



**GÜVENLİK SİSTEMLERİ İZ BİLGİLERİNİN VERİ MADENCİLİĞİ
KULLANILARAK ETKİN ANALİZİ**

Işıl ÇİNAR

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

HAZİRAN 2015

Işıl ÇİNAR tarafından hazırlanan “GÜVENLİK SİSTEMLERİ İZ BİLGİLERİNİN VERİ MADENCİLİĞİ KULLANILARAK ETKİN ANALİZİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgisayar Mühendisliği Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Doç. Dr. Hasan Şakir BİLGE

Bilgisayar Mühendisliği (Mühendislik Fakültesi), Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum

.....

Başkan: Prof. Dr. Fatih Vehbi ÇELEBİ

Bilgisayar Mühendisliği, Yıldırım Beyazıt Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum

.....

Üye: Doç.Dr. Suat ÖZDEMİR

Bilgisayar Mühendisliği (Mühendislik Fakültesi), Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum

.....

Tez Savunma Tarihi: 24/06/2015

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Şeref SAĞIROĞLU

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

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

- 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 çalışmada 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.

Işıl ÇİNAR

24/06/2015

GÜVENLİK SİSTEMLERİ İZ BİLGİLERİNİN VERİ MADENCİLİĞİ
KULLANILARAK ETKİN ANALİZİ
(Yüksek Lisans Tezi)

Işıl ÇİNAR

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Haziran 2015

ÖZET

İz bilgileri (loglar) bilgisayar sistemlerinin çalışmasıyla ilgili çok önemli bilgiler içermektedir. Bu bilgiler performans ölçümü, istatistiksel tespitler, anormalliklerin tespiti, hata ayıklama, güvenlik tehditlerinin tespiti gibi işlemlerde kullanılmaktadır. Webde yer alan bilgilerin doğrusal olmayan bir biçimde hızla artışına paralel olarak web iz bilgilerinin analiz ihtiyacı da artmıştır. Son yıllarda gerçekleşen siber saldırıların büyük çoğunluğu uygulama katmanında webe yönelik ataklar olmaktadır. Bu tez çalışmasında veri madenciliği yöntemleri kullanılarak web iz bilgilerinin 3 aşamada analizi yapılmıştır. Birinci aşamada genel istatistiksel analiz, ikinci aşamada web robotlarının isteklerinin temizlenmesiyle sadece kullanıcılara ait iz bilgilerinin analizi ve son aşamada da saldırı tespit edilmesine yönelik analiz yapılmıştır. Sonuçların daha hızlı ve doğru bir şekilde elde edilebilmesi amacıyla her bir aşamanın amacına özgü veri ön işleme adımı gerçekleştirilmiştir. WEKA yazılımı kullanılarak, veri madenciliği yaklaşımlarının web verileri üzerinde uygulanmasını sağlayan web madenciliği teknikleri ile çeşitli çıkarımlarda bulunulmuştur. Saldırı girişimlerinin ve türlerinin tespiti için, WEKA'dan elde edilen sonuçlar ışığında iz bilgisi verisi filtrelenerek açık kaynak web saldırı tespit aracı olan Apache Scalp ile analiz edilmiştir. Apache Scalp aracı analiz sonuçlarında yanlış alarmlar üretebilmekte ve bunun sonucunda saldırı analistinin çok sayıda saldırı kaydını incelemesi gerekmektedir. Web madenciliği ile elde edilen örüntülerden yararlanılarak Apache Scalp ile üretilen incelenecek saldırı girişimleri sayısı %88.7 oranında azalırken, gerçekleştirilen analizin işlem süresi de %90.1 oranında kısalmıştır. Elde edilen 3 aşamalı analiz sonuçları doğrultusunda web sitesinin etkinliğini ve güvenilirliğini arttırmak için önerilerde bulunulmuştur.

Bilim Kodu : 902.1.014
Anahtar Kelimeler : Veri madenciliği, web madenciliği, istatistiksel analiz, saldırı tespiti, iz bilgisi analizi
Sayfa Adedi : 92
Danışman : Doç. Dr. Hasan Şakir BİLGE

EFFECTIVE ANALYSIS OF SECURITY SYSTEMS LOGS BY USING DATA
MINING METHODS

(M. Sc. Thesis)

Işıl ÇİNAR

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

June 2015

ABSTRACT

Log files contain crucial information related to the operations of computer systems. This information is used in processes such as performance measurement, statistical determinations, detection of anomaly, debugging and detection of security threats. The needs of analysis in web logs has increased in parallel with the information that increases in a non-linear manner in the web. In recent years, vast majority of the occurring cyber attacks are against web in application layer. In this thesis, web logs were analyzed in three stage by using data mining methods. In the first stage general statistical analysis, in the second stage analysis of trace information, which only belongs to the user, with cleaning requests of web robots and in the final stage for intrusion detection analysis was performed. In order to obtain results more quickly and accurately, the data preprocessing step which is appropriate for purpose of each stage is performed. There has been various inferences via web mining techniques which performs data mining approaches to web data by using WEKA software. In order to detect the intrusion attempts and types, log data, which is filtered according to results obtained from WEKA, is analyzed with Apache Scalp that is an open source intrusion detection tool. Apache scalp can produce false alarms in the analysis results, therefore attack analyst has to examine large number of attack records. As the number of intrusion attempts to be examined, which is produced via Apache Scalp, is decreasing by 88.7%, the processing time of performed analysis is also reduced by the rate of 90.1% by using the patterns obtained from web mining. In accordance with the obtained results from 3-stage analysis, some suggestions are offered to improve the effectiveness and reliability of the website.

Science Code : 902.1.014

Key Words : Data mining, web mining, statistical analysis, intrusion detection, log analysis

Page Number : 92

Supervisor : Assoc. Prof. Dr. Hasan Şakir BİLGE

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren danışman hocam Sayın Doç. Dr. Hasan Şakir BİLGE'ye, bu süreçte beni sabır ve özveri ile destekleyen eşim Muhammet Serkan ÇİNAR'a, manevi desteğiyle beni yalnız bırakmayan kızım ve aileme teşekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ	1
2. LİTERATÜR İNCELEMESİ.....	5
2.1. İz Bilgilerinin İstatistiksel Analizi ile İlgili Yapılan Çalışmalar	5
2.2. İz Bilgilerinin Güvenlik Açısından Analizi ile İlgili Yapılan Çalışmalar	8
3. WEB MADENCİLİĞİ	11
3.1. Web Madenciliği Sınıflandırması	12
3.1.1. Web içerik madenciliği	12
3.1.2. Web yapı madenciliği	13
3.1.3. Web kullanım madenciliği	13
3.2. Web Madenciliği Teknikleri	13
3.3. Web İz Bilgileri	14
4. WEB SALDIRI YÖNTEMLERİ	19
4.1. Web Saldırıları	19
4.1.1. SQL Enjeksiyon	21
4.1.2. Siteler arası betik yazma (XSS)	22
4.1.3. İhlal edilmiş kimlik doğrulama ve oturum yönetimi	24

4.1.4. Emniyetsiz doğrudan nesne referansı.....	24
4.1.5. Güvenlik ayarları hataları.....	25
4.1.6. Hassas veri teşhiri	25
4.1.7. İşlev seviyesi erişim kontrol eksikliği.....	25
4.1.8. Siteler Ötesi İstek Sahteciliği (CSRF)	25
4.1.9. Bilinen güvenlik açıklıkları ile bileşenleri kullanma	26
4.1.10. Geçersiz yönlendirme ve iletimler	27
4.2. WebGoat Programı ile Web Saldırıları Örnekleri.....	28
4.2.1. WebGoat programı ile SQL Enjeksiyon örnekleme.....	29
4.2.2. WebGoat programı ile XSS örnekleme.....	31
5. UYGULAMA	33
5.1. Önışleme.....	33
5.2. İz Bilgilerinin Analizi.....	39
5.2.1. Web madenciliği programları ile iz bilgisi analizi.....	41
5.2.2. WEKA ile örüntü keşfi ve analiz	50
5.2.3. WEKA ile saldırı tespiti.....	66
5.2.4. Apache Scalp ile iz bilgisi analizi	69
6. SONUÇ VE ÖNERİLER	79
6.1. İstatistiksel Sonuçların Değerlendirilmesi	79
6.2. Güvenlik Açısından Sonuçların Değerlendirilmesi.....	80
KAYNAKLAR	83
ÖZGEÇMİŞ	91

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Erişim dosyasından kesit	15
Çizelge 3.2. Başlıca durum kodları ve açıklamaları	16
Çizelge 3.3. İz bilgisi dosyaları ve tüm trafik analizinin karşılaştırılması.....	16
Çizelge 4.1. OWASP 10 güvenlik riski	21
Çizelge 4.2. Düzenli ifadeler	23
Çizelge 5.1. Erişimlerdeki veri sayıları.....	38
Çizelge 5.2. Erişim dosyası özellikleri	40
Çizelge 5.3. Hata günlüğü dosyası özellikleri	40
Çizelge 5.4. Erişim iz bilgisi dosyası istatistiksel sonuçlar	56
Çizelge 5.5. Hata iz bilgisi dosyası istatistikler	57
Çizelge 5.6. Erişim iz bilgisi dosyası istatistiksel sonuçlar	61
Çizelge 5.7. Scalp analiz sonuçlarının karşılaştırılması.....	76
Çizelge 5.8. Apache Scalp ile tespit edilen saldırı girişimleri	77

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 4.1. SQL Inject Me aracı.....	22
Şekil 4.2. ZAP ile sızma testi.....	24
Şekil 4.3. Geçersiz yönlendirme ve iletimler zafiyeti araması	28
Şekil 4.4. WebGoat anasayfası	29
Şekil 4.5. String SQL Enjeksiyon gerçekleştirmek üzere parametrelerin girilmesi	30
Şekil 4.6. String SQL Enjeksiyon sonucu veritabanında erişilen personel listesi	30
Şekil 4.7. Personel bilgileri güncelleme ekranı	31
Şekil 4.8. XSS gerçekleştirmek üzere betik yazılan ekran	32
Şekil 4.9. XSS atağının başarıyla gerçekleştirilmesi sonucu açılan uyarı mesajı.....	32
Şekil 5.1. Veri dosyasının excel ortamına aktarılması.....	34
Şekil 5.2. Önişleme aşamaları.....	35
Şekil 5.3. Gereksiz alanların silinmesi.....	35
Şekil 5.4. Silinen .png uzantılı dosyalar	36
Şekil 5.5. 400 serili durum kodlarını içeren kayıtlar	37
Şekil 5.6. Filtrenmesi gereken web robot istekleri.....	39
Şekil 5.7. Genel sistem mimarisi	41
Şekil 5.8. Web madenciliği programları ile iz bilgisi analizi uygulama şeması.....	42
Şekil 5.9. Ziyaretlerle ilgili genel istatistikler grafiği	43
Şekil 5.10. Ziyaret genel istatistikleri ile ilgili detaylı bilgi	44
Şekil 5.11. Aktivite istatistikleri özet bilgisi.....	44
Şekil 5.12. En çok kullanılan işletim sistemleri.....	45
Şekil 5.13. En çok kullanılan tarayıcılar	45
Şekil 5.14. En çok kullanılan arama robotları.....	46

Şekil	Sayfa
Şekil 5.15. En çok erişimin sağlandığı araç tipleri	46
Şekil 5.16. En çok kullanılan mobil cihazlar	47
Şekil 5.17. Günün saatlerine göre erişim analizi	48
Şekil 5.18. Erişimin sağlandığı IP numaralarına göre analiz.....	48
Şekil 5.19. En çok ziyaret eden ülkeler.....	49
Şekil 5.20. En çok erişimin sağlandığı şehirler.....	49
Şekil 5.21. Sayfaya erişimde en çok kullanılan arama motorları	50
Şekil 5.22. ARRF dönüştürücü program	52
Şekil 5.23. Bellek yetersizliği hatası.....	53
Şekil 5.24. DSN oluşturulması.....	54
Şekil 5.25. WEKA'dan MS Access veritabanına yapılan bağlantı ekranı.....	55
Şekil 5.26. WEKA'da önışlem penceresi	56
Şekil 5.27. WEKA'nın Visualize sekmesinde analiz edilen verinin görüntüsü	58
Şekil 5.28. WEKA'nın Visualize sekmesinde analiz edilen veriyle ilgili detaylı bilgi..	59
Şekil 5.29. Saldırı denemelerinde bulunan isteklerle ilgili görüntü.....	60
Şekil 5.30. Önışleme-2 verisi yayılım dağılımı	62
Şekil 5.31. Önışleme-2 verisi yayılım dağılımından elde edilen detaylı bilgi.....	63
Şekil 5.32. Kmeans kümeleme algoritması kullanılarak yapılan analiz	64
Şekil 5.33. WEKA'da işlenen verinin durum kodları ve sayıları	64
Şekil 5.34. Apriori algoritması çıktısı.....	65
Şekil 5.35. Önışleme-3 verisinin analizi	66
Şekil 5.36. Önışleme-3 verisinin visualize ekranından analizi	67
Şekil 5.37. Apriori algoritması ile elde edilen sonuçlar.....	68
Şekil 5.38. Kmeans ile elde edilen kümeler	69
Şekil 5.39. Scalp script çalıştırma ve sonuçları	71

