunulmuştur.

Çizelge 6.11. HBA üzerindeki APT gruplarının Leiden kümelenmesi

APT Grubu Adları	Topluluk
rocket kitten, ajaxtm, flying kitten, operation saffron rose, …	1
turla, snake, uroboros, venomous bear, group 88, uroburos, …	2
bronze union, emissary panda, apt27, luckymouse, emissarypanda, …	3
armageddon, gamaredon, primitive bear, trident ursa, uac-0010, …	4
uac-0056, saintbear, unc2589, ta471, nodaria, …	5

Çizelge 6.12. (devam) HBA üzerindeki APT gruplarının Leiden kümelenmesi

APT Grubu Adları	Topluluk
gothic panda, apt3, ups, buckeye, ups team, tg-0110, …	6
deep panda, black vine, pupa, shell_crew, deeppanda, webmasters, …	7
byzantine candor, apt1, la unit 61398, uglygorilla, dota, superhard, ...	8
apt29, the dukes, cozybear, yttrium, midnight blizzard, bluebravo, …	9
kimsuky, apt43, thallium, velvet chollima, ta406, konni group, …	10

Tezin giriş bölümünde de belirtildiği gibi, STİ zenginleştirmesinin APT saldırılarının tehdit aktörleriyle ilişkilendirmesinde ne kadar önemli olduğunu ortaya çıkarmak için iki farklı senaryo çalışılmıştır. İki farklı senaryoya uygun olarak oluşturulan HBA'lara yönelik üretilen tahminleyici modeller kıyaslanmaktadır. Model adayları AUCPR ölçütüne göre karşılaştırılmaktadır ve Çizelge 6.12'de sunulmaktadır.

Çizelge 6.13. ALSO_KNOWN_AS ilişkisi tahmin modeli adaylarının AUCPR skoru

Senaryo	FastRP	EWP	Algoritma	Eğitim Skoru	Test Skoru
Senaryo 1	✓	✓	Random Forest	0.6436	0.6365
			Multilayer Perceptron	0.7545	0.7918
			Logistic Regression	0.7847	0.8136
	✓	×	Random Forest	0.5880	0.6017
			Multilayer Perceptron	0.6749	0.6812
			Logistic Regression	0.7767	0.7753
	×	✓	Random Forest	0.6068	0.6233
			Multilayer Perceptron	0.7357	0.7430
			Logistic Regression	0.7798	0.7804
Senaryo 2	✓	✓	Random Forest	0.8012	0.8339
			Multilayer Perceptron	0.9515	0.9446
			Logistic Regression	0.9636	0.9788
	✓	×	Random Forest	0.7857	0.7981
			Multilayer Perceptron	0.9407	0.9378
			Logistic Regression	0.9561	0.9524
	×	✓	Random Forest	0.7885	0.7976
			Multilayer Perceptron	0.9455	0.9494
			Logistic Regression	0.9604	0.9630

En yüksek performansa sahip model Lojistik Regresyon kullanılarak eğitilmiştir. Verinin %20'si test, %80'i eğitim amaçlı kullanılmış ve 10 katlı çapraz doğrulama uygulanmıştır. AUCPR ölçümleri incelendiğinde eğitim skoru = %96,36 ve test skoru = %97,88 olarak hesaplanmıştır. AUCPR skoru, kesinlik ve duyarlılık eğrilerinin kesişiminin altında kalan alanı ifade ettiği için, bu skoru hesaplamakta kullanılan bir adet kesinlik ve duyarlılık değeri söz konusu değildir. Tez çalışmasında sunulan doğruluk, kesinlik, duyarlılık ve F1-Skoru değerleri alanın

hesaplanmasındaki en yüksek noktalara ait değerlerdir. Logistic Regression algoritması kullanılarak elde edilmiş en yüksek başarıya sahip model için sunulmuş AUCPR skorunun hesaplanmasında kullanılan en yüksek doğruluk = 0.9863, kesinlik = 0.9789, duyarlılık = 0.9216 ve F1-Skoru = 0.9494 şeklindedir. Şekil 5.2'deki (G) adımında, eğitimden geçen model, yeni ilişkiler oluşturmak için bir bellek içi çizge üzerinde kullanılmıştır. Yeni oluşturulan ilişkiler üzerinde yeni bir özellik daha oluşturulmuştur. Olasılık adı verilen bu yeni özellik, üretilen tahminin gerçekleşme olasılığını temsil etmektedir.

Lojistik Regresyon algoritması üretilen modelin tahmin fonksiyonunu benzerlik ölçütü olarak kullanmaktadır ve bunu yaparken çalışma süresini kısaltmak ile doğruluğu artırmak arasında bir denge kurmayı sağlamaktadır. Bu bağlamda doğruluk, arama yoluyla elde edilebilecek en iyi tahminleri temsil eden ve modeller tarafından tahmin edilen optimal olası bağlantılara sonucun yakınlığıyla ilişkilidir. Her düğüm için bağlantıların ilk seçimi rastgeledir ve daha sonra önceden tahmin edilen bağlantılara dayalı olarak birden fazla yineleme yoluyla iyileştirme yapılmaktadır. İyileştirme işleminden sonra üretilen tahminler büyükten küçüğe doğru sıralanmaktadır. Elde edilen APT grupları arasında en yüksek olasılığa sahip ALSO_KNOWN_AS ilişkileri Çizelde 6.13'te sunulmaktadır.

Çizelge 6.14. ALSO_KNOWN_AS ilişkilerini gösteren ilk 20 tahmin

APT Grubu	APT Grubu	Olasılık Skoru
tsar team	group-4127	0.86281486
sednit	group 74	0.86280345
pawn storm	group 74	0.86275330
apt-c-56	mythic leopard	0.86270876
mythic leopard	c-major	0.86265844
calcium	sekur	0.86260804
gold niagara	sekur	0.86250445
pawn storm	sourface	0.86250003
cobalt illusion	ajax	0.86243249
strontium	sourface	0.86240224
group 74	group-4127	0.86230205
mint sandstorm	apt35	0.86220094
yellow garuda	apt35	0.86210003
ajax	newscaster	0.86198999
ajax	ta453	0.86191896
mint sandstorm	itg18	0.86184341
yellow garuda	itg18	0.86170235
mint sandstorm	unc788	0.86162572
yellow garuda	unc788	0.86156715
turla	venomous bear	0.86152699

Diğer bir tahminleme işlemi olarak, faili meçhul APT saldırılarının tehdit aktörleri tahmin edilmiştir. NER işlemi sonrasında 734 APT raporunun 232'unda tehdit aktörü bulunmadığı tespit edilmiştir. Geliştirilen ilk tahminleme modeli apt_group düğümleri arasında ALSO_KNOWN_AS ilişkileri tahminlemek için eğitilmiş ve en başarılı model adayı Lojistik Regresyon algoritması ile üretilmiştir. Bu noktada apt_report ve apt_group düğümleri arasında HAS_APT_GROUP ilişkilerinin tahminlenmesi hedeflenmiştir. Yeni <apt_report, HAS_APT_GROUP, apt_group> üçlüleri üretmek için düğüm türleri ve ilişki türünü hedefleyen yeni bir model eğitilmiştir. Üretilen adaylar arasında en başarılı model Çok Katmanlı Algılayıcılar (Multi-layer Perceptron) algoritması ile üretilmiştir.

HBA'nın %20'i test, %80'i eğitim için kullanılmış ve 10 katlı çapraz doğrulama uygulanmıştır. AUCPR ölçümleri incelendiğinde eğitim skoru = %77,35 ve test skoru = %76,65 olarak gözlemlenmektedir. AUCPR skoru, kesinlik ve duyarlılık eğrilerinin kesişiminin altında kalan alanı ifade ettiği için, bu skoru hesaplamakta kullanılan bir adet kesinlik ve duyarlılık değeri söz konusu değildir. Tez çalışmasında sunulan doğruluk, kesinlik, duyarlılık ve F1-Skoru değerleri alanın hesaplanmasındaki en yüksek noktalara ait değerlerdir. Multilayer Perceptron algoritması kullanılarak elde edilmiş en yüksek başarıya sahip model için sunulmuş AUCPR skorunun hesaplanmasında kullanılan doğruluk = 0.9378, kesinlik = 0.8235, duyarlılık = 0.7152 ve F1 skoru = 0.7655 şeklindedir. İki farklı senaryoya uygun olarak oluşturulan HBA'lara yönelik üretilen tahmin modelleri kıyaslanmıştır. Model adayları AUCPR ölçütüne göre karşılaştırılmaktadır ve Çizelge 6.14'de sunulmaktadır. ALSO_KNOWN_AS ilişkilerinin tahmin edilmesinden farklı olarak, HAS_APT_GROUP ilişkilerini tahmin etmede Çok Katmanlı Algılayıcılar az da olsa Lojistik Regresyondan daha iyi AUCPR skoru elde etmiştir.