Şekil	Sayfa
Şekil 5.40. Sonuç dosyasından bir kesit.....	73
Şekil 5.41. XSS ve SQL Enjeksiyon için çalıştırılan scriptler.....	73
Şekil 5.42. Tarihe göre filtrelenererek çalıştırılan script	74
Şekil 5.43. Filtreleme sonrası Scalp'dan elde edilen saldırı sonuçları kesiti.....	75
Şekil 5.44. Düzenlenmiş Scalp raporu çıktısı	76

SİMGELER VE KISALTMALAR

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

Kısaltmalar	Açıklama
ARRF	Attribute Relation File Format
CSRF	Cross Site Request Forgery
CWE	Common Vulnerabilities and Exposures
DoS	Denial of Service
DSN	Data Source Name
HTTP	Hypertext Transfer Protocol
IIS	Internet Information Service
IYTE	İzmir Yüksek Teknoloji Enstitüsü
LFI	Local File Inclusion
NCSA	National Center for Supercomputing Applications
ODBC	Open Database Connectivity
OWASP	Open Web Application Security Project
PHPIDS	Personal Home Page Intrusion Detection System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WEKA	Waikato Environment for Knowledge Analysis
WUM	Web Utilization Miner
WUMPREP	Web Utilization Miner Preprocessing
XSS	Cross Site Scripting
ZAP	Zed Attack Proxy

1. GİRİŞ

İnternet servisleri, web ve mobil uygulamalar günümüzde iletişimle ilgili ihtiyaçlarımızı geniş ölçüde karşılamakta, bununla birlikte muazzam miktarda veri üretilmektedir. Bu verilerin %90'ı önceden tanımlanmış bir yapı ve modelde değildir. Genellikle yapılandırılmamış veri, veri madenciliği ve analiz teknikleri uygulanmadığı sürece kullanışsız olmaktadır [1].

Veri madenciliği, basit ve net olmayan, önceden bilinmeyip tahmin edilemeyen, faydalı olan desenlerin ve bilginin çok büyük miktarlardaki veriden elde edilmesi işlemidir [2]. Veri madenciliği aslında bilgi keşfi sürecinin bir bölümü olarak kabul görmektedir. Bilgi keşfi sürecinin aşamaları aşağıda sunulmuştur [3].

- Veri Temizleme: Gürültülü ve tutarsız verilerin çıkarılması.
- Veri Bütünleştirme: Birden fazla veri kaynağının birleştirilmesi.
- Veri Seçme: Analize özel verilerin seçilmesi.
- Veri Dönüşümü: Verinin veri madenciliği yöntemlerinden kullanılabilecek forma dönüşümünün sağlanması.
- Veri Madenciliği: Veri örüntü ve bilgilerini tespit etmek için çeşitli metotları uygulamak.
- Örüntü Değerlendirme: Bazı ölçümlere göre elde edilmiş bilgiyi temsil eden ilginç örüntülerin tanımlanması.
- Bilgi Sunumu: Analizi yapılmış olan bilginin sunumunun gerçekleştirilmesi.

Sistemlerde ağ cihazları iz bilgileri, güvenlik duvarı iz bilgileri, saldırı tespit sistemleri iz bilgileri, web iz bilgileri gibi birçok çeşitte iz bilgisi tutulmaktadır. Son 10 yılda webde yer alan bilgilerde doğrusal olmayan çok hızlı bir artış söz konusu olmuştur. Bu büyümeyle webde yer alan bilgilerin analiz ihtiyacı da artmıştır [4]. Bununla birlikte son yıllarda gerçekleşen siber saldırıların yaklaşık %75'inden fazlası uygulama katmanında webe yönelik ataklar olmaktadır [5].

Web madenciliği, veri madenciliği teknikleri ile web belgelerinden ve servislerinden otomatik olarak bilginin ayıklanması, ortaya çıkarılması ve tahlil edilmesidir [6].

Web madenciliği ile kullanıcılar hakkında detaylı çıkarımlarda bulunulabilmekte, kullanıcıların eğilimlerine göre içerik düzenlenebilmekte, web sitesinin kullanılabilirliğini artırmaya yönelik iyileştirmeler yapılabilmekte ve anomali tespitleri yapılarak çeşitli güvenlik önlemleri alınabilmektedir. Son yıllarda e-ticaret ve çevrimiçi alışveriş hizmetlerinin çoğalmasıyla, bu alanda rekabet sonucu gerçekleştirilen çalışmalar, web madenciliğinin önemini fazlasıyla ortaya çıkarmaktadır [7]. Diğer taraftan kurumsal sayfalarda kurum çalışanlarına yönelik durum değerlendirmesi yapılabilmekte, kullanıcıların karakteristik özellikleri tahmin edilebilmektedir [8].

Web iz bilgilerinde her bir isteğe karşılık sunucunun verdiği başarılı, başarısız vb. cevaplar durum kodu olarak tutulmaktadır. Literatürdeki bazı çalışmalarda analizler için önışleme aşamasında bu durum kodlarından 400'li kodlar olarak da bilinen hata kodlarının bulunduğu satırların temizlendiği görölmüşür. Ancak güvenlik açısından analiz gerçekleştirileceği durumlarda özellikle 400 ve 500'li durum kodlarının incelenmesi gerekmektedir. Çünkü birçok saldırıda saldırgan sayfaya çeşitli istekler göndererek denemelerde bulunmakta ve bu denemeler başarısız olduğunda sunucu çoğu zaman arka arkaya 404 koduyla yanıt vermektedir. Saldırgan bir açıklık bulup deneme yaptığında ve başarısız olduğunda ise sunucu hata kodu olan 500'li durum kodlarıyla yanıt vermektedir. Literatürde web saldırı tespitinin gerçekleştirildiği birçok çalışmada bu durumun önışleme aşamasında gözardı edildiği görölmüşür. Bunun sonucunda hem analiz süresinin uzayacağı hem de hatalı sonuçlar alınabileceği düşünölmektedir.

Literatür taramalarında dikkat çeken bir diğere önemli nokta ise birçok çalışmada web robotları tarafından gerçekleştirilen girdilerin insan girdilerinden ayrılmamış olmasıdır. Ancak çoğu zaman; bu tez çalışmasında da oluşturulan ilk önışleme dosyasından elde edilen analiz sonuçlarında da gösterildiği gibi; web robotlarının girdileri insan girdileriyle neredeyse eşit oranda olmaktadır ve bu durum analiz sonucunda hatalı çıkarımlarda bulunulmasına sebep olmaktadır.

Literatür incelemelerinde özellikle web saldırı tespitine yönelik çalışmalarda saldırıların içeriği ve nasıl gerçekleştirildiğine dair çok fazla bilgi verilmediği görölmüşür. Ancak saldırı girişimlerini tespit ettikten sonra çıkan sonuçları yorumlayabilmek saldırıları bertaraf etmek açısından çok önemlidir. Bu tez çalışmasında saldırıları tespit ettikten sonra özellikle yanlış alarmları eleyebilmek, saldırıların etkisini anlayabilmek ve zafiyetleri

giderebilmek için saldırıların içeriği ile ilgili bilgi verildikten sonra WebGoat adlı açık kaynak uygulama ile en çok gerçekleştirilen saldırılar için örnekler sunulmuştur.

Bu tez çalışmasında web iz bilgilerinde istatistiksel analizi gerçekleştirmek üzere önışlemeden geçirilmiş olan veri Nihuo 4 ve Sawmill 8.7.5 web iz bilgisi analiz ve raporlama programları ile analiz edilmiştir. Web madenciliği ile veri analizi ve saldırı tespiti için WEKA yazılımı tercih edilmiştir. Kümeleme için Kmeans ve birliktelik kurallarını bulmak amacıyla Apriori algoritmaları kullanılmıştır. WEKA'dan elde edilen sonuçlara göre saldırı türü ve yerini tespit etmek amacıyla Apache Scalp aracı kullanılmıştır. Apache Scalp'ı çalıştırmak ve Linux komutlarıyla analizi yapabilmek için bir sanallaştırma yazılımı olan Vmware Workstation 9.0 kurularak üzerine Linux tabanlı bir güvenlik kontrol işletim sistemi olan BackTrack 5-r3 kurulmuştur.

Bu çalışmanın literatüre katkısı özellikle web sunucu iz bilgileri önışlemesi aşamasında web kullanım madenciliği ve web saldırı tespiti için yapılan önışlemelerin farklarının ortaya konulması amacıyla 3 aşamada önışlemenin gerçekleştirilmesi ve her bir aşama için analiz sonuçlarının sunulması olmuştur. Böylece web iz bilgilerinden istatistiksel analiz elde etmenin yanı sıra web saldırı tespitinde önışlemenin önemi ortaya konulmuştur.

Bu tezin amacı, karmaşık ve düzensiz olan web iz bilgilerinden web madenciliğinin amacına yönelik olarak gerçekleştirilecek önışleme ile yapılacak analizlerle hem istatistiksel hem de güvenlik açısından aşağıda sunulan çıkarımlarda bulunmaktır:

1. Gerçekleştirilen istatistiksel analizler neticesinde kullanıcı eğilimleri ve genel site kullanımı hakkında elde edilen bulgularla site geliştiricilerine web sitesinin iyileştirilmesine ve geliştirilmesine yönelik katkıda bulunmak amaçlanmıştır.
2. Web madenciliği yöntemleri ile web saldırı tespitlerini çok daha kısa sürede ve yanlış alarm oranını azaltarak gerçekleştirmek ve elde edilen bulgular neticesinde web sitesinin zafiyetlerini ortaya çıkararak alınabilecek güvenlik önlemleri hakkında çözüm önerileri sunmak hedeflenmiştir.

Bu tez çalışmasının 2. Bölümünde, iz bilgilerinin analizi istatistiksel ve güvenlik olmak üzere 2 başlık altında ele alınarak literatürdeki çalışmalara yer verilmiştir. 3. Bölümde,

web madenciliđi, sınıflandırmaları, teknikleri ve web iz bilgilerinin yapısı ve içeriğinden detaylı olarak bahsedilmiştir. 4. Bölümde web saldırı yöntemleri anlatılarak WebGoat Programı ile örnekler sunulmuştur. 5. Bölümde genel sistem mimarisi sunularak web madenciliđi yöntemleri ile gerçekleştirilen uygulamaya yer verilmiştir. 6. Bölümde ise tez çalışmasının sonucuna yer verilmiştir.

2. LİTERATÜR İNCELEMESİ

İz bilgisi dosyaları bilgisayar sisteminin tarih kitabı olarak kabul edilmektedir [9]. İz bilgileri sistemin çalışmasıyla ilgili önemli bilgiler içermektedir. Bu bilgiler genellikle performans ölçümü, işlemsel profil tespiti, anormalliklerin tespiti, hata ayıklama, güvenlik tehditlerinin tespiti gibi işlemlerde kullanılmaktadır. Son zamanlarda çeşitli veri madenciliği ve makine öğrenme algoritmaları iz bilgisi dosyalarındaki bilgileri analiz etmek üzere yoğun olarak kullanılmaktadır [10].

İz bilgisi dosyaları çok fazla bilgi içermektedir, ancak kullanılan dosya formatından bu bilgilerden çıkarım yapmak oldukça zordur. Dolayısıyla verileri analiz etmeye yarayacak bir araca gereksinim duyulmaktadır [11].

Önceleri iz bilgisi analizleri sistem yöneticileri tarafından manuel olarak gerçekleştirilmiştir. Bu durum bazı olayların gözden kaçmasına sebep olabileceği gibi; her geçen gün artan veri miktarı manuel yöntemlerle analizin yetersiz kalmasına sebep olmuş ve analistler çeşitli araçlar geliştirmeye başlamışlardır [12].

Son yıllarda iz bilgisi yönetim sistemleri kurumlar tarafından yaygın olarak kullanılmaya başlanmıştır. Bazı şirketler (IBM, MacAfee ve Splunk gibi) kendilerine özel iz bilgisi yönetim çözümlerini sunmaktadır. Ancak bu sistemlerin uygun donanım gerektirmesi ve iz bilgisi verilerini analiz etmek üzere web kullanım madenciliğini içermemesi problem olmaktadır [13].

İz bilgisi analiziyle ilgili gerçekleştirilen literatür çalışmalarının büyük bir kısmında web iz bilgileri analiz edilmiştir. Bu tez çalışmasında da iz bilgisi analizi web iz bilgileri üzerinde gerçekleştirilmiş ve literatür taramalarına bu yönde ağırlık verilmiştir.

2.1. İz Bilgilerinin İstatistiksel Analizi ile İlgili Yapılan Çalışmalar

Web sayfalarına giriş isteklerinin bilinmesi, hangi sayfalara daha sık girildiği, hangi sayfaların birlikte ziyaret edildiği gibi bilgiler istatistiksel yöntemlerle kullanıcı

davranışlarını öğrenmeyi ve bu sayede site ile ilgili düzenlemeler yapmayı sağlamaktadır [14].

Web iz bilgisi dosyaları üzerinde web madenciliği konusunda yapılan çalışmaların önemli bir kısmını önışleme adımı oluşturmaktadır. Web iz bilgisi dosyalarında gereksiz bilginin çok büyük yer tutması, önışlemeyi gerekli kılmaktadır. Bu işlem özellikle web site yöneticileri ve kullanıcıları tarafından web iz bilgisi dosyalarının yönetimini kolaylaştırmaktadır.

Hussain, Asghar ve Masood (2010) çalışmalarında web iz bilgisi dosyalarını; client iz bilgisi dosyası, proxy iz bilgisi dosyası ve sunucu iz bilgisi dosyası olarak ele almışlardır. Çalışmada iz bilgisi dosyalarındaki ön işleme sürecinin tüm veri madenciliği sürecinin %80'ini kapsadığı belirtilmiştir. Önışleme adımı daha sonraki adımlar olan örüntü keşfi ve örüntü analizinin daha etkili ve kolay yapılmasını sağlamaktadır. Web iz bilgisi dosyalarında en sık kullanılan ortak özellikler IP adresi, tarih, saat, URL ve User-agent olarak belirtilmiştir [15].

Özakar ve Püskülcü (2002) çalışmalarında web içerik ve web kullanım madenciliği tekniklerinin entegrasyonu ile oluşmuş bir veritabanından nasıl yararlanılabileceğinden bahsetmektedirler. Çalışmada IYTE web madenciliği sistem mimarisi çıkarılmıştır. Çalışmanın amacı IYTE web sunucusu üzerinde biriken web erişim, hata, kullanıcı dosyalarındaki veriyi site içerik verisi ile bir araya getirerek analiz etmek ve kullanım örüntülerini tespit etmektir. Sistem, “Veri Hazırlama” ve “Sorgu Mekanizması” olmak üzere iki ana bölümden oluşmaktadır. Bu çalışmada önemle üzerinde durulan nokta analiz edilen verinin yorumlanmasından çok verinin bulunduğu sistemin mimarisinin tasarımı ve geliştirilmesidir [16].

Diğer bir çalışmada, özellikle web kullanım madenciliği konusu üzerinde durularak Sisodia ve Verma (2012) tarafından 1 aylık süre için NTR web sunucu iz bilgileri web iz bilgisi keşif programıyla analiz edilmiştir [17]. Web iz bilgisi analizcisi bilinen veri madenciliği tekniklerini erişim iz bilgilerine uygulayarak çeşitli web kullanım erişim modellerini oluşturmuştur. Bu analizler istatistiksel tanımlamalar ve birliktelik kurallarını içermektedir. Analiz sonucunda, kullanıcıların erişim modelleri, band genişliği tüketimi, haftalık ziyaretçi sayısı gibi bilgiler alınmıştır. Bu çalışma alanında yapılan araştırmada,

ana zorlayıcı noktanın web robotları tarafından gerçekleştirilen girdilerin insan girdilerinden ayrılması konusu olduğu belirtilmiştir [17].

Bir başka çalışmada ise e-ticaret alanında web iz bilgisi madenciliği çalışması yapılmıştır. Yapılan çalışmada Apriori algoritması kullanılmış ve daha önce 2005 yılında geliştirilen “Improved AprioriAll” Algoritması ile performans yönünden karşılaştırması yapılmıştır. E-web Miner adı verilen çalışmanın zaman bakımından ve sonuçların doğruluğu bakımından diğer algoritma ile karşılaştırıldığında çok daha başarılı olduğu görülmüştür. Çalışmada küçük bir veri seti ile çalışıldığı belirtilmiştir, Apriori algoritmasının büyük veri tabanlarında performansının azaldığı için, bu çalışmanın çok büyük veri tabanlarında yavaş kalabileceği düşünülmektedir [18].

Web kullanım madenciliği ile terör saldırıları takibi gerçekleştirilen başka bir çalışmada [19] doğru ve kaliteli analiz yapabilmek için web madenciliğinin ön işleme adımı kullanılmıştır. İlgisiz verilerin temizlenmesi ve veri azaltma işlemi 2 açıdan ele alınmıştır. İlki birçok incelenen çalışmada olduğu gibi .gif, .jpeg, .css vb. URL isteklerinin temizlenmesi adımıdır. İkinci temizleme işlemi ise hata durum kodları ele alınarak gerçekleştirilmiştir. Hatalı isteklerin madencilik işinde çok kullanışsız olduğu, durum kodlarının kontrol edilerek bu isteklerin temizlenmesi gerektiği belirtilmiştir. Örneğin; durum kodu 404 ise bu isteğin ulaşmaya çalıştığı kaynağın bulunmadığı anlamına gelmektedir. Bu iz bilgisi satırlarının temizlenmesi gerektiği savunulmuştur [19]. Hatalı kodların tespiti için 400 ve 500 durum kodları arasında yer alan isteklerin belirlenmesi gerekmektedir [20].

İsmail Haberal (2007), Başkent Üniversitesi web sitesinin geliştirilmesi amacıyla web günlük erişimlerini veri içindeki sınıflandırma bilgisini veren java platformunda geliştirilmiş WEKULA yazılımı ile gerçekleştirerek; serbest kullanıma sunulmuş olan WUM, WUMPREP ve WEKA yazılımları ile veri önışlemesini gerçekleştirmiştir [14].

Daş (2008) yaptığı çalışmada yol analizi yöntemi ile web kullanıcı erişim kütük dosyalarından bilgi çıkarımı yapmış ve birliktelik kuralları yöntemi ile web sayfaları arasındaki ilişkileri belirlemiştir. İstatistiksel analiz yöntemi ile Nihuo ve Web iz bilgisi analiz programlarını kullanarak web sitesinin genel kullanımına ilişkin detaylı bilgiler elde etmiştir [8].

Mustafa Turan (2011), web madenciliği teknikleriyle makine öğrenme tekniklerini birlikte kullanarak hibrit bir yapı geliştirmiştir. Web sayfalarındaki verileri yapısal olarak incelemenin yanında metin analiz işlevlerini de gerçekleştirmiştir [21].

Güncel Sarıman (2011), mevcut yazılımlarla gerçekleştirilen web erişim kütük analizinin daha kısa sürede gerçekleştirilebilmesi için paralel programlama teknikleri kullanmıştır. Sonuçta istatistiksel analiz gerçekleştirilmiştir. Paralel ve seri algoritmalar çalıştırılarak analizler süre bakımından karşılaştırılmıştır [22].

Kadir Can Burçak (2012), Kırıkkale Üniversitesi web sunucularına ait kullanıcı erişim günlüklerinde önışleme gerçekleştirerek iz bilgisi analizini Nihuo Web İz Bilgisi Analiz Programı ile gerçekleştirmiş ve site kullanımına ait istatistiksel sonuçlar elde etmiştir [11].

2.2. İz Bilgilerinin Güvenlik Açısından Analizi ile İlgili Yapılan Çalışmalar

Literatür taramalarında iz bilgileri üzerinde analizler gerçekleştirilirken özellikle web saldırılarına yönelik çalışmaların yapıldığı görölmüştür.

Bilişim sistemlerine yönelik saldırılar aktif ve pasif olmak üzere temelde iki yöntemle belirlenmektedir. Aktif saldırı belirleme ve engelleme sistemleri genellikle ağ/host tabanlı çalışmakta olup; anlık ağ trafiği ya da işletim sistemi fonksiyonlarını kullanarak saldırıları belirlemekte ve engellemektedir [23]. Pasif saldırı belirleme sisteminde ise iz bilgileri incelenmektedir. Saldırıların büyük bir kısmı iz bilgilerindeki anormallikler incelenerek belirlenebilmektedir [23].

Vigna, G. ve diğerleri (2003) web sitelerinin ve web uygulamalarının hızla büyümesiyle uygulamaların daha karışık bir hale geldiği ve buna paralel olarak web hacking faaliyetlerinin de arttığını belirtmektedirler. Yaptıkları çalışmada gelen isteklerin görüntülenmesi ve analiz edilmesi için web saldırı tespit aracı geliştirmişler ve bu aracı güvenlik mimarisine entegre ederek kullandıklarını göstermişlerdir [24].

Auxilia ve Tamilselvan (2010) ise makalelerinde önemli web ataklarını; “Sofistike HTTP Saldırıları”, “SQL Enjeksiyon Saldırıları” ve “XSS” başlıkları altında incelemişler ve negatif güvenlik modeli geliştirmişlerdir. Bu çalışmanın negatif güvenlik modeli olmasının

sebebi gelen isteklerin anormallığı, sıradışı hareketleri ve bilinen web uygulama saldırılarını izlemesi olarak belirtilmiştir [25].

Yine saldırıların tespitine yönelik yapılan bir başka çalışmada üniversite yerel alan ağı için ağ saldırılarını tespit etmek üzere web iz bilgisi madenciliği kullanılmıştır. Çok büyük iz bilgisi dosyalarındaki anormal verileri tespit etmek için “sapan veri tespiti” kullanılmıştır. Ağ saldırıları “servis kaynaklarını tüketmeye yönelik saldırılar” ve “zararlı çeşitli illegal bilgilerin elektronik posta sunucular aracılığı ile gönderilmesi ile önemli bilgilere zarar verilmesi” başlıkları altında ele alınmıştır. Bu çalışmada anomali bulma teknikleri; istatistiksel, uzaklık tabanlı ve göç tabanlı olarak sınıflandırılmıştır. Uygulamada uzaklık tabanlı anomali tekniği kullanılmıştır ve k en yakın komşu algoritması kullanılarak işlem yapılmıştır [26].

Elmas Yıldız (2010) çalışmasında DoS ataklarından biri olan Brute Force tipi bir saldırının veri madenciliği teknikleriyle tespit edilip engellendiği bir saldırı tespit uygulaması tasarlamıştır [27].

Salama, Marie, El-Fargary ve Helmy (2011) web saldırı tespiti için web sunucu iz bilgileri önışlemesi ile ilgili yaptıkları çalışmada web saldırı ve web kullanım madenciliği için yapılan önışlemelerin farklarını açıklayarak saldırı tespitinde önışlemenin öneminden bahsetmişlerdir [28].

Patil (2012) web iz bilgilerinin önışlenmesiyle kötüye kullanım ve anomali tabanlı saldırı tespiti üzerine bir çalışma gerçekleştirmiştir. Apriori ile birliktelik kuralları oluşturarak normal olmayan davranışları select, insert, <script> vb. SQL ve XSS saldırısına işaret eden anahtar kelimeleri kullanarak tespit etmiştir [29].

Gržinic, Kišasondi ve Šaban (2013), Apache web sunucu iz bilgilerinin önışleme aşamasında sadece 400 ve 500 arasındaki durum kodlarının yer aldığı satırları alarak web madenciliği işlemleri için WEKA programında yer alan çeşitli sınıflandırma algoritmalarıyla anomali sınıflandırması gerçekleştirmişlerdir [20].

Mabzool ve Lighvan (2014) web madenciliği teknolojisini kullanarak, çevrimdışı saldırı tespit sistemini gerçekleştirmek üzere Kmeans kümeleme algoritmasını kullanarak anomali davranışları tespit etmişlerdir [30].

3. WEB MADENCİLİĞİ

Web madenciliği teriminin ilk kez Etzioni tarafından 1996 yılında kullanıldığı bilinmektedir. Etzioni web madenciliğini, “World Wide Web’de bulunan dosya ve servislerden veri madenciliği tekniklerinin kullanılmasıyla otomatik olarak örüntüler bulmak ve öngörülme bilgiye ulaşmak” şeklinde tanımlamıştır [31].

Web madenciliğinde yoğun olarak kullanılan temel terimler bulunmaktadır. Çalışma içerisinde de geçmekte olan ve World Wide Web Konsorsiyumu’nun (W3C) önerdiği [32,33] önemli web terimleri aşağıda sunulmuştur:

Web Tarayıcı: İnternet üzerindeki sayfalarda gezinmeyi ve bilgi kaynaklarını aramayı sağlayan istemci yazılımlarıdır (Mozilla, Opera, İnternet Explorer vb.).