Çizelge 6.15. HAS_APT_GROUP ilişkisi tahmin modeli adaylarının AUCPR skoru

Senaryo	FastRP	EWP	Algoritma	Eğitim Skoru	Test Skoru
Senaryo 1	✓	✓	Random Forest	0.6175	0.6238
			Multilayer Perceptron	0.6634	0.6646
			Logistic Regression	0.6456	0.6577
	✓	×	Random Forest	0.5578	0.5697
			Multilayer Perceptron	0.5956	0.6079
			Logistic Regression	0.5811	0.5780
	×	✓	Random Forest	0.6023	0.6142
			Multilayer Perceptron	0.6485	0.6491
			Logistic Regression	0.6355	0.6269

Çizelge 6.14. (devam) HAS_APT_GROUP ilişkisi tahmin modeli adaylarının AUCPR skoru

Senaryo	FastRP	EWP	Algoritma	Eğitim Skoru	Test Skoru
Senaryo 2	✓	✓	Random Forest	0.7233	0.7298
			Multilayer Perceptron	0.7735	0.7665
			Logistic Regression	0.7514	0.7653
	✓	×	Random Forest	0.6813	0.6867
			Multilayer Perceptron	0.7270	0.7279
			Logistic Regression	0.7444	0.7431
	×	✓	Random Forest	0.7032	0.7094
			Multilayer Perceptron	0.7468	0.7573
			Logistic Regression	0.7482	0.7529

STİ zenginleştirme işlemleri HAS_APT_GROUP ilişkilerinin tahminlenmesinde de başarısını kanıtlamıştır. Zenginleştirme süreci, STİ'den meydana gelen HBA'nın bağlamsallığının artmasını sağlamıştır. Oluşturulan HBA; düğüm sayısı, düğüm türü, ilişki sayısı ve ilişki türü açısından genişledikçe makine öğrenimi algoritmaları daha başarılı performans gösterdiği gözlemlenmiştir. Çizelge 6.15'te APT raporları ve APT grupları arasında kurulabilecek ilişkiye ait olasılıklar listelenmektedir.

Çizelge 6.16. HAS_APT_GROUP ilişkilerini gösteren ilk 20 tahmin

APT Analiz Raporu	APT Grubu	Olasılık Skoru
HPSR_SecurityBriefing_Episode16_NorthKorea.pdf	Kimsuky	0.6945
plugx-goes-to-the-registry-and-india.pdf	Aurora Panda	0.6936
PaloAlto_T9000-Advanced-Modular-Backdoor-Uses-Complex-Anti-Analysis-Techniques(Feb-04-16).pdf	ta459	0.6919
PaloAlto_T9000-Advanced-Modular-Backdoor-Uses-Complex-Anti-Analysis-Techniques(Feb-04-16).pdf	lexus	0.6902
Sentinelone_AcidRain-Wiper-Rains-Down-Europe(03-31-2022).pdf	Sandworm	0.6899
P2P_PlugX_Analysis.pdf	Aurora Panda	0.6886
tr-12-circl-plugx-analysis-v1.pdf	Aurora Panda	0.6878
DellSecureworks_ThreatGroup-4127-Targets-Clinton-Campaign(06-16-2016).pdf	bronze union	0.6871
Appendix_Cyclops_Blink_Sets_Sights_on_ASUS_Routers.pdf	gano sec team	0.6862
ArborNetworks_Additional-Insights-Shamoon2(02-21-2017).pdf.iocs	gano sec team	0.6853
Democracy_HongKong_Under_Attack.pdf	bronze union	0.6844
DrWeb-telecom_research_en(03-24-2022).pdf	bronze union	0.6840
Group-ib_DarkPink(01-11-2023).pdf	ruag	0.6839
Group72_Opening_ZxShell.pdf	ruag	0.6831
C5_APT_SKHack.pd	gano sec team	0.6825

Çizelge 6.15. (devam) HAS_APT_GROUP ilişkilerini gösteren ilk 20 tahmin

APT Analiz Raporu	APT Grubu	Olasılık Skoru
PaloAlto_The-Blockbuster-Sequel(04-07-2017).pdf	unc2589	0.6820
PaloAlto_The-Blockbuster-Sequel(04-07-2017).pdf	tg-0110	0.6812
Citizenlab_NilePhish-Large-Scale-Targeting-Egyptian(02-02-2017).pdf	gano sec team	0.6798
Lookout_Dark-Caracal_srr_20180118_us_v.1.0(01-18-2018).pdf	ruag	0.6786
PaloAlto_The-Blockbuster-Sequel(04-07-2017).pdf	zombie zero	0.6774

7. SONUÇLAR VE ÖNERİLER

Tez çalışması kapsamında, STİ içeren HBA üretimi ve zenginleştirilmesi yönelik kapsamlı bir yöntemler bütünü oluşturmak için aşağıdaki araştırma sorularına cevaplar aranmıştır:

1. APT analiz raporlarından STİ çıkarımı yapılarak HBA oluşturulabilir mi?
2. STİ zenginleştirme işlemleri uygulanarak HBA'nın bağlamsallığı yüksek seviyeye ulaştırılabilir mi?
3. Çizge tabanlı tahminleme modelleri geliştirilerek APT grupları arasında takma ad ilişkileri ve faili meçhul saldırılar ile APT grupları ilişkiler kurulabilir mi?

Belirlenen araştırma sorularına cevap vermek için kapsamlı bir literatür taraması yapılmıştır. Literatür taraması kapsamında temelde iki ana konuya odaklanılmıştır. Bunlardan birincisi STİ, ikincisi ise HBA'dır. Bu iki alan arasındaki bağın kurulması ve bunu yaparken bilimsel temelin doğru oturtulabilmesi için çizge teorisi ve ayrık matematik çalışma alanlarından yararlanılmıştır. Belirlenen araştırma sorularına yönelik yapılan çalışmalar ve ulaşılan sonuçlar aşağıda özetlenmiştir.

Birinci araştırma sorusuna cevap ararken, açık kaynaklı, topluluk desteği yüksek, sık güncellenen ve güvenlik şirketlerinin de katkı sağladığı bir APT analiz raporu koleksiyonu kullanılmıştır. APT analiz raporları farklı türde birçok GİG türü barındırdığı gözlemlenmiştir. RegEx ve NER yöntemleriyle mümkün olan en yüksek çeşitlilikte ve sayıda GİG ayrıştırılmıştır. Elde edilen GİG'ler karşılaşılan APT raporları ile ilişkilendirilerek bir çizgeye dönüştürülmüştür. Oluşturulan çizgede 25 düğüm türü ve 1 ilişki türü elde edilmiştir. Bu yönüyle bir HBA elde edilmiş ve 57,496 adet düğüm ile 68,873 adet ilişki meydana gelmiştir. Birinci araştırma sorusu için yapılan çalışmalar başarılı olmuş ve APT analiz raporlarından STİ içeren bir HBA oluşturulmuştur.

İkinci araştırma sorusuna cevap ararken, STİ zenginleştirme sürecinde kullanılabilecek veri kaynakları belirlenmiştir. Veri kaynakları arasında, açık kaynaklardan toplanan blok listeleri, zararlı yazılım analizinden elde edilen GİG'ler, DNS sorgusu, Whois sorgusu ve uzak bilgisayarlarda port tarama gibi aktif veri toplama yöntemleriyle ortaya çıkarılan bilgi, X (Twitter) sosyal medya platfomundan toplanan operasyonel STİ gibi veri kaynaklarından veriler toplanmıştır. STİ zenginleştirmeye yönelik toplanan veriler HBA üzerinde düğümler halinde kaydedilmiştir. Düğümler arasında yeni ilişkiler kurulmuştur.

Zenginleştirme işlemi sonucunda HBA'da toplam 48 düğüm türü ve 25 ilişki türü elde edilmiştir. Tez çalışmasında önerilen pasif ve aktif veri toplama yöntemleri başarıyla uygulanarak HBA'nın bağlamsallığı artırılmıştır.

Üçüncü araştırma sorusuna cevap ararken, çizge tabanlı tahminleme modelleri üzerinde çalışılmıştır. Tez çalışmasına başlarken APT alanıdaki bilinmezliklere ışık tutmak hedeflenmiştir. Bu motivasyon dahilinde iki problem belirlenmiştir. Bunlardan birincisi APT saldırıları yapabilecek kabiliyetlere sahip tehdit aktörleri siber güvenlik organizasyonları tarafından profillenerek isimlendirilmektedir. Fakat bu isimler isimlendiren taraflar arasında farklılık göstermektedir. Birçok tehdit aktörünün çok sayıda takma ismi bulunmaktadır. Aktörlerin takma isimleri arasındaki eşleştirme geleneksel yaklaşıma göre sahip oldukları TTP'ler üzerinden yapılmaktadır. Tez çalışmasında, çok daha geniş bir açıdan, bütüncül olarak ele alınan STİ ağı oluşturulmuş ve APT grupları arasında takma isim tahminleri üretilmiştir. Gözlemlenen ikinci problem ise APT grupları ile ilişkilendirilememiş, faili meçhul saldırıların bulunmasıdır. Bu tür saldırıların faillerinin ortaya çıkarılması için tahminler üretilmiştir. Bu tahminlerin üretilmesi için Rastgele Orman (Random Forest), Çok Katmanlı Algılayıcı (Multilayer Perceptron) ve Lojistik Regresyon (Logistic Regression) olmak üzere üç farklı algoritma kullanılarak model adayları üretilmiş ve en başarılı olan modeller üzerinden ilerlenmiştir. APT grupları arasında takma adların tahminlenmesinde en başarılı model %96,36 eğitim skoru ve %97,88 test skoru ile Logistic Regression algoritması kullanılarak elde edilmiştir. Faili meçhul sadırılar ile APT grupları arasındaki ilişkilerin tahminlenmesinde en başarılı model %77,35 eğitim skoru ve %76,65 test skoru ilse Multilayer Perceptron algoritması ile elde edilmiştir.

STİ içeren HBA'ların üretimi ve zenginleştirilmesine yönelik yapılan bu tez çalışması kapsamında aşağıda listelenen sonuçlara ulaşılmıştır;

1- Teknik, taktik ve operasyonel STİ'den meydana gelen, çok sayıda kaynağın bir araya getirilmesi ile yüksek bağlamsallığa sahip bir HBA üretilebildiği,
2- Farlı seviyelerdeki teknik, taktik ve operasyonel STİ bilgisinin ayrıştırılarak, sınıflandırılarak, içerdikleri bilginin semantik bağlamı korunarak HBA'ya dönüştürülmesi sonucunda katma değeri yüksek, geçmişte bilinmeyen, yeni STİ'lerin üretilmesinin mümkün olduğu,
3- STİ alanında, çizge veri yapısı ile ifade edilen HBA'ların, bu alandaki verinin

potansiyelini ortaya çıkarmak için bir fırsat sunduğu,

4- Faili meçhul APT saldırıları ve APT gruplarına ilişkin tahminlerin yapılmasında, STİ zenginleştirme işlemlerinin etkin şekilde yerine getirilmesinin tahmin kalitesini artırdığı,
5- Çizge tabanlı tahminleme modellerinin ürettiği bilginin, STİ yaşam döngüsünde yeni iterasyonları tetikleme kabiliyetinin yüksek olduğu, tehdit avcılığı faaliyetlerinde yol gösterici nitelikte değerli bilgiler üretebildiği görülmüştür.

Tez çalışması süresince karşılaşılan zorluklar iki ayrı perspektifte ele alınmıştır. Bunlar uygulanan yöntemlere dair zorluklar ve operasyonel zorluklardır. Çalışmada uygulanan yöntemlere ilişkin olarak karşılaşılan zorluklar aşağıda listelenmiştir;

1- Yüksek bağlamsallığa sahip bir HBA elde etmek için, doğru ve güvenilir STİ kaynakları seçilmesi gerekmektedir. Veri kaynaklarının birbiriyle ilişkilendirilmesi gerekmekte ve bu işlem yapıldığında anlamlı bir ağ oluşturulması gerekmektedir. Ağ üzerinde birbiriyle bağlantısız çok sayıda alt çizgenin oluşması durumunda, tahmin kalitesinin düşürme riskini ortaya çıkarmaktadır. Bu sorunu çözmek için HBA'yı yüksek düzeyde bağlamsallaştırmaya izin veren veri kaynakları belirlenmiş ve kullanılmıştır.
2- Diğer bir zorluk ise ham STİ verilerinin ayrıştırılması ve HBA üretimine uygun bir formata dönüştürülmesidir. APT analiz raporlarından GİG'leri çıkarmak için literatürde bulunan birçok GİG ayrıştırıcısı denenmiştir. Caballero ve diğerleri tarafından sunulan çözüm [107] ile karşılaşılmış ve tez çalışmasında kullanılmıştır.
3- Kazanan çizge tabanlı tahminleme modelini oluşturmakta da zorluklar ile karşılaşılmıştır. Sonuçta ortaya çıkan HBA'da birçok APT grubu olmasına rağmen, bunların birbiriyle ilişki sayısı düşüktü ve çalışmamızın çözmeye çalıştığı asıl zorluk burada yatmaktadır. APT gruplarının komşu APT grubu sahibi oluşunun az olması, HBA'da dengesiz sınıf dağılımını meydana getirmiştir. Aday modelleri karşılaştırmak için bir değerlendirme ölçütü olarak, veri kümesinin dengesiz sınıf dağılımına sahip olduğu durumlarda önerilen AUCPR kullanılmıştır.

Çalışmada karşılaşılan operasyonel zorlukların çoğu dış bağımlılıklardan meydana gelmektedir. Bu zorlukların paylaşılması, gelecek araştırmacılara yol göstermesi açısından öncemli görülmüştür ve aşağıda listelenmiştir;

1- Tez çalışmasının önemli bir yönü olarak internetten yeterli hale gelinceye kadar uzun

süre gerçek dünya verisi toplanmış olmasıdır. Sentetik veya dar kapsamlı veri kümeleriyle sınırlı kalınmaması hedeflenmiştir. Veri toplama ve işleme uygulamalarını çevrimiçi, çalışır ve sağlıklı olacak şekilde yönetmek başlıbaşına bir zorluğa dönüşmüştür. Kullanılan hedef veri kaynakları zaman zaman veri formatlarını değiştirmektedir, bu nedenle değişen veri formatlarına ayak uydurmak zorunda kalınmıştır.

2- Bir diğer zorluk ise pasif ve aktif STİ verisi toplamayla ilgilidir. Veri toplama uygulamalarının üzerinde çalıştığı bulut platformları, düzenli olarak web sitelerini ve uzaktaki bilgisayarları tarayan kullanıcılara izin vermemektedir. Yürütülen faaliyetlerden bazıları, kullanıcı davranışlarını sınıflandırmaya yönelik gelişmiş tekniklere sahip oldukları için bulut sağlayıcılar tarafından kötü amaçlı olarak değerlendirilmiştir. Bulut sunucularda çalıştırılan kod ve işlemler, ağ aktiviteleri, bant genişliği kullanımı, trafik hacmi, kullanılan IP adresleri ile ilgili hedef sistemlerden, bulut sağlayıcılara gelen kötüye kullanım bildirimleri, karar verme algoritmalarının potansiyel parametrelerinden bazıları olduğu değerlendirilmektedir. Olağandışı ve şüpheli işlemler nedeniyle birçok kez yürütülen faaliyetler ve kullanıcı hesapları engellenmiştir. Benzer operasyonları yürütmekle ilgilenen araştırmacıların planlarını bulut sunucu sağlayıcılarla görüşerek izin almayı denemeleri ya da daha az katı kontrollere sahip küçük veya orta ölçekli sunucu barındırma şirketlerini tercih etmeleri önerilmektedir.