Web Kaynağı: HTTP protokolleri kullanılarak ulaşılabilen herhangi bir İnternet kaynağıdır.

Web Sunucu: Hazırlanan web materyallerini internet üzerinde erişim hizmetine açan yüksek hızda çalışan bilgisayarlardır.

Web Sayfası: ASP, HTML, PHP vb. internette yayınlanabilecek farklı uzantılardaki dokümanlardır.

Web Sitesi: Web sunucusu tarafından internet’e sunulan veritabanları ile bağlantılı belge ve dosyalardır.

Kullanıcı: İnterneti kullanan kişilerdir.

Web İsteği: Kullanıcının bir web kaynağına yapmış olduğu isteklerdir.

Ziyaret: Belli bir zamanda kullanıcı oturumu esnasında yapılmış olan web sayfalarının görüntülenmesidir.

Kullanıcı Oturumu: Sunucu üzerinde tanımlanan ölçüde bir kullanıcının belirli bir zamanda arka arkaya gerçekleştirdiği web istekleridir.

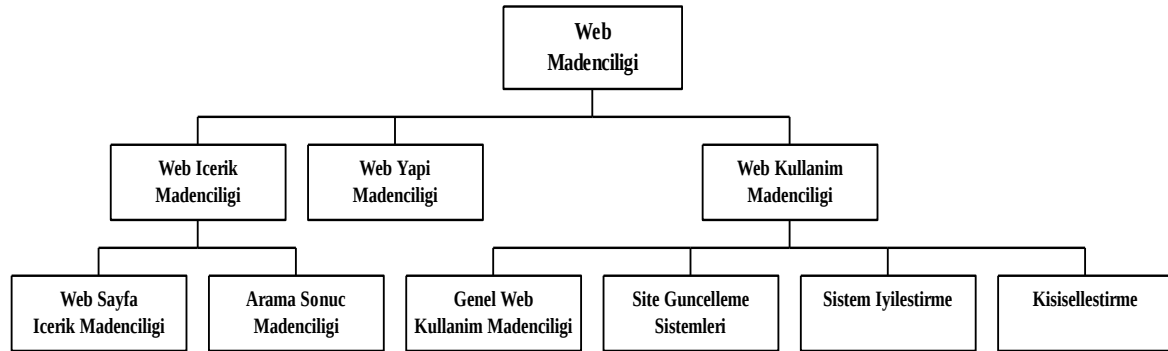
URI (Uniform Resource Identifier): İnternet web sayfalarının fiziksel kaynağını tanımlayan karakterler zinciridir.

Oturum Tanımlama: Kullanıcının bir web sitesinde oturum açıp kapattığı zamana kadar geçen sürenin tanımlanarak sınıflandırılmasıdır.

Kullanıcı Tanımlama: Siteye istekte bulunan kullanıcıların tarayıcıları ve siteyi kullanma özelliklerine göre sınıflandırılmasıdır.

3.1. Web Madenciliği Sınıflandırması

Web madenciliği genel olarak 3 araştırma alanından oluşmaktadır; içerik madenciliği (content), web kullanım madenciliği (usage) ve web yapı madenciliği (structure) [34]. Şekil 3.1’de web madenciliği sınıflandırması görülmektedir.



Şekil 4.1. Web Madenciliği Çalışmalarının Sınıflandırılması [16]

3.1.1. Web içerik madenciliği

Web içerik madenciliği web kaynaklarından otomatik bilgi arama tekniklerini tanımlamaktadır. Verinin farklı tiplerde oluşu ve yapısal olmayışı bu konudaki teknikleri karmaşıktırıştır [16]. Amaç erişilebilir web dokümanları içerisindeki saklı bilgiyi ortaya çıkarmaktır [14].

İçerik; kullanıcıların web sayfalarında eriştiği ve kullandıkları grafik, resim, metin, şekil, ses ve görüntü dosyaları gibi gerçek verilerdir [17].

3.1.2. Web yapı madenciliği

Web yapı madenciliği ile web sitesi ve web sayfası hakkında bağlantı verisine bakarak bilgi üretilmektedir. Teknik olarak, web içerik madenciliği dokümanın içeriğine, yapı madenciliği ise dokümanlar arası bağlantılara yoğunlaşmaktadır [16].

Yapı; web sitesi içeriğinin organizasyonunu gösteren veridir. Web sitesindeki yapı verisi, tasarımcı bakış açısını göstermektedir. Örneğin, web sitesinde bulunan sayfalar arasındaki bağlantı bilgisini, HTML (Hyper Text Markup Language) ve XML (Extensible Markup Language) dokümanları ağaç yapısını gösterir. Web sitesi yapı verisi, site haritalama araçları ile otomatik olarak oluşturulan sitenin harita bilgisidir [35].

3.1.3. Web kullanım madenciliği

Web kullanım madenciliği kullanıcıların webde dolaşırken yaptıkları erişim hareketleri sonucu oluşturulan veriden bilgi üretmeyi hedeflemektedir. Şekil 3.1’de görüldüğü gibi 4 ana başlık altında incelenmektedir.

Kullanım bilgileri içerisinde kullanıcının IP adresi, sayfa referansları, bağlantı zaman bilgileri, kullanıcının İnternet tarayıcısının adı ve sürümü gibi bilgiler yer almaktadır [17].

3.2. Web Madenciliği Teknikleri

Yaygın olarak kullanılan web madenciliği teknikleri aşağıda sunulmuştur [14]:

İstatistik; Bir web sitesi ve ziyaretçileri hakkında bilgi açığa çıkarmaya yaramaktadır. Böylece web sayfalarındaki güvenlik sorunları, sistem performansı vb. konularda bilgi edinilebilmektedir.

İlişkilendirme Kuralları; Web madenciliğinde daha çok alışveriş uygulamalarında sepet analizi amacıyla kullanılmaktadır.

Sıralı Desen; Kullanıcı oturumları arasında desen bulunmaya çalışılmaktadır. Belirli zaman aralıklarındaki oturumlar incelenmekte ve eğilim analizi, değişen nokta bulma veya benzerlik analizleri gerçekleştirilmektedir.

Kümeleme; Kullanıcı grupları ve sayfa grupları olmak üzere web madenciliğinde 2 kümeleme yaklaşımı bulunmaktadır. Kullanıcı grupları ile benzer sayfa görüntülemesi yapan kullanıcılar tespit edilir ve bir grupta toplanır. Sayfa grupları ile de benzer içerikli sayfalar arama motorları için bir arada gruplandırılmaktadır.

Sınıflandırma; Bir veriyi daha önceden tanımlanmış sınıflara dağıtma işlemidir.

3.3. Web İz Bilgileri

Web iz bilgisi dosyaları web sitelerinin kullanımıyla ilgili verileri depolamaktadır. Bu verilerin analizi son derece önemlidir. Genellikle web iz bilgisi dosyaları farklı formatlarda kaydedilmekte bu durum analiz edilmelerini zorlaştırmaktadır [36].

Bir web sunucuda 4 çeşit iz bilgisi dosyası bulunmaktadır [11].

1. *Sunucu iz bilgileri (Erişim iz bilgileri)*: Bir İnternet sayfasını görüntülemek için web sunucusuna gönderilen istekle ilgili bilgileri içeren kayıttır. Bulunduğu işletim sistemine bağlı olarak farklılık gösterebilmektedir.
2. *Hata iz bilgileri*: Web sunucu üzerinde gerçekleşen hataların tutulduğu kayıt dosyasıdır.
3. *İstek kayıt dosyaları*: İnternet_kullanıcısının sayfa isteklerinin tutulduğu kayıt dosyasıdır.
4. *Etmen kayıt dosyaları*: Kullanıcıya ait istemci bilgisayara ait bilgilerin tutulduğu kayıt dosyalarıdır.

World Wide Web konsorsiyum web sunucu iz bilgileri için standart formatı belirlemektedir; ancak farklı formatlar da bulunmaktadır. Örneğin; IIS, 6 farklı iz bilgisi dosyası formatını IIS tabanlı sitelerde ve servislerde bilgilerin izlenmesi ve analiz edilmesi için sağlamaktadır. Bu formatlar aşağıda listelenmiştir [17]:

- 1- W3C Extended Log File Format
- 2- W3C Centralized Logging
- 3- NCSA Common Log File Format
- 4- IIS Log File Format
- 5- ODBC Logging
- 6- Centralized Binary Logging

Genel olarak kaydedilen verinin miktarına göre değişen iki format mevcuttur; “Common Log Format” ve “Combined Log Format”. Çalışılan erişim dosyasının kayıt formatı “Combined Log Format”dır. Çizelge 3.1’de erişim dosyasından bir kesit gösterilmiştir.

Çizelge 3.1. Erişim dosyasından kesit

IPNO	TRH&SAAT	ISTEKURL	DRM	BYT	BASVYER	TARAYICI
66.249.71.181	[26/Aug/2012 :06:51:59 +0300]	GET /10957-alt-- HTTP/1.1	200	18512	-	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
66.249.71.181	[26/Aug/2012 :06:54:10 +0300]	GET /11001-xxxx- balikcisi HTTP/1.1	200	18693	-	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
213.186.127.13	[26/Aug/2012 :06:54:13+0300]	GET /xxxx- restoranlari?start=40 HTTP/1.1	200	18793	-	Mozilla/5.0 (compatible; AhrefsBot/3.1; +http://ahrefs.com/robot/)

66.249.71.181: İstekte bulunan tarayıcının IP adresidir. Bazı durumlarda bu alanda gelen tarayıcının alan adı da bulunabilmektedir.

[26/Aug/2012:06:51:59 +0300]: İsteğin yapıldığı saat ve tarihtir.

GET /11001-xxxx-balikcisi HTTP/1.1: İlk kısım istek türünü (bu örnekte Get ve Post) ikinci kısım istenilen dosyayı (burada 11001-xxxx-balikcisi) ve son kısım protokolü vermektedir (burada HTTP/1.1).

200: Bu kısım sunucunun verdiği yanıtı göstermektedir. ‘200’ kodu hiç bir sorun oluşmadığını göstermektedir. Bu kodlar RFC2616 teknik belgesinin 10. kısmında belgelendirilmiştir. Çizelge 3.2’de bu kodlar gösterilmiştir.

Çizelge 3.2. Başlıca durum kodları ve açıklamaları

Durum Kodu	Açıklaması
200	Tamam
201	Oluşturuldu
202	Onaylandı
302	Geçici taşındı
303	Diğerlerine bak
304	Nitelenemedi
400	Kötü istek
401	Yetkisiz
403	Yasaklandı
404	Sayfa bulunamadı
500	İç sunucu hatası

18512: Gönderilmiş olan dosyanın Byte cinsinden boyutu.

BASVYER: İsteğe nereden ulaşıldığı konusunda bilgi vermektedir.

Mozilla/4.0 (compatible; MSIE 6.0; Windows 98): İstekte bulunan tarayıcı bilgileri.

Web sunucular üzerindeki erişim ve hata kayıt dosyalarında kullanıcının site içinde gezinirken yaptığı her bir tıka karşılık bir ya da birden çok hareket kaydı tutulmaktadır. Böylece dosyalar hızla büyümekte ve yer kazanmak için periyodik olarak bu dosyalar temizlenmektedir [37]. Ancak; bu veriler analiz edildiğinde çeşitli istatistiksel sonuçlar çıkarılabileceği gibi güvenlikle ilgili önlemler almak ve saldırıları tespit etmek mümkün olabilmektedir.

Tez çalışmasında tüm trafiğin incelenmesinden ziyade, iz bilgisi dosyası incelenmesi gerçekleştirilmiştir. İz bilgisi dosyalarının incelenmesinin, tüm trafiğin incelenmesine göre getirdiği avantajlar Çizelge 3.1’de sunulmuştur [38]:

Çizelge 3.3. İz bilgisi dosyaları ve tüm trafik analizinin karşılaştırılması

	Avantaj	Dezavantaj
İz Bilgisi Dosyaları	Veri kolaylıkla erişilebilir durumdadır (dosyalarda).	İz bilgileri tüm verinin belli bir kısmından oluşmaktadır.
Tüm Trafik	Tüm bilgi analiz edilebilir.	Öncelikle verinin elde edilmesi gerekmektedir. Verinin birleştirilmesi, ayrıştırılması, normalize edilmesi gerekmektedir. Şifrelenmiş trafik, yüksek trafik yükü vb. durumlardan dolayı verinin elde edilmesi zorlaşabilmektedir. Ek donanım ve yüksek maliyet gerektirmektedir.