Tez çalışması sürecinde STİ özelinde yapılan araştırmalarda görülmüştür ki, ülkemizde bu alanda lisans üstü seviyede yapılan çalışmalar oldukça kısıtlıdır. Henüz STİ alanında ulusal ölçekte ve akademik düzeyde kapsayıcı bir ontoloji çalışması yapılmadığı görülmektedir. Bilgi güvenliği son yıllarda ülkemizde de mühendislik seviyesinde çalışılabilir durumdadır ve uzmanlaşmak mümkündür. Bilgi Güvenliği Mühendisliği'nin özelleşmiş bir alt alanı olarak Siber Tehdit İstihbaratı oldukça yeni bir çalışma alanıdır. Ülkemizde STİ alanında ulusal güvenlik odaklı çalışmalar, siber güvenlik endüstrisinde faaliyet yürüten az sayıdaki uluslararası başarıya sahip şirket ve kamu kurumlarının çabalarıyla desteklenmeye ve zenginleştirmeye çalışılmaktadır.

STİ alanında özgün teknoloji, süreç, politika ve ürün geliştirilmesi için bilimsel düşüncenin desteklenmesi ve yaygınlaştırılması gerektiği değerlendirilmektedir. Araştırmacıların, lisans ve lisansüstü öğrencilerin STİ alanına teşvik edilmesi, bilimsel çalışmaların özel

sektöre ve kamuya liderlik etmesi, bu disiplinin ülkemizde doğru temellendirilmesi için kritik öneme sahiptir. Aksi taktirde STİ alanı, yetkin olmayan kişilerin vasat çalışmalarına mahkûm olacaktır. Günümüzde siber istihbarat geleneksel istihbarat faaliyetlerinin sağlayamadığı faydaları ve fırsatları mümkün kılmaktadır. Öte yandan siber tehditler ve siber espiyonaj ise yine geleneksel istihbarat ve casusluk faaliyetlerinin oluşturamadığı zarar ve tehditleri mümkün kılmaktadır.

Tez sürecinde edinilen bilgi ve tecrübe ışığında geleceğe yönelik çeşitli araştırma fikirleri oluşmuştur ve aşağıda sunulmuştur;

1- Derin web ve karanlık web veri kaynaklarının HBA oluşturma hattına entegrasyonu ile katma değeri yüksek STİ üretme kapasitesi kazanılabileceği,
2- Zararlı olduğu şüphesiyle yaklaşılan uzak bilgisayarlara, ağdaki vekil (proxy) cihazlar üzerinden gidilmesinin toplanan verinin kalitesini ve doğruluğunu artırabileceği,
3- Yazılım dünyasında kullanılan teknolojiler (programlama dilleri, programlama çerçeveleri, kütüphaneler, açık kaynak yazılım bileşenleri vb.), sistem bileşenleri (işletim sistemleri, sunucu donanımları, ağ cihazları ve yazılımları vb.) ve bunların versiyonları HBA oluşturma hattına entegre edilmesiyle organizasyonların, potansiyel saldırıların kendilerine nasıl yönlenebileceği ile ilgili önleyici tedbirler almasının mümkün kılınabileceği,
4- STİ ve APT saldırıları alanında ihtiyaç olarak görülen bir diğer konu, bu alanda yüksek başarıya sahip bir NER modeli bulunmamasıdır. Geliştirilecek yeni bir NER'in metin tabanlı kaynaklardan STİ ve APT bilgisi çıkarmada kritik öneme sahip olabileceği değerlendirilmektedir.
5- STİ ve APT alanındaki metinsel veri kaynaklarından HBA'yı beslemeye yönelik bilgi çıkarımı yapmak ve STİ üretiminde kullanmak için büyük dil modellerinin (large language models, LLM) gelecek vaat ettiği değerlendirilmektedir.

Tez çalışması boyunca karşılaşılan zorluklar ve literatürdeki çalışmaların yöneldiği araştırma alanları göz önüne alınarak yukarıdaki liste oluşturulmuştur. Bu öneri listesinin gelecekteki araştırmacıları katma değeri yüksek yeni STİ'ler üretmelerini sağlayacağı değerlendirilmektedir.

KAYNAKLAR

1. Gunderman, R. B. ve Applegate, K. E. (2005). Managing risk: Threat or opportunity? *American Journal of Roentgenology,* 185(1), 43-45.

2. Morgan, J. (2003). What is meta-reality? *Journal of Critical Realism*, 1(2), 115-146.

3. Mintzberg, H. (1987). The Strategy Concept I: Five Ps for Strategy. *California Management Review*, 30(1), 11-24.

4. Bayraktar, G. (2014). Harbin beşinci boyutunun yeni gereksinimi: siber istihbarat. *Güvenlik Stratejileri Dergisi*, 10(20), 119-148.

5. Sipper, J. A. (2020, 15-19 June). *Cyber threat ıntelligence and the cyber meta-reality and cyber microbiome.* International Conference on Cyber Security and Protection of Digital Services, (Cyber Security), 1-5. Dublin, Ireland

6. AL-Aamri, A. S., Abdulghafor, R., Turaev, S., Al-Shaikhli, I., Zeki, A., ve Talib, S. (2023). Machine learning for APT detection. *Sustainability (Switzerland),* 15(18), 13820-13836.

7. Kim, Y., Lee, I., Kwon, H., Lee, K., ve Yoon, J. (2023). BAN: Predicting APT attack based on bayesian network With MITRE ATT&CK framework. *Institute of Radio Engineers Access*, 11, 91949-91968.

8. Sun, H., Yang, X., Yang, L. X., Huang, K., ve Li, G. (2023). Impulsive artificial defense against advanced persistent threat. *Institute of Radio Engineers Transactions on Information Forensics and Security*, 18, 3506-3516.

9. Sun, N., Ding, M., Jiang, J., Xu, W., Mo, X., Tai, Y., ve Zhang, J. (2023). Cyber threat ıntelligence mining for proactive cybersecurity defense: a survey and new perspectives. *International Research Journal of Modernization in Engineering Technology and Science*, 25(3), 1748-1774.

10. Hindy, H., Brosset, D., Bayne, E., Seeam, A. K., Tachtatzis, C., Atkinson, R., ve Bellekens, X. (2020). A taxonomy of network threats and the effect of current datasets on ıntrusion detection systems. *Institute of Radio Engineers Access*, 8, 104650-104675.

11. Corcho, O., Fernández-López, M., ve Gómez-Pérez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data and Knowledge Engineering,* 46(1), 41-64.

12. Fenz, S. ve Ekelhart, A. (2009). *Formalizing information security knowledge.* Proceedings of the 4th International Symposium on ACM Symposium on Information, Computer and Communications Security, 183–194, New York, USA.

13. Wang, J. A., ve Guo, M. (2009). *OVM: An ontology for vulnerability management*. ACM International Conference Proceeding Series, 1, 1-4.

14. Obrst, L., Chase, P., ve Markeloff, R. (2014). *Developing an ontology of the cyber security domain.* Proceedings of the 10th Annual Cyber and Information Security Research Conference, 1-4, Oak Ridge, TN, USA.

15. More, S., Matthews, M., Joshi, A., ve Finin, T. (2012). *A knowledge-based approach to intrusion detection modeling.* IEEE Symposium on Security and Privacy Workshops, 75-81.

16. Oltramari, A., Cranor, L. F., Walls, R. J., ve McDaniel, P. (2014). *Building an ontology of cyber security*. CEUR Workshop Proceedings, 54-61.

17. Grecio, A., Bonacin, R., Nabuco, O., Afonso, V. M., De Geus, P. L., ve Jino, M. (2014). *Ontology for malware behavior: A core model proposal.* Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE, 453-458.

18. Salem, M. B. ve Wacek, C. (2015). *Enabling new technologies for cyber security defense with the ICAS cyber security ontology*. CEUR Workshop Proceedings, 1523, 42-49.

19. Iannacone, M., Bohn, S., Nakamura, G., Gerth, J., Huffer, K., Bridges, R., Ferragut, E., ve Goodall, J. (2015). *Developing an ontology for cyber security knowledge graphs.* Proceedings of the 10th Annual Cyber and Information Security Research Conference, 1-4.

20. Grégio, A., Bonacin, R., De Marchi, A. C., Nabuco, O. F., ve De Geus, P. L. (2016). An ontology of suspicious software behavior. *Applied Ontology*, 11(1), 29-49.

21. Atighetchi, M., Simidchieva, B., Yaman, F., Eskridge, T., Carvalho, M., ve Paltzer, N. (2016). Using ontologies to quantify attack surfaces. *Semantic Technology for Intelligence, Defense, and Security*, 1788, 10-18.

22. Pendleton, M., Garcia-Lebron, R., Cho, J. H., ve Xu, S. (2016). A survey on systems security metrics. *ACM Computing Surveys*, 49(4), 1-35.

23. Syed, Z., Pädia, A., Finin, T., Mathews, L., ve Joshi, A. (2016). *UCO: A Unified Cybersecurity Ontology*. AAAI Workshop: Artificial Intelligence for Cyber Security, 2-8.

24. Mavroeidis, V. ve Bromander, S. (2017). *Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence*. European Intelligence and Security Informatics Conference (EISIC), 91-98. Athens, Greece.

25. Bocij, P., Greasley, A., ve Hickie, S. (2015). *Business ınformation systems: technology, development and management for the e-business* (4th Edition). Canada: Pearson Education, 3-36.

26. Das, V. V. (2006). *Principles of data structures using C and C++* (1st Edition). India: New Age Publishers, 1-16.

27. Peterson, L. L. ve Davie, B. S. (2011). *Computer networks: A systems approach* (5th Edition). USA: Morgan Kaufmann Publishers, 1-13.

28. Losee, R. M. (1997). A discipline independent definition of information. *Journal of the American Society for Information Science*, 48(3), 254-269.

29. Floridi, L. (2010). *Information: A very short ıntroduction* (1st Edition). USA: Oxford University Press, 1-17.

30. Uçak, Ö. N. (2010). Bilgi: çok yüzlü bir kavram. *Türk Kütüphaneciliği*, 24(4), 8-12.

31. Yuexiao, Z. (1988). Definitions and sciences of information. *Information Processing and Management*, 24(4), 479-491.

32. Madden, A. D. (2000). A definition of information. *Aslib Proceedings, 52*(9), 343–349.

33. Bhandari, C. (2021). Intelligence in the national security enterprise: An ıntroduction: By Roger Z. George. *Journal of APF Command and Staff College,* 4(1), 161–166.

34. Shirey, R. (2007). Internet security glossary, version 2. *Request for Comments*, *2*.

35. Furlani, C.M. (2009). *Minimum security requirements for federal ınformation and ınformation systems* (1st Edition). USA: DIANE Publishing, 9-10.

36. Mattern, T., Felker, J., Borum, R., ve Bamford, G. (2014). Operational Levels of Cyber Intelligence. *International Journal of Intelligence and CounterIntelligence*, *27*(4), 702-719.

37. Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., ve Beyah, R. (2016). *Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence*. Proceedings of the ACM Conference on Computer and Communications Security, 755-766.

38. Catakoglu, O., Balduzzi, M., ve Balzarotti, D. (2016). *Automatic extraction of indicators of compromise for web applications.* 25th International World Wide Web Conference, 333-343. https://doi.org/10.1109/EISIC.2017.20

39. Shi, C., Li, Y., Zhang, J., Sun, Y., ve Yu, P. S. (2017). A Survey of Heterogeneous Information Network Analysis. *IEEE Transactions on Knowledge and Data Engineering*, *29*(1), 17-37.

40. Shi, C., Hu, B., Zhao, W. X., ve Yu, P. S. (2019). Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 31(2), 357-370.

41. Phythian, M. (2013). *Understanding the ıntelligence cycle* (1st Edition). England: Routledge, 1-13.

42. von Solms, B. ve von Solms, R. (2018). Cybersecurity and information security – what goes where? *Information and Computer Security*, 26(1), 2-9.

43. Marble, J. L., Lawless, W. F., Mittu, R., Coyne, J., Abramson, M., ve Sibley, C. (2015). The human factor in cybersecurity: Robust & intelligent defense. *Advances in Information Security*, 56, 173-206.

44. Sillaber, C., Sauerwein, C., Mussmann, A., ve Breu, R. (2016). *Data quality challenges and future research directions in threat intelligence sharing practice.* Proceedings of the 2016 ACM Workshop on Information Sharing and Collaborative Security, 65-70, Salt Lake City, UT, ABD.

45. Schaberreiter, T., Kupfersberger, V., Rantos, K., Spyros, A., Papanikolaou, A., Ilioudis, C., ve Quirchmayr, G. (2019). *A quantitative evaluation of trust in the quality of cyber threat intelligence sources.* ACM International Conference Proceeding Series, 1-10, Association for Computing Machinery New York NY United States.

46. Dudorov, D., Stupples, D., ve Newby, M. (2013). *Probability analysis of cyber attack paths against business and commercial enterprise systems*. Proceedings European Intelligence and Security Informatics Conference, 38-44, Uppsala, Sweden.

47. Qamar, S., Anwar, Z., Rahman, M. A., Al-Shaer, E., ve Chu, B. T. (2017). Data-driven analytics for cyber-threat intelligence and information sharing. *Computers and Security*, 67, 35-58.

48. Guitton, C. (2017). *Inside the enemy's computer: Identifying cyber attackers* (1st Edition). USA: Oxford University Press, 1-12.

49. Smith, C. L., ve Brooks, D. J. (2013). *Security science*. USA: Butterworth-Heinemann, 1-22.

50. Arnold, N., Ebrahimi, M., Zhang, N., Lazarine, B., Patton, M., Chen, H., ve Samtani, S. (2019). *Dark-net ecosystem cyber-threat intelligence (CTI) Tool.* IEEE International Conference on Intelligence and Security Informatics, 92-97, Shenzhen, China.

51. Chen, C. M., Lai, G. H., ve Wen, D. W. (Marian). (2019). Evolution of Advanced Persistent Threat (APT) Attacks and Actors. *Communications in Computer and Information Science*, 1013, 76-81.

52. Barnum, S. (2014). Standardizing cyber threat intelligence information with the Structured Threat Information eXpression. *MITRE Corporation*. 1-9.

53. İnternet: Cyber Threat Intelligence (CTI) TC, *Oasis Open*. URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cti, Son Erişim Tarihi: 18/11/2023

54. İnternet: Malware Attribute Enumeration and Characterization V5.0, *MITRE*. URL: https://maecproject.github.io/, Son Erişim Tarihi: 18/11/2023.

55. İnternet: OpenIOC V1.1, *Fireeye*. URL: https://github.com/fireeye/OpenIOC_1.1, Son Erişim Tarihi: 18/11/2023.

56. Zhao, H., Yao, Q., Li, J., Song, Y., ve Lee, D. L. (2017). *Meta-graph based recommendation fusion over heterogeneous information networks.* Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 635-644, Auckland, New Zealand.

57. Zhai, X., Tang, Z., Liu, Z., Zhou, W., Hu, H., Fei, G., ve Hu, G. (2023). Sparse representation for heterogeneous information networks. *Neurocomputing*, 525, 111-122.

58. Mohammadfam, I., Ghasemi, F., Kalatpour, O., ve Moghimbeigi, A. (2017). Constructing a Bayesian network model for improving safety behavior of employees at workplaces. *Applied Ergonomics*, 58, 35-47.

59. Yang, Z., Algesheimer, R., ve Tessone, C. J. (2016). A comparative analysis of community detection algorithms on artificial networks. *Scientific Reports*, 6(1), 1-18.

60. Bedi, P., ve Sharma, C. (2016). Community detection in social networks. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 6(3), 115–135.

61. Liang, Y., Cao, X., He, D., Chuan, W., Xiao, W., ve Weixiong, Z. (2016). *Modularity based community detection with deep learning*. International Joint Conference on Artificial Intelligence, 1, 2252-2258.

62. Zhang, M., ve Chen, Y. (2018). Link prediction based on graph neural networks. *Advances in Neural Information Processing Systems*, 31, 5171-5181.

63. Zhu, L., Guo, D., Yin, J., Steeg, G. Ver, ve Galstyan, A. (2016). Scalable temporal latent space inference for link prediction in dynamic social networks. *IEEE Transactions on Knowledge and Data Engineering*, 28(10), 2765-2777.