4. WEB SALDIRI YÖNTEMLERİ

Web uygulamalarına yönelik gerçekleştirilen saldırılar günümüzde oldukça artmıştır. Bu durum güvenlik duvarları, saldırı tespit sistemlerinin yanında webe yönelik saldırıların iz bilgilerinden analiz edilmesi ihtiyacını doğurmaktadır. Web saldırıları çok çeşitli ve karmaşık olduğu için web saldırılarının neler olduğuna, nasıl gerçekleştirildiğine ve bu saldırılardan korunma yöntemlerinin neler olduğuna ilerleyen bölümlerde yer verilmiştir.

4.1. Web Saldırıları

İz bilgisi analizi gerçekleştirilirken istatistiksel sonuçlar elde etmek kullanıcılara geri dönüş, web sayfasının etkin kullanımı, sayfa yapısı vb. açısından çok önemli olduğu gibi saldırılara yönelik analizlerin gerçekleştirilmesi de büyük önem taşımaktadır. Bu konuyla ilgili birçok ticari, açık kaynak araçlar mevcut olsa da çoğu zaman bu araçlar yanlış alarm üretmekte ve kullanıcı gözüyle analiz gerekmektedir. Güvenlik duvarları ve saldırı tespit sistemleri de web uygulamalarını korumaya yönelik tasarlanmışlardır ancak; güncel saldırılar sadece 80 (HTTP) ve 443 (HTTPS) portlarının engellenmesiyle bertaraf edilememektedir. Web uygulama iz bilgisi dosyaları kullanıcı hareketleri konusunda detaylı analiz sağlamaktadır [38]. Öncelikle iz bilgisi analiziyle ilgili yapılan literatür çalışmalarında web uygulamalarına yönelik saldırıların çok yaygınlaştığı görülmüştür.

Saldırıların tespitine yönelik “Statik Kurallar” ve “Dinamik Kurallar” olmak üzere 2 strateji bulunmaktadır [38].

Statik kurallar (kural tabanlı tespit): Analiz yapılmadan önce tanımlanmış olan kurallardır. Statik kurallar bir kere tanımlanmaktadır ve tespit süresince değişmeden kalmaktadır. Bu kurallar girdi karakterleri, parametreler için uygun uzunluk veya transfer miktarı için en üst limit olabilmektedir. Negatif güvenlik modeli ile saldırı olarak kabul edilen durumlar dışındaki tüm durumlara izin verilmektedir. Pozitif güvenlik modelinde ise sadece hangi durumların izinli olduğu tanımlanmakta ve bu durumlar haricindeki herşeyin geçişi engellenmektedir.

Dinamik kurallar (anomali tabanlı tespit): Kurallar statik değildir. Öğrenme sürecinde tanımlanmaktadır. Başlangıçta saldırı barındırmayan temiz bir rol model ele alınmakta ve normal modelin dışına çıkılan durumlarda alarm üretilmektedir.

Open Web Application Security Project (OWASP), 2007, 2010 ve 2013 yıllarında “OWASP TOP 10” başlığı altında web uygulamaları güvenliğine yönelik kritik riskleri açıklamıştır [39,40]. İncelenen birçok çalışmada OWASP’ın hazırlamış olduğu bu kritik riskler temel alınmıştır.

OWASP’ın açıkladığı liste genellikle web uygulamaları açıklık değerlendirmelerindeki minimum standart olarak kullanılmaktadır. Bunun yanında web uygulamaları açıklıkları sınıflandırmaları Common Vulnerabilities and Exposures (CWE) veritabanı ve Web Application Security Consortium (WASC) da bulunmaktadır [41-43].

2013 yılında, OWASP ve CWE, XSS ve SQL Enjeksiyon açıklıklarını web tabanlı sistemlerdeki en ciddi açıklıklar olarak raporlamıştır [40,41]. Bu ataklar çok güçlü olmalarının yanı sıra kolay uygulanabilen saldırı teknikleridir. Web uygulamalarına yönelik gerçekleştirilen bu saldırıların giderilmesine yönelik yapılan çalışmalarda karşılaşılan en önemli problemlerden biri de yanlış pozitif ve yanlış negatif alarmlarla sıkça karşılaşılmasıdır [44].

OWASP’ın açıkladığı en önemli 10 güvenlik riski ve bu risklerin giderilmesine yönelik en faydalı araçlar Çizelge 4.1’de sunulmuştur [45].

Çizelge 4.1. OWASP 10 güvenlik riski

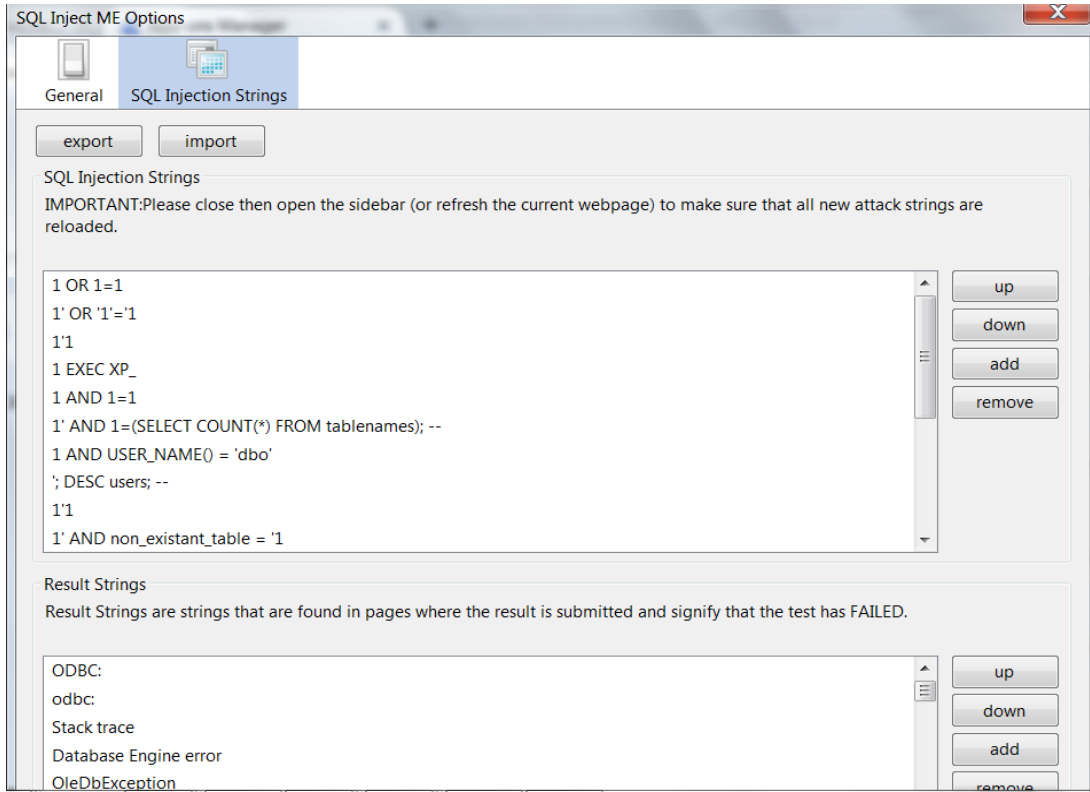
OWASP 10 Güvenlik Riski
SQL Enjeksiyon
Siteler Arası Betik Yazma (XSS)
İhlal edilmiş kimlik doğrulama ve oturum yönetimi
Emniyetsiz doğrudan nesne referansı
Güvenlik ayarları hataları
Hassas veri teşhiri
İşlev seviyesi erişim kontrol eksikliği
Siteler Ötesi İstek Sahteciliği (CSRF)
Bilinen güvenlik açıklıkları ile bileşenleri kullanma
Geçersiz yönlendirme ve iletimler

Aşağıdaki bölümlerde bu riskler detaylı olarak ele alınarak bu risklerin giderilmesine yönelik kullanılabilecek araçlar veya yöntemlerle ilgili bilgi verilmiştir.

4.1.1. SQL Enjeksiyon

SQL Enjeksiyon, kullanıcı girdileri ile uygulamaları istismar etmek amacıyla veritabanında SQL komutları çalıştırma tekniğidir [46]. Açıklık kullanıcı girdileriyle SQL deyimini yapılandıracak izin verilmemesi gereken bazı karakterlerin kullanılmasıyla ortaya çıkmaktadır [47].

OWASP'ın listesinde yer alan SQL Inject Me, en yaygın bilinen Firefox eklentilerinden biridir (Şekil 4.1). Görüntülenen web sayfasının SQL Enjeksiyon açıklıklarını içerip içermediğini test etmektedir.



Şekil 4.1. SQL Inject Me aracı

4.1.2. Siteler arası betik yazma (XSS)

XSS, istemci tabanlı kodun, HTML kodlarının arasına eklenerek, kurbanın kullandığı tarayıcıda istenen istemci tabanlı zararlı kodun çalıştırılabilmesi olarak tanımlanmaktadır [38,48].

Basit saldırılar `<h1>` veya `<script>` gibi HTML etiketleri içermektedir. En sık kullanılan örnek: “`<script>alert('XSS')</script>`” dir. Bu HTML etiketlerini tespit etmek için kullanılan düzenli ifadeler aşağıdaki gibidir [38]:

`/((\%3C)|<)((\%2F)|\V)*[az09\%]+((\%3E)|>)/ix`

Bu düzenli ifadelerle ilgili açıklamalar Çizelge 4.2’de sunulmuştur.

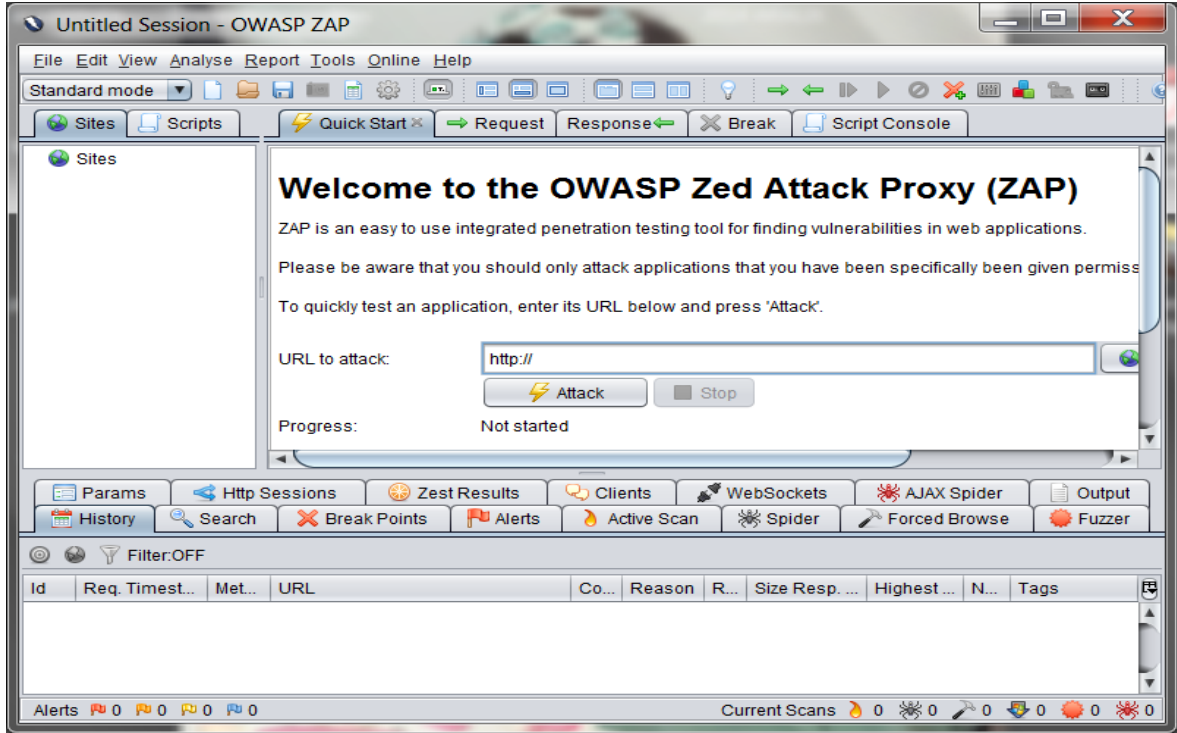
Çizelge 4.2. Düzenli ifadeler

Açıklamalar	
<code>((\%3C) <)</code>	“<” işareti veya hex karşılığı olan ('3C')
<code>((\%2F) /)*</code>	kapatma etiketinde kullanılan “/” veya hex karşılığı olan ('2F')
<code>[az09\%]+</code>	etiket içerisindeki alfanumerik veya hex karşılıkları
<code>((\%3E) >)</code>	“>” işareti veya hex karşılığı olan ('3E')

XSS 2015 yılında halen web uygulamaları için büyük bir problem olmaya devam etmektedir. XSS saldırılarını etkili bir şekilde yok edecek tek bir çözüm bulunmamaktadır. Uygulama kaynak kodlarının açıklık konusunda detaylı analiz edilmesine ihtiyaç duyulmaktadır [49].