64. Benson, A. R., Gleich, D. F., ve Leskovec, J. (2016). Higher-order organization of complex networks. *Science*, 353(6295), 163-166.

65. Zhou, Y. ve Liu, L. (2013). *Social influence based clustering of heterogeneous information networks*. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 338-346, Chicago Illinois USA.

66. Zhuang, H., Sun, Y., Tang, J., Zhang, J., ve Sun, X. (2013). *Influence maximization in dynamic social networks*. Proceedings IEEE International Conference on Data Mining, 1, 1313-1318.

67. Peng, S., Zhou, Y., Cao, L., Yu, S., Niu, J., ve Jia, W. (2018). Influence analysis in social networks: A survey. *Journal of Network and Computer Applications*, 106, 17-32.

68. Satoh, A., Nakamura, Y., Fukuda, Y., Sasai, K., ve Kitagata, G. (2019). A cause-based classification approach for malicious dns queries detected through blacklists. *IEEE Access*, 7. 142991-143001.

69. Kim, B. I., Kim, N., Lee, S., Cho, H., ve Park, J. (2018). *A Study on a Cyber Threat Intelligence Analysis (CTI) Platform for the Proactive Detection of Cyber Attacks Based on Automated Analysis*. International Conference on Platform Technology and Service, 1-6, Jeju, Korea (South).

70. Lu, J., Zhang, X., Junfeng, W., ve Lingyun, Y. (2017). *APT traffic detection based on time transform*. Proceedings International Conference on Intelligent Transportation, Big Data and Smart City, 9-13, Changsha, China.

71. Ghafir, I., ve Prenosil, V. (2015). *Blacklist-based malicious IP traffic detection.* Global Conference on Communication Technologies, 229-233, Thuckalay, India.

72. Ghafir, I., Prenosil, V., Hammoudeh, M., Baker, T., Jabbar, S., Khalid, S., ve Jaf, S. (2018). BotDet: A system for real time botnet command and control traffic detection. *IEEE Access*, 6. 39947-38958.

73. Zhang, J. ve Wang, X. (2021). Consensus-based distributed quickest detection of attacks with unknown parameters. *IEEE Transactions on Information Theory*, 67(3), 1864-1885.

74. Choi, J., Choi, C., Lynn, H. M., ve Kim, P. (2015). *Ontology Based APT Attack Behavior Analysis in Cloud Computing*. Proceedings International Conference on Broadband and Wireless Computing, Communication and Applications, 375-379, Krakow, Poland.

75. Almohannadi, H., Awan, I., Al Hamar, J., Cullen, A., Disso, J. P., ve Armitage, L. (2018). *Cyber threat intelligence from honeypot data using elasticsearch*. Proceedings International Conference on Advanced Information Networking and Applications, 900-906, Krakow, Poland.

76. Chen, T. M. ve Abu-Nimeh, S. (2011). Lessons from Stuxnet. *Computer*, 44(4), 91-93.

77. Bencsáth, B., Pék, G., Buttyán, L., ve Félegyházi, M. (2012, April). *Duqu: Analysis, detection, and lessons learned.* In ACM European Workshop on System Security (EuroSec) (Vol. 2012), Bern, Switzerland.

78. Bencsáth, B., Pék, G., Buttyán, L., ve Félegyházi, M. (2012). The cousins of Stuxnet: Duqu, Flame, and Gauss. *Future Internet*, 4(4), 971-1003.

79. Kambourakis, G., Kolias, C., ve Stavrou, A. (2017). *The mirai botnet and the ıot zombie armies*. Proceedings IEEE Military Communications Conference, 267-272, Baltimore, MD, USA.

80. Fachkha, C., ve Debbabi, M. (2016). Darknet as a source of cyber ıntelligence: survey, taxonomy, and characterization. *Communications Surveys and Tutorials*, 18(2), 1197-1227.

81. Innab, N. ve Alamri, A. (2018). *The Impact of DDoS on E-commerce.* 21st Saudi Computer Society National Computer Conference, 1-4, Riyadh, Saudi Arabia.

82. White, A., Galloway, M., Boyle, P. O., ve Wyllie, S. (2021). *The Impact of DDoS attacks on the power usage of virtual execution environments.* Conference Proceedings IEEE SoutheastCon, (pp. 1-6), Atlanta, GA, USA.

83. Abhishta, A., Joosten, R., Dragomiretskiy, S., ve Nieuwenhuis, L. J. M. (2019). *Impact of successful DDoS attacks on a major crypto-currency exchange.* Proceedings 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, (pp. 379-384), TMO Univeristy, Saint-Petersburg, Russian Federation.

84. Marchetti, M., Guido, A., Pierazzi, F., ve Colajanni, M. (2016). *Countering Advanced Persistent Threats through security intelligence and big data analytics*. International Conference on Cyber Conflict, 243-261.

85. Gong, S., Cho, J., ve Lee, C. (2018). A reliability comparison method for OSINT validity analysis. *IEEE Transactions on Industrial Informatics*, 14(12), 5428-5435.

86. Cantelli-Forti, A. ve Colajanni, M. (2018). *Adversarial fingerprinting of cyber attacks based on stateful honeypots.* Proceedings International Conference on Computational Science and Computational Intelligence, 19-24, Las Vegas, NV, USA.

87. Noor, U., Anwar, Z., ve Rashid, Z. (2018). *An Association Rule Mining-Based Framework for Profiling Regularities in Tactics Techniques and Procedures of Cyber Threat Actors*. International Conference on Smart Computing and Electronic Enterprise, 1-6, Shah Alam, Malaysia.

88. Ioannou, G., Louvieris, P., ve Clewley, N. (2019). A Markov Multi-Phase Transferable Belief Model for Cyber Situational Awareness. *IEEE Access*, 7, 39305-39320.

89. Jin, X., Cui, B., Yang, J., ve Cheng, Z. (2018). *An adaptive analysis framework for correlating cyber-security-related data*. Proceedings International Conference on Advanced Information Networking and Applications, 915-919, Krakow, Poland.

90. Deliu, I., Leichter, C., and Franke, K. (2018). *Collecting Cyber Threat Intelligence from Hacker Forums via a Two-Stage, Hybrid Process using Support Vector Machines and Latent Dirichlet Allocation*. Proceedings - 2018 IEEE International Conference on Big Data, 5008-5013, Seattle, WA, USA.

91. Abdullah, M. S., Zainal, A., Maarof, M. A., ve Nizam Kassim, M. (2018). *Cyber-Attack Features for Detecting Cyber Threat Incidents from Online News*. Proceedings of the 2018 Cyber Resilience Conference, 1-4, Putrajaya, Malaysia.

92. Me, G., Pesticcio, L., ve Spagnoletti, P. (2017). *Discovering Hidden Relations Between Tor Marketplaces Users.* IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence ve Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 494–501, Orlando, FL, USA.

93. Paradise, A., Shabtai, A., Puzis, R., Elyashar, A., Elovici, Y., Roshandel, M., ve Peylo, C. (2017). Creation and management of social network honeypots for detecting targeted cyber attacks. *Transactions on Computational Social Systems*, *4*(3). 65-79.

94. Chanajitt, R., Pfahringer, B., ve Gomes, H. M. (2021). *Combining Static and Dynamic Analysis to Improve Machine Learning-based Malware Classification*. IEEE 8th International Conference on Data Science and Advanced Analytics, 1-10, Porto, Portugal.

95. Cruickshank, I. J. ve Carley, K. M. (2020). Analysis of malware communities using multi-modal features. *IEEE Access*, 8, 77435-77448.

96. Bromander, S., Jøsang, A., ve Eian, M. (2016). *Semantic cyber threat modelling.* CEUR Workshop Proceedings, 1788, 74-78.

97. Mirza, Q. K. A., Brown, M., Halling, O., Shand, L., ve Alam, A. (2021). *Ransomware Analysis using Cyber Kill Chain*. Proceedings 2021 International Conference on Future Internet of Things and Cloud, 58-65, Rome, Italy.

98. Zhu, X., Jiang, L., Chen, Z., ve Yan, L. (2021). *Searching for similar binary code based on the influence of basic block vertex*. Proceedings 3rd International Conference on Advances in Computer Technology, Information Science and Communication, 17-24.