Bir OWASP projesi olan ZAP, web uygulamalarındaki zafiyetleri bulmak için kullanılan tarama ve penetrasyon testi aracıdır [42,50]. Güvenlik konusunda uzman kişiler tarafından kullanılması gerekmektedir [42].

ZAP’ın anasayfası aşağıda gösterilmiştir. Sızma testinin yapılacağı URL doğrudan yazılarak özellikle XSS konusunda çok başarılı sonuçlar elde edilmektedir.



Şekil 4.2. ZAP ile sızma testi

4.1.3. İhlal edilmiş kimlik doğrulama ve oturum yönetimi

Kimlik doğrulama ve oturum yönetimiyle ilgili fonksiyonlar genellikle doğru olarak uygulanmamaktadır. Bu durum saldırganların şifrelere, anahtarlara veya oturum bilgilerine ulaşmalarına yol açmaktadır.

Örnek senaryo: Uygulamaların zaman aşımaları doğru olarak ayarlanmamaktadır. Kullanıcı bir web sayfasına erişmek için genel kullanıma açık bir bilgisayardan işlem yaptığında uygulamadan çıkış yapmayı tercih etmek yerine daha kolay bir yöntem olan tarayıcıyı kapatarak ayrılmaktadır. Bir süre sonra saldırgan halen yetkilendirilmeye devam eden tarayıcıyı kullanmaktadır [51].

4.1.4. Emniyetsiz doğrudan nesne referansı

Bu durum geliştiricinin dosya, izin veya veritabanı anahtarı gibi nesnelere doğrudan referans etmesiyle ortaya çıkmaktadır. Erişim kontrolü veya koruması olmaksızın, saldırganlar bu referansları izinsiz olmayan verilere erişmek için manipüle etmektedir [40].

4.1.5. Güvenlik ayarları hataları

Güvenlik; uygulama, framework, uygulama sunucu, web sunucu, veritabanı sunucu ve platformlarına güvenli konfigürasyon tanımlanması ve uygulanmasını gerektirmektedir [40].

Güvenli ayarlar tanımlanmalı, uygulanmalı ve sürdürülmelidir. Varsayılan ayarlar her zaman güvensizdir. Ayrıca yazılımların sürekli güncellenmesi gerekmektedir [40].

4.1.6. Hassas veri teşhiri

Birçok web uygulaması kredi kartı bilgileri gibi hassas verileri uygun olarak korumamaktadır. Saldırganlar bu verileri çalarak veya değiştirerek kredi kartı sahteciliği, kimlik hırsızlığı veya diğer suçları işleyebilmektedir. Hassas veriler için şifreleme gibi ekstra önlemler alınması gerekmektedir [40].

4.1.7. İşlev seviyesi erişim kontrol eksikliği

Sunucu üzerinde erişilen fonksiyonların erişim haklarının doğru olup olmadığının kontrolünün yapılması gerekmektedir [40]. Tarayıcıdan gelen isteğin yetkili kullanıcıdan mı yoksa kötü niyetli kullanıcıdan mı gönderildiğinin kontrolünün yapılmadığı durumlarda ortaya çıkan zafiyettir [52]. Örneğin sadece yöneticinin kullandığı ve bilmesi gereken URL'e başka kullanıcılar tarafından erişilmesi durumudur [40]:

<http://example.com/app/getappInfo>

http://example.com/app/admin_getappInfo

Bu saldırıdan korunmak için yetkilendirme yönetimi kullanılması gerekmektedir. Her fonksiyona, sayfaya herkes için ayrı ayrı erişim yetkisi tanımlanması gerekmektedir.

4.1.8. Siteler Ötesi İstek Sahteciliği (CSRF)

CSRF saldırısında, zafiyet içeren bir internet uygulamasına istek yollaması için kurbanın tarayıcısına zorla giriş yapılarak; kurbanın adına saldırı gerçekleştirilmektedir. CSRF yeni

bir saldırı türü değildir fakat uygulanması kolaydır ve zarar verici olabilmektedir. Günümüzde birçok internet uygulaması sadece, otomatik olarak yollanan, oturum çerezleri, temel kimlik doğrulama bilgileri, kaynak IP adresleri, SSL sertifikaları vb. güvenmektedir [53]. Konuya ilişkin örnek aşağıda sunulmuştur.

Saldırganın kurbandan istek yapmasını istediği URL:

<http://example.com/app/transferFunds?amount=1500&destinationAccount=4673243243>

Saldırganın kurbanı sunduğu URL:

```

```

Bu URL de eni boyu olmayan bir resim görülmektedir. Dolayısıyla burada sayfa yüklendiği anda kurban haberi olmadan bu URL'e istek göndermiş olmaktadır. Kurbanın tarayıcısında açık olan banka hesabı esnasında bu URL ile arka planda hesap transfer işlemi yapılabilmektedir [40, 52].

Bu saldırıdan korunmak için her oturuma benzersiz bir işaret değeri atanmalı ve her tekrarda kontrol edilmelidir [52]. Hassas veri işlemleri için, tekrar kimlik doğrulanmalı veya isteğin gerçek olduğunun kesinleşmesi için işlem işareti kullanılmalıdır [53].

4.1.9. Bilinen güvenlik açıklıkları ile bileşenleri kullanma

Kütüphane, çerçeve (framework) ve diğer yazılım modülleri gibi bileşenlerin hepsi her zaman tam ayrıcalıkla çalışmaktadır. Zafiyet içeren bir bileşen varsa saldırganlar tarafından istismar edilme ihtimali vardır. Saldırganlar zayıf bileşenleri tarama yoluyla veya manuel yöntemlerle tespit ederek saldırıyı gerçekleştirebilmektedirler [54].

Örneğin kullanılan Apache sunucunun içerdiği zafiyetlerden ötürü yeni bir sürümü yayınlanabilir. Kurban kullandığı yazılımı yeni sürüme güncellemediği durumda, saldırgan çeşitli tarama yöntemleri ile kurbanın kullandığı yazılımın eski sürümde olduğunu gördüğü anda o sürümün içerdiği zafiyeti istismar edebilmektedir [52,54].

Bu zafiyet daha önceki maddelerde bahsedilen tüm riskleri kapsayabileceği için (enjeksiyon, XSS vb.) vereceği zarar da çok ciddi boyutta olabilmektedir [54].

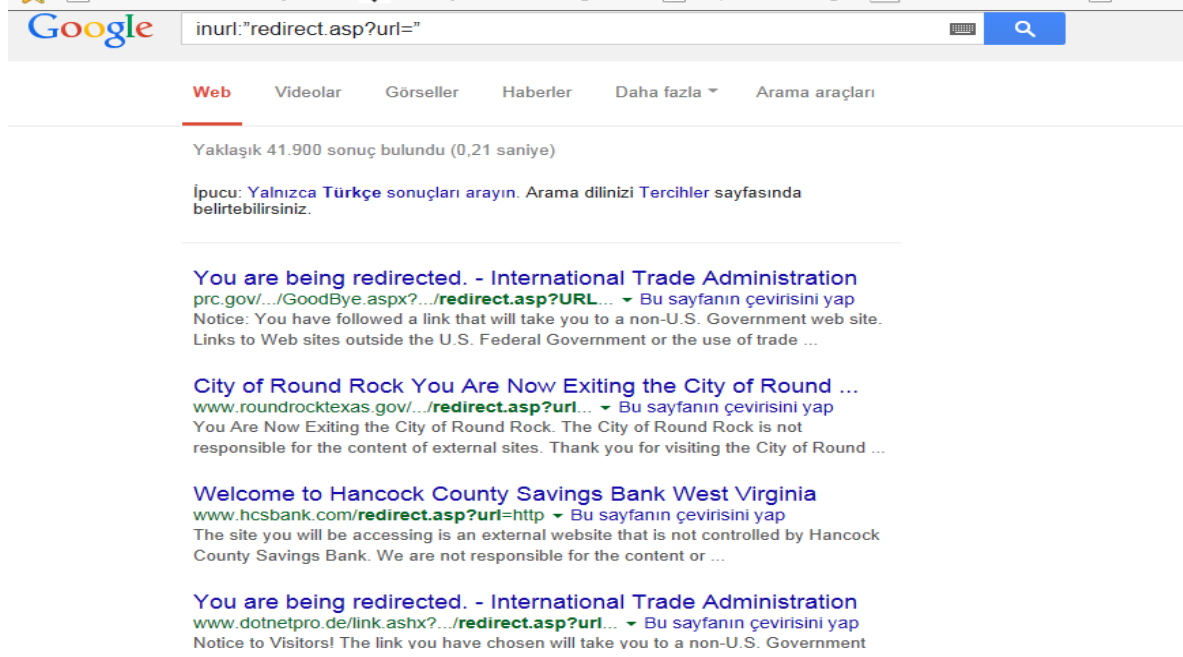
Bu zafiyetten korunmak için kullanılan tüm uygulama ve yazılımların içerik ve versiyonlarının tanımlanması, güncel tutulması ve güvenlik testlerini geçen yazılımların tercih edilmesi gerekmektedir [52,54].

4.1.10. Geçersiz yönlendirme ve iletimler

Web uygulamaları sıklıkla kullanıcıları diğer sayfalara, web sitelerine yönlendirmekte ve hedefteki sayfaları tanımlamak için güvensiz veri kullanmaktadır. Uygun doğrulama yapılmadan saldırganlar kurbanları oltalama (phishing) veya zararlı yazılım sayfalarına yönlendirebilmekte veya yetkilerinin olmadığı sayfalara erişmek için kullanabilmektedirler [40].

Bu saldırıdan korunmak için yönlendirmeler kullanılmayabilir veya kullanılması gerekiyorsa dışardan parametre girilecek şekilde uygulanmaması gerekmektedir [52].

“Geçersiz yönlendirme ve iletimler” zafiyetinin ne kadar yaygın olduğunu görmek amacıyla arama motoruna “inurl:”redirect.asp?url=”” ifadesi yazılmıştır alınan sonuçlar sunulmuştur (Şekil 4.3).



Şekil 4.3. Geçersiz yönlendirme ve iletimler zafiyeti araması

4.2. WebGoat Programı ile Web Saldırıları Örnekleri

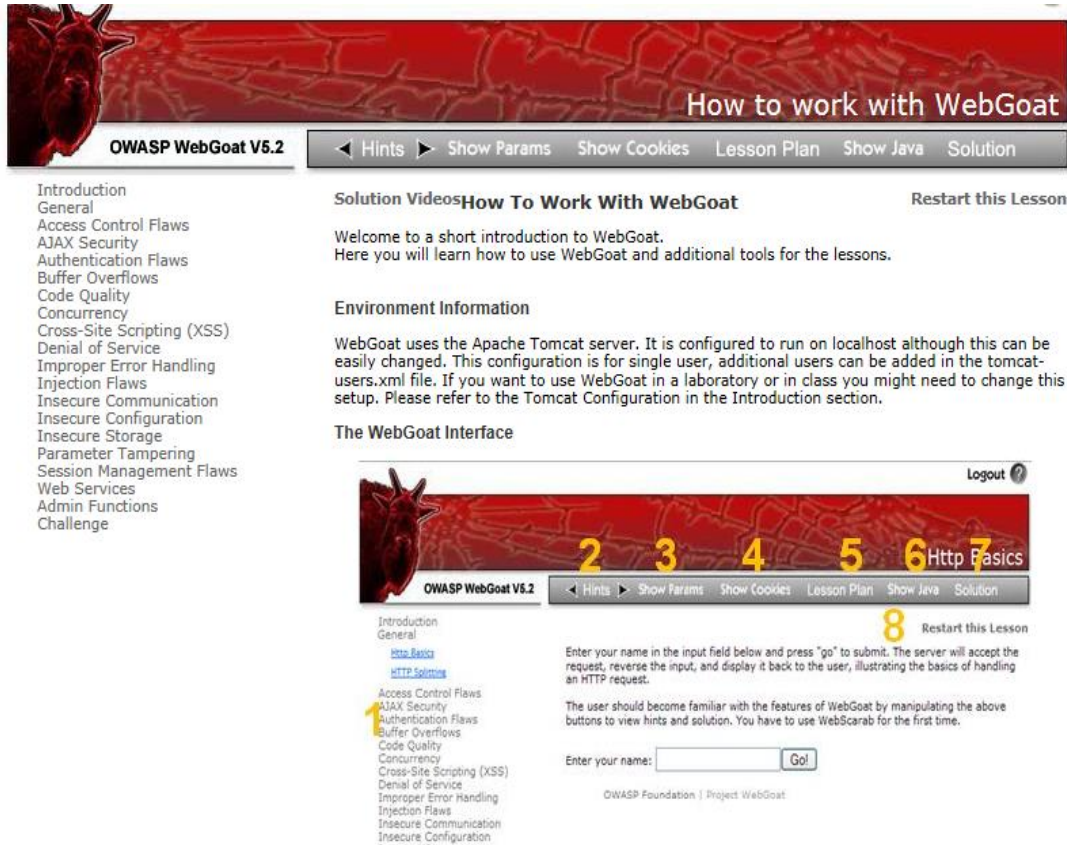
WebGoat, OWASP (Open Web Application Security Project) tarafından J2EE platformunda geliştirilmiş olan, içerisinde eğitim amaçlı yüzlerce açık barındıran, web uygulamalarındaki açıkları öğrenebilmeyi sağlayan güvenlik test yazılımıdır [55,56].