99. Grammatikakis, K. P., Koufos, I., Kolokotronis, N., Vassilakis, C., ve Shiaeles, S. (2021). *Understanding and mitigating banking trojans: From Zeus to emotet.* Proceedings of the 2021 International Conference on Cyber Security and Resilience, 121-128, Rhodes, Greece.

100. Henderson, A., Yan, L. K., Hu, X., Prakash, A., Yin, H., ve McCamant, S. (2017). DECAF: A platform-neutral whole-system dynamic binary analysis platform. *IEEE Transactions on Software Engineering*, *43*(2), 164-184.

101. Rasthofer, S., Arzt, S., Triller, S., ve Pradel, M. (2017). *Making malory behave maliciously: targeted fuzzing of android execution environments.* Proceedings IEEE/ACM 39th International Conference on Software Engineering, 300-311, Buenos Aires, Argentina.

102. Wu, D. J., Mao, C. H., Wei, T. E., Lee, H. M., ve Wu, K. P. (2012). *DroidMat: Android malware detection through manifest and API calls tracing.* Proceedings of the 7th Asia Joint Conference on Information Security, 62-69.

103. Wang, H., Zhu, Z., Tong, Z., Yin, X., Feng, Y., Shi, G., ve Meng, D. (2021). *An Effective Approach for Malware Detection and Explanation via Deep Learning Analysis*. Proceedings of the International Joint Conference on Neural Networks, 1-10.

104. Saxe, J., ve Berlin, K. (2015). *Deep neural network based malware detection using two dimensional binary program features*. 10th International Conference on Malicious and Unwanted Software, 11-20.

105. Li, J., Sun, L., Yan, Q., Li, Z., Srisa-An, W., ve Ye, H. (2018). Significant Permission Identification for Machine-Learning-Based Android Malware Detection. *IEEE Transactions on Industrial Informatics, 14*(7). 3216-3225.

106. Chen, L., Ye, Y., ve Bourlai, T. (2017). *Adversarial machine learning in malware detection: Arms race between evasion attack and defense*. Proceedings European Intelligence and Security Informatics Conference, 99-106.

107. Morency, L. P. ve Baltrušaitis, T. (2017). *Multimodal machine learning: Integrating language, vision and speech*. ACL 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference Tutorial Abstracts, 3-5.

108. Chen, Q. ve Bridges, R. A. (2017). *Automated behavioral analysis of malware: A case study of wannacry ransomware*. Proceedings 16th IEEE International Conference on Machine Learning and Applications, 454-460.

109. Almashhadani, A. O., Kaiiali, M., Sezer, S., ve O'Kane, P. (2019). A Multi-Classifier Network-Based Crypto Ransomware Detection System: A Case Study of Locky Ransomware. *IEEE Access*, 7. 47053-47067.

110. Pletinckx, S., Trap, C., ve Doerr, C. (2018). *Malware coordination using the blockchain: An analysis of the cerber ransomware*. IEEE Conference on Communications and Network Security, 1-9.

111. Hampton, N., Baig, Z., ve Zeadally, S. (2018). Ransomware behavioural analysis on windows platforms. *Journal of Information Security and Applications*, *40*, 44-51.

112. Wang, Z. H., Liu, C. G., Qiu, J., Tian, Z. H., Cui, X., ve Su, S. (2018). Automatically Traceback RDP-Based Targeted Ransomware Attacks. *Wireless Communications and Mobile Computing*, 1-13.

113. Mbol, F., Robert, J. M., ve Sadighian, A. (2016). An efficient approach to detect torrentlocker ransomware in computer systems. *Cryptology and Network Security*, 10052, 532–541.

114. Liao, K., Zhao, Z., Doupe, A., ve Ahn, G. J. (2016). Behind closed doors: Measurement and analysis of CryptoLocker ransoms in Bitcoin. *Symposium on Electronic Crime Research*, 1-13.

115. Aidan, J. S., Verma, H. K., ve Awasthi, L. K. (2017). *Comprehensive survey on petya ransomware attack*. Proceedings International Conference on Next Generation Computing and Information Systems. 122-125.

116. Alotaibi, F. M., ve Vassilakis, V. G. (2021). SDN-Based Detection of Self-Propagating Ransomware: The Case of BadRabbit. *IEEE Access*, *9*. 28039-28058.

117. Genc, Z. A. ve Lenzini, G. (2020). *Dual-use Research in Ransomware Attacks: A Discussion on Ransomware Defence Intelligence*. International Conference on Information Systems Security and Privacy, 585-592.

118. Bajpai, P. ve Enbody, R. (2020). *Preparing smart cities for ransomware attacks*. Proceedings 3rd International Conference on Data Intelligence and Security, 127-133.

119. Caballero, J., Gomez, G., Matic, S., Sánchez, G., Sebastián, S., ve Villacañas, A. (2023). The Rise of GoodFATR: A Novel Accuracy Comparison Methodology for Indicator Extraction Tools. *Future Generation Computer Systems*, 144, 74-89.

120. Gao, C., Zhang, X., Han, M., ve Liu, H. (2021). A review on cyber security named entity recognition. In *Frontiers of Information Technology and Electronic Engineering*, 22(9), 1153–1168.

121. Yuli Vasiliev. (2020). *Natural Language Processing with Python and spaCy: A Practical Introduction (1st Edition)*. USA: No Starch Press, 15-31.

122. Alam, M. T., Bhusal, D., Park, Y., ve Rastogi, N. (2022). CyNER: A Python Library for Cybersecurity Named Entity Recognition, *arXiv*.

123. İnternet: Crypto coin phishing domain addresses. URL: https://github.com/409H/EtherAddressLookup/Blob/Master/Blacklists/Domains.Json, Son Erişim Tarihi: 12/05/2024.

124. İnternet: *Anudeep crypto coin miner blocklist*. URL: https://raw.githubusercontent.com/anudeepND/blacklist/master/coinMiner.txt, Son Erişim Tarihi: 12/05/2024.

125. İnternet: *Anudeep adserver blocklist*. URL: https://raw.githubusercontent.com/anudeepND/blacklist/master/adservers.txt, Son Erişim Tarihi: 12/05/2024.

126. İnternet: *USOM blocklist*. URL: https://www.usom.gov.tr/url-list.txt, Son Erişim Tarihi: 12/05/2024.

127. İnternet: *URLHaus malware domains list*. URL: https://urlhaus.abuse.ch/downloads/hostfile.txt, Son Erişim Tarihi: 12/05/2024.

128. İnternet: *Antisocial Blocklist - UK Centric Community Version Domain Blocklist*. URL: https://raw.githubusercontent.com/TheAntiSocialEngineer/AntiSocial-BlockList-UK-Community/main/UK-Community.txt, Son Erişim Tarihi: 12/05/2024.

129. İnternet: *Proofpoint compromised IPs*. URL: http://rules.emergingthreats.net/blockrules/compromised-ips.txt, Son Erişim Tarihi: 12/05/2024.

130. İnternet: *SANS DShield Blocklist*. URL: https://feeds.dshield.org/block.txt, Son Erişim Tarihi: 12/05/2024.

131. İnternet: *Ciarmy blocklist*. URL: http://www.ciarmy.com/list/ci-badguys.txt, Son Erişim Tarihi: 12/05/2024.

132. İnternet: *SSLBL Botnet C2 IP Blacklist*. URL: https://sslbl.abuse.ch/blacklist/sslipblacklist.csv, Son Erişim Tarihi: 12/05/2024.

133. İnternet: *Feodo Tracker IP Recommended Blocklist*. URL: https://feodotracker.abuse .ch/downloads/ipblocklist_recommended.json, Son Erişim Tarihi: 12/05/2024.

134. İnternet: *Feodo Tracker IP Blocklist*. URL: https://feodotracker.abuse.ch/ downloads/ipblocklist.json, Son Erişim Tarihi: 12/05/2024.

135. İnternet: *Darklist.de Network Attackers IP List*. URL: https://darklist.de/raw.php, Son Erişim Tarihi: 12/05/2024.

136. İnternet: *Tor IPs*. URL: https://raw.githubusercontent.com/SecOps-Institute/tor-IP-Addresses/master/tor-nodes.lst, Son Erişim Tarihi: 12/05/2024.

137. İnternet: *Crypto Coin Phishing URL Addresses*. URL: https://raw. githubusercontent.com/409H/EtherAddressLookup/master/blacklists/uri.json, Son Erişim Tarihi: 12/05/2024.

138. İnternet: *URLHaus Recent Malicious URL List*. URL: https://urlhaus.abuse.ch/ downloads/csv_recent.txt, Son Erişim Tarihi: 12/05/2024.

139. İnternet: *URLHaus Full Malicious URL List*. URL: https://urlhaus.abuse.ch/ downloads/csv.zip, Son Erişim Tarihi: 12/05/2024.

140. İnternet: *Threatfox Recent Malicious URL List*. URL: https://threatfox.abuse.ch /export/csv/recent, Son Erişim Tarihi: 12/05/2024.

141. İnternet: *Threatfox Full Malicious URL List*. URL: https://threatfox.abuse.ch/export/csv/full, Son Erişim Tarihi: 12/05/2024.

142. İnternet: *Malware bazaar recent malicious file hashes*. URL: https://bazaar.abuse.ch/export /csv/recent, Son Erişim Tarihi: 12/05/2024.

143. İnternet: *Malware Bazaar full malicious file.* URL*:* https://bazaar.abuse.ch/export/csv/full, Son Erişim Tarihi: 12/05/2024.

144. İnternet: *DNS Nameservers*. URL: https://public-dns.info/nameservers-all.txt, Son Erişim Tarihi: 12/05/2024.

145. İnternet: *DNS Valid Nameservers*. URL: https://public-dns.info/nameservers.txt, Son Erişim Tarihi: 12/05/2024.

146. İnternet: *Abuse.ch SSL Blocklist*. URL: https://sslbl.abuse.ch/blacklist/sslblacklist.csv, Son Erişim Tarihi: 12/05/2024.

147. Le Pochat, V., Van Goethem, T., Tajalizadehkhoob, S., Korczynski, M., ve Joosen, W. (2019). Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. *arXiv*.

148. Sun, Y., Norick, B., Han, J., Yan, X., Yu, P. S., ve Yu, X. (2013). PathSelClus: Integrating meta-path selection with user-guided Object clustering in heterogeneous information networks. *ACM Transactions on Knowledge Discovery from Data*, *7*(3), 1-23.

149. Li, C.T. ve Lin, S.D. (2012). *Centrality Analysis, Role-Based Clustering, and Egocentric Abstraction for Heterogeneous Social Networks.* International Conference on Privacy, Security, Risk and Trust and International Conference on Social Computing, 1–10.

150. Zhang, L., Wu, Y., Wang, J., Liu, S., Chen, C., ve Wang, X. (2023). *Efficient Algorithm for Maximizing Betweenness Centrality in Large Networks*. IEEE International Conference on High Performance Computing Communications, Data Science Systems, Smart City Dependability in Sensor, Cloud Big Data Systems Application, 647–654, Melbourne, Australia.

151. Brandes, U. ve Pich, C. (2007). Centrality Estimation in Large Networks. *International Journal of Bifurcation and Chaos*, 17(07), 2303–2318.

152. Brandes, U. (2001). A Faster Algorithm for Betweenness Centrality. *The Journal of Mathematical Sociology*, *25*(2), 163–177.

153. Zhao, L., ve Zhao, J. (2017). *Comparison study of three shortest path algorithm.* 2017 International Conference on Computer Technology, Electronics and Communication, 748–751, Dalian, China.

154. Aliyan, M., Hasan, M. Z., Qayoom, H., Hussain, M. Z., Nosheen, S., Mustafa, M., Chuhan, S. H., Atif Yaqub, M., Awan, R., ve Bilal, A. (2024). *Analysis and Performance Evaluation of Various Shortest Path Algorithms*. 3rd International Conference for Innovation in Technolog, 1–16, Bangalore, India.

155. Barbehenn, M. (1998). A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices. *IEEE Transactions on Computers*, *47*(2), 263.

156. Zhang, J., Ma, X., Liu, W., ve Bai, Y. (2012). *Inferring community members in social networks by closeness centrality examination*. Ninth Web Information Systems and Applications Conference, 131–134, Haikou, China.

157. Jin, B., Yang, W., Yin, Z., ve Zhang, Y. (2020). *Finding key actors on social movement networks through improved k-shell method.* 5th International Conference on Mechanical, Control and Computer Engineering, 2026–2030, Harbin, China.

158. Brodka, P., Skibicki, K., Kazienko, P., ve Musial, K. (2011). *A degree centrality in multi-layered social network*. International Conference on Computational Aspects of Social Networks, 237–242.

159. Srinivas, A. ve Velusamy, R. L. (2015). *Identification of influential nodes from social networks based on Enhanced Degree Centrality Measure*. IEEE International Advance Computing Conference, 1179–1184, Banglore, India.

160. Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3), 75-174.

161. Chang, Q., Li, X., ve Duan, Z. (2024). A novel approach for rumor detection in social platforms: Memory-augmented transformer with graph convolutional networks. *Knowledge-Based Systems*, 292, 111625-111637.

162. Sun, P. G. (2014). Weighting links based on edge centrality for community detection. *Physica A: Statistical Mechanics and Its Applications*, 394, 346–357.

163. Kabir, H., ve Madduri, K. (2017, 29 May). *Parallel k-core decomposition on multicore platforms.* International Parallel and Distributed Processing Symposium Workshops, (pp.1482–1491), Lake Buena Vista, FL, USA.

164. Dasari, N. S., Desh, R., and Zubair, M. (2014, 27-30 October). *ParK: An efficient algorithm for k-core decomposition on multicore processors.* International Conference on Big Data, (pp. 9–16), Washington, DC, USA.

165. Kong, Y.X., Shi, G.Y., Wu, R.J., ve Zhang, Y.C. (2019). K-Core: theories and applications. *Physics Reports*, 832, 1–32.

166. Raghavan, U. N., Albert, R., ve Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3), 1-11.

167. Yan, R., Yuan, W., Su, X., ve Zhang, Z. (2023). FLPA: A fast label propagation algorithm for detecting overlapping community structure. *Expert Systems with Applications*, 234, 120971-120984.

168. Shaabani, E., Guo, R., ve Shakarian, P. (2018). *Detecting Pathogenic Social Media Accounts without Content or Network Structure*. 1st International Conference on Data Intelligence and Security, 57–64, South Padre Island, TX, USA.

169. Chen, H., Sultan, S. F., Tian, Y., Chen, M., ve Skiena, S. (2019, 3–7 November). *Fast and accurate network embeddings via very sparse random projection*. International Conference on Information and Knowledge Management, Proceedings, 399-408, Beijing, China.

170. Szabo, F. E. (2015). *The linear algebra survival guide: Illustrated with mathematica*. Netherlands: Academic Press, 101-115.

171. Atar, B., Bhaskara, K., Cook, A., Da Silva, S., Harada, M., Rajchgot, J., Van Tuyl, A., Wang, R., ve Yang, J. (2024). Hadamard products and binomial ideals. *Journal of Pure and Applied Algebra*, 228(6), 107568-107592.

172. Boyd, K., Eng, K. H., ve Page, C. D. (2013). Area under the precision-recall curve: Point estimates and confidence intervals. *Machine Learning and Knowledge Discovery in Databases*, 8190, 451–466.

173. Branco, P., Torgo, L., ve Ribeiro, R. P. (2017). A Survey of Predictive Modeling on Imbalanced Domains. *ACM Computing Surveys*, *49*(2), 1–50.

174. Shin, Y., Kim, K., Lee, J. J., ve Lee, K. (2021). ART: Automated reclassification for threat actors based on ATT&CK matrix similarity. *World Automation Congress Proceedings*, 15-20, Taipei, Taiwan.

175. Ahmed, N. K., Rossi, R. A., Zhou, R., Lee, J. B., Kong, X., Willke T. L. ve Eldardiry, H. (2018). Inductive Representation Learning in Large Attributed Graphs, *arXiv*.

176. Grover, A. ve Leskovec, J. (2016). node2vec: Scalable Feature Learning for Networks, *arXiv*.

177. Wu, W., Li, B., Luo, C. ve Nejdl, W. (2021). Hashing-Accelerated Graph Neural Networks for Link Prediction, *arXiv*.



Gazili olmak ayrıcalıktır