Web iz bilgilerindeki saldırıları analiz etmek için saldırıların nasıl gerçekleştiğini öğrenme ve test etme amacıyla WebGoat aracı kullanılmıştır. WebGoat Apache Tomcat ve Java geliştirme ortamı kullanılmaktadır [57]. İçeriğinde web güvenliğine yönelik dersler bulunmaktadır. Bu bölümde yazılımın kullanımıyla ilgili genel bilgiler verilmiştir ve çeşitli web saldırıları pratik olarak denenmiştir.

Yazılımın amacı web uygulamalarına yönelik çeşitli kategorilerdeki dersler ile güvenlik uzmanlarına veya web uygulaması geliştirenlere; güvenlik açığını anlayabilme, çözebilme ve istismar (exploit) edebilme yetenekleri kazandırmaktır [55].

Örnek olarak SQL Enjeksiyon açığı kullanarak sahte kredi kartı bilgileri ile alışveriş yapılması istenmektedir. Burada sistemdeki form bilgilerinin SQL Enjeksiyon yöntemi ile

istismar edilmesi gerekmektedir. Başarıya ulaşıldığı durumda bir sonraki teste geçilmektedir [55]. WebGoat anasayfası Şekil 4.4’de gösterilmiştir.



Şekil 4.4. WebGoat anasayfası

WebGoat web uygulamalarını güvenlik açısından test etmekten ziyade eğitim amaçlı tasarlanmış bir araçtır [42].

Aşağıdaki bölümlerde web uygulamalarında en sık karşılaşılan ve en yaygın 2 saldırı çeşiti olan SQL Enjeksiyon ve XSS ile ilgili WebGoat kullanılarak örnekleme yapılmıştır.

4.2.1. WebGoat programı ile SQL Enjeksiyon örnekleme

WebGoat kullanarak, Enjeksiyon Flaws-> String SQL Enjeksiyon adımına gelerek listedeki yöneticilerden biri olan Neville Bortholomew adıyla SQL Enjeksiyon ile giriş yapılarak saldırı örneği gerçekleştirilmiştir. Sisteme yönetici olarak giriş yapıldığında listedeki tüm personelin profil bilgilerine erişilebilmektedir. Listedeki personelden Larry

Stooge profil bilgilerindeki ad-soyad alanları değiştirilerek veritabanında SQL Enjeksiyon saldırısı sonucunda veri güncelleme işlemi gerçekleştirilmiştir.

Burada String SQL Enjeksiyon yapılmaktadır. Bunun için Firefox'un bir eklentisi olan "tamper data" kullanılmıştır (Şekil 4.5). Bu eklenti sayesinde sayfada post ve get ile gönderilen tüm parametreler görüntülenmekte ve değiştirilebilmektedir. Listedeki tüm bilgileri dökmek için password alanına SQL Enjeksiyon ifadesi (x' or 'a'='a) yazılmıştır (Şekil 4.5).

Request Header Name		Request Header Value	Post Parameter Name	Post Parameter Value
Host		localhost:8080	employee_id	112
User-Agent		Mozilla/5.0 (Windows NT 6.1; WOW64; rv:21.0) Gecko/20100101	password	x' or 'a'='a
Accept		text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0	action	Login
Accept-Language		en-US,en;q=0.5		
Accept-Encoding		gzip, deflate		
Referer		http://localhost:8080/WebGoat/attack?Screen=147&menu=110		
Cookie		JSESSIONID=903E43941AB3604CF3C0FF5F8EE8CEB2		

Şekil 4.5. String SQL Enjeksiyon gerçekleştirmek üzere parametrelerin girilmesi

SQL Enjeksiyon sonucu personel bilgileri listelenmiştir (Şekil 4.6).

Code Quality
Concurrency
Cross-Site Scripting (XSS)
Improper Error Handling
Injection Flaws

Command Injection
Numeric SQL Injection
Log Spoofing
XPath Injection

String SQL Injection
LAB: SQL Injection

Stage 1: String SQL Injection
Stage 2: Parameterized Query #1
Stage 3: Numeric SQL Injection
Stage 4: Parameterized Query #2

Modify Data with SQL Injection
Add Data with SQL Injection
Database Backdoors
Blind Numeric SQL Injection
Blind String SQL Injection

Denial of Service
Insecure Communication
Insecure Configuration
Insecure Storage
Malicious Execution
Parameter Tampering
Session Management Flaws
Web Services
Admin Functions
Challenge

THIS LESSON ONLY WORKS WITH THE DEVELOPER VERSION OF WEBGOAT

Implement a fix to block SQL injection into the fields in question on the Login page 1. Verify that the attack is no longer effective.

*** You have completed Stage 1: String SQL Injection.**
*** Welcome to Stage 2: Parameterized Query #1**

Goat Hills Financial
Human Resources

Welcome Back Neville - Staff Listing Page

Select from the list below

- Moe Stooge (manager)
- Curly Stooge (employee)
- Eric Walker (employee)
- Tom Cat (employee)
- Jerry Mouse (hr)
- David Giambi (manager)
- Bruce McGuire (employee)
- Sean Livingston (employee)
- Joanne McDougal (hr)
- John Wayne (admin)
- Neville Bartholomew (admin)

SearchStaff
ViewProfile
CreateProfile
DeleteProfile
Logout

Şekil 4.6. String SQL Enjeksiyon sonucu veritabanında erişilen personel listesi

Buradaki profili güncelleme seçeneği ile profildeki bilgiler değiştirilebilmekte, yeni bir profil eklenebilmektedir. Aynı zamanda var olan profili tamamen silme seçeneği de bulunmaktadır.

4.2.2. WebGoat programı ile XSS örnekleme

WebGoat kullanarak, Cross Site Scripting (XSS) -> Stored XSS adımına gelerek personel bilgilerinin yer aldığı ve kullanıcıların kendilerine ait bilgileri görüntüleme ve güncelleme işlemlerini yapabildikleri sayfada XSS atağı gerçekleştirilmiştir. Şekil 4.7’de Larry Stooage isimli kullanıcı için profil bilgileri görülmektedir.

Access Control Flaws
AJAX Security
Authentication Flaws
Buffer Overflows
Code Quality
Concurrency
Cross-Site Scripting (XSS)
Phishing with XSS
LAB: Cross Site Scripting
Stage 1: Stored XSS
Stage 2: Block Stored XSS using Input Validation
Stage 3: Stored XSS Revisited
Stage 4: Block Stored XSS using Output Encoding
Stage 5: Reflected XSS
Stage 6: Block Reflected XSS
Stored XSS Attacks
Cross Site Request Forgery (CSRF)
Reflected XSS Attacks
HTTPOnly Test
Cross Site Tracing (XST) Attacks
Denial of Service
Improper Error Handling
Injection Flaws
Insecure Communication
Insecure Configuration
Insecure Storage
Parameter Tampering
Session Management Flaws
Web Services
Admin Functions

THIS LESSON ONLY WORKS WITH THE DEVELOPER VERSION OF WEBGOAT

Implement a fix to block the stored XSS before it can be written to the database. Repeat stage 1 as 'Eric' with 'David' as the manager. Verify that 'David' is not affected by the attack.

Şekil 4.7. Personel bilgileri güncelleme ekranı

Bu kullanıcı için profili güncelleme seçeneği seçilmiştir ve Street alanında XSS gerçekleştirmek üzere “ <script>alert('isil cinar')</script>” betiği yazılmıştır (Şekil 4.8).

Introduction
General
Access Control Flaws
AJAX Security
Authentication Flaws
Buffer Overflows
Code Quality
Concurrency
Cross-Site Scripting (XSS)
Phishing with XSS
LAB: Cross Site Scripting
Stage 1: Stored XSS
Stage 2: Block Stored XSS using Input Validation
Stage 3: Stored XSS Revisited
Stage 4: Block Stored XSS using Output Encoding
Stage 5: Reflected XSS
Stage 6: Block Reflected XSS
Stored XSS Attacks
Cross Site Request Forgery (CSRF)
Reflected XSS Attacks
HTTPOnly Test
Cross Site Tracing (XST) Attacks
Denial of Service
Improper Error Handling
Injection Flaws
Insecure Communication
Insecure Configuration
Insecure Storage
Parameter Tampering
Session Management Flaws
Web Services
Admin Functions

Solution VideosStage 2: Block Stored XSS using Input Validation.

Restart this Lesson

THIS LESSON ONLY WORKS WITH THE DEVELOPER VERSION OF WEBGOAT

Implement a fix to block the stored XSS before it can be written to the database. Repeat stage 1 as 'Eric' with 'David' as the manager. Verify that 'David' is not affected by the attack.

Şekil 4.8. XSS gerçekleştirmek üzere betik yazılan ekran

Betik yazıldıktan sonra profil bilgileri kaydedildiğinde aşağıdaki gibi uyarı mesajının çıktığı görülmüştür. Başka bir kullanıcı hesabından örneğin yönetici olarak giriş yapılarak Larry isimli kullanıcının profil bilgileri görüntülenmek istendiğinde her seferinde artık bu betik ile aşağıdaki uyarı penceresi görüntülenmektedir. XSS atağı başarıyla gerçekleştirilmiştir.



Şekil 4.9. XSS atağının başarıyla gerçekleştirilmesi sonucu açılan uyarı mesajı

5. UYGULAMA

5.1. Önişleme

Veri madenciliğinde olduğu gibi web madenciliği de veri temizleme ve hazırlama işlemiyle başlamakta ve önceden tahmin edilemeyen gizli bilgilerin ortaya çıkarılmasıyla sonlanmaktadır. Bu işlemler sırasında en önemli odak noktası veri temizleme ve ön işleme aşamasıdır. Veri analizinde yapılacak çalışmanın kalitesi bu aşamayla doğrudan ilişkilidir [58]. Bu aşama web kullanım madenciliğinin en önemli aşaması olarak kabul edilmektedir; çünkü etkili bir şekilde yapıldığında zaman ve kaynak tasarrufu sağlamaktadır [59].

Madencilik işlemlerinin kolaylıkla yapılabilmesi için verinin etkinliğini artırmak üzere verinin önişlenmesi gerekmektedir. Veri önişlemede yapılması gereken ana işlem gürültülü ve ilgisiz verinin temizlenmesi ve verinin azaltılmasıdır [60].

Veri önişleme web madenciliğinde amaca yönelik olarak değişkenlik göstermektedir. Veri temizleme aşamasında analiz edilen iz bilgisi dosyasına özel olarak analiz amacı için önem arzetmeyen nesneler (stil dosyaları, grafik dosyaları, ses dosyaları vb.) temizlenmektedir [60]. Bununla birlikte bazı özel resim sitelerinde veya haber sayfalarında kullanıcıların ilgisinin analiz edilmesi gibi bir durumda özellikle bu dosyalara odaklanmak da gerekebilmektedir. Burada önemli olan web iz bilgisi madenciliğinin amacının doğru olarak belirlenmesidir [61].

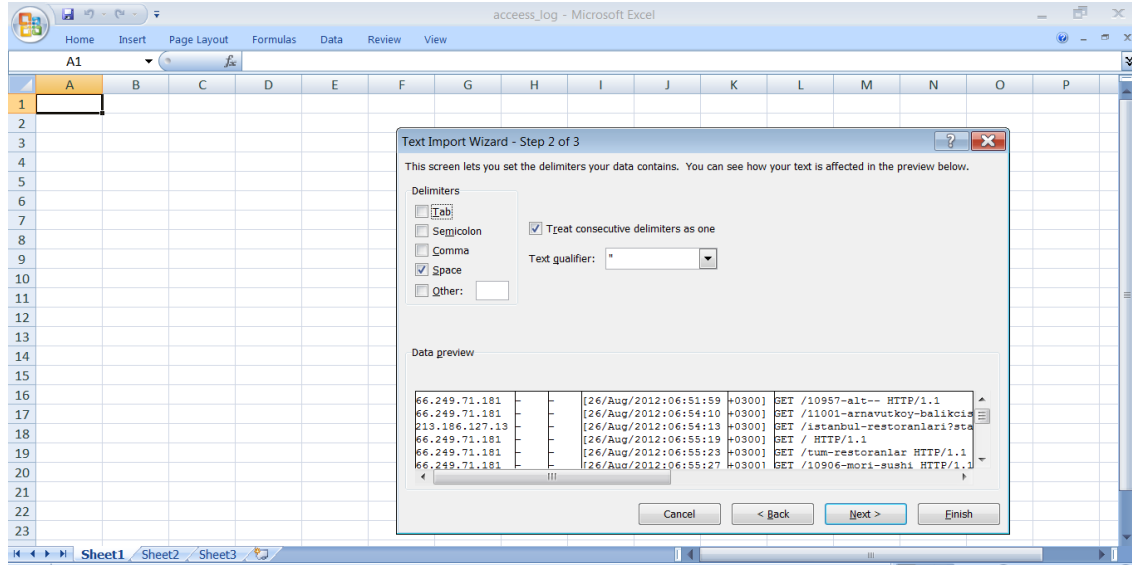
Web kullanım madenciliğinde özellikle 200 serili durum kodlarının yer aldığı isteklerin analiz edilmesi gerekmektedir. Web iz bilgilerinden saldırı analizi yapılacağına ise web kullanım madenciliği önişlemesinin aksine anomali tespiti içerebileceğinden 400 serili durum kodlarının korunması gerekmektedir. 200 serili isteğin başarılı olduğunu gösteren durum kodlarının yer aldığı isteklerin ise silinmesi gerekmektedir. Çünkü bu karakterdeki bir isteğin web saldırısı olma ihtimali neredeyse yoktur. Ancak saldırı başarılı olduğunda hata kodlarının ardından başarı durum kodu alınacağı dikkate alınmalıdır [28].

Önişlemenin etkin yapılması analizin hızını, güvenilirliğini, genel olarak performansını ciddi oranda etkilemektedir. Veri madenciliği yöntemleri ile web saldırı tespiti yapan

çalışmaların [18, 62, 63] birçoğunda önışleme aşamasında hata durum kodları veya başarı durum kodlarına yönelik önışleme çalışması yapılmadığı görölmüşür.

T. Aye yaptığı çalışmada veri temizleme algoritmasıyla analiz edilen verideki gereksiz olan kısımları çıkarmıştır [60]. İz bilgisi girdileri bölünmesi gereken çeşitli alanlar içermektedir. İz bilgisi dosyasındaki tek bir satırdan alanların bölünerek çıkarılması işleminde “alan ayrıştırma” adı verilmektedir. Burada en çok kullanılan ayırıcı karakterler ‘,’ veya ‘boşluk’ karakteridir. T. Aye bu işlemi gerçekleştirmek için ayrıştırma algoritması oluşturmuştur [60].

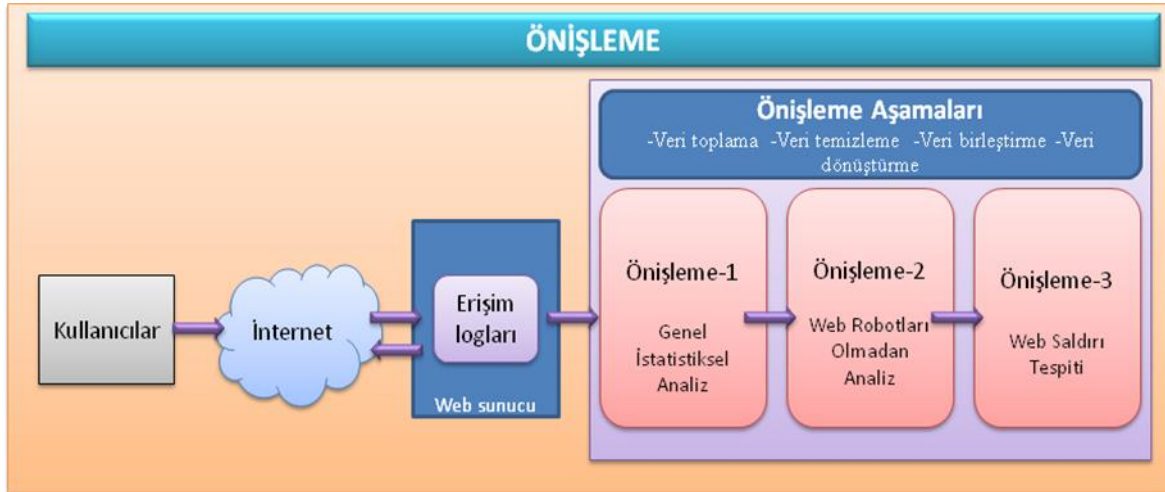
Bu çalışmada alan ayrıştırma işlemini gerçekleştirmek için Notepad++ kullanılarak açılan iz bilgisi dosyası Microsoft Office Excel 2007 kullanılarak alanlarına göre ayrılarak yüklenmiştir. Bu alanlar standart web iz bilgisi dosyasında bulunan temel alanlara göre belirlenmiştir. Dönüşürme işlemi tablo alanları için uygun veri tipleri seçilerek tamamlanmıştır (Şekil 5.1).



Şekil 5.1. Veri dosyasının excel ortamına aktarılması

Devamında veriler üzerinde, gürültü temizleme işlemini daha hassas gerçekleştirebilmek için ve analiz etmek amacıyla çeşitli sorgular yazmak üzere Excel’de alanlarına göre ayrıştırılmış olan veriler Microsoft Office Access 2007 veritabanı programına aktarılmıştır.

Bu tez çalışmasında web madenciliği amacına yönelik olarak önışleme 3 aşamada gerçekleştirilmiş; her bir aşamaya özel önışleme dosyası oluşturularak web madenciliği analizi yapılmıştır. Şekil 5.2’de bu aşamalar gösterilmektedir:



Şekil 5.2. Önışleme aşamaları

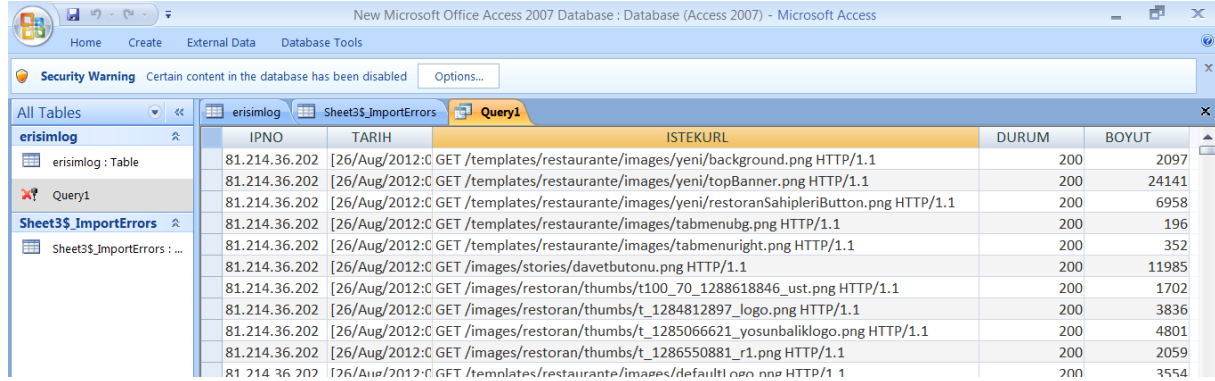
Her bir aşamaya özel gerçekleştirilen önışleme bilgileri aşağıda sunulmuştur.

1. Web kullanım madenciliğine yönelik genel istatistiksel sonuçlar çıkarma amacıyla gerçekleştirilen önışleme adımları aşağıda sunulmuştur:
 - a. Veri tutulmayan alanlar silinmiştir. Şekil 5.3’de Kimlik (Identity) ve Kullanıcı (User) alanlarında veri tutulmadığı için bu alanların silindiği görülmektedir.

	A	B	F	G
1	66.249.71.181	-	GET /10957-alt-- HTTP/1.1	200
2	66.249.71.181	-	GET /11001-arnavutkoy-balikcisi HTTP/1.1	200
3	213.186.127.13	-	GET /istanbul-restoranlari?start=40 HTTP/1.1	200
4	66.249.71.181	-	GET / HTTP/1.1	200
5	66.249.71.181	-	GET /tum-restoranlar HTTP/1.1	200
6	66.249.71.181	-	GET /10906-mori-sushi HTTP/1.1	200
7	66.249.71.181	-	GET /10986-anzer-sofrasi HTTP/1.1	200
8	66.249.71.181	-	GET /beykoz-restoranlari HTTP/1.1	200
9	66.249.71.181	-	GET /beykoz-restoranlari HTTP/1.1	200

Şekil 5.3. Gereksiz alanların silinmesi

- b. Resim, müzik ve diğer bağlı olan dosyalar olmadan kaç ziyaret olduğunu tespit etmek amacıyla .gif, .jpg, .css, .js, .png, javascript vb. uzantılı dosyalar silinmiştir. Şekil 5.4’de silinen .png uzantılı dosyaların bir bölümü gösterilmektedir.



IPNO	TARİH	İSTEKURL	DURUM	BOYUT
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/yeni/background.png HTTP/1.1	200	2097
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/yeni/topBanner.png HTTP/1.1	200	24141
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/yeni/restoranSahipleriButton.png HTTP/1.1	200	6958
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/tabmenubg.png HTTP/1.1	200	196
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/tabmenuright.png HTTP/1.1	200	352
81.214.36.202	[26/Aug/2012:0	GET /images/stories/davetbutonu.png HTTP/1.1	200	11985
81.214.36.202	[26/Aug/2012:0	GET /images/restoran/thumbs/t100_70_1288618846_ust.png HTTP/1.1	200	1702
81.214.36.202	[26/Aug/2012:0	GET /images/restoran/thumbs/t_1284812897_logo.png HTTP/1.1	200	3836
81.214.36.202	[26/Aug/2012:0	GET /images/restoran/thumbs/t_1285066621_yosunbaliklogo.png HTTP/1.1	200	4801
81.214.36.202	[26/Aug/2012:0	GET /images/restoran/thumbs/t_1286550881_r1.png HTTP/1.1	200	2059
81.214.36.202	[26/Aug/2012:0	GET /templates/restaurante/images/default.png HTTP/1.1	200	3554

Şekil 5.4. Silinen .png uzantılı dosyalar

- c. Tarih alanı için ayrıştırma işlemi yapılmıştır. Tarih alanındaki veriler sürekli. WEKA’da yapılacak örüntü keşfi aşamaları için tarih formatı “gün/ay/yıl” formatına dönüştürülmüştür.
- d. Metot, protokol ve URL alanları birleştirilerek tek bir alanda toplanmıştır. URL alanındaki veriler iz bilgisi formatına uygun olması amacıyla tırnak işareti içinde olacak şekilde birleştirilmiştir.
- e. İstatistiksel sonuçlarda web robotlarının durumlarının da tespit edilmesi amacıyla bu kayıtlar korunmuştur.
2. İlk maddede gerçekleştirilen önışlemede web robotları erişimleri de bulunmaktadır. Web robotları (crawler, spider, bot vb.) “robots.txt” dosyasına erişmekte ve yöneticinin verdiği izinlere göre işlem yapmaktadırlar. İz bilgilerinin kaydadeğer bir bölümünü web robotları oluşturduğundan bu isteklerin temizlenmesi analizde kullanıcı odaklı sonuç çıkarma açısından çok önemlidir. İlk maddede gerçekleştirilen önışlemeye ilave olarak web robotlarına ait iz bilgileri de temizlenmiştir. Bölüm 5.1.1.’de arama robotlarının analize etkisi hakkında bilgi verilerek web robotlarının nasıl tespit edildiği gösterilmiştir.

3. Web saldırı tespitine yönelik analiz yapmak amacıyla ilk 2 maddede gerçekleştirilen önışlemeye ilave olarak 400 serili hata durum kodlarının dışındaki durum kodlarının yer aldığı istekler silinmiştir (Şekil 5.5).

IPNO	TARİH	URL	DURUM	BOYUT	BASVURULAN	TARAYICI
72.14.199.216	26/Aug/2012	"GET /10933-botanica-cafe HTTP	403	699	"	"AdsBot-Google (+http://www.google.com/adsbo
72.14.199.216	26/Aug/2012	"GET /11045-akademi-restaurant	403	699	"	"AdsBot-Google (+http://www.google.com/adsbo
180.76.5.56	26/Aug/2012	"GET /11100-dilr HTTP/1.1"	404	695	"	"Mozilla/5.0 (compatible; Baiduspider/2.0; +http:/
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 6.0;
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 6.0;
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Mozilla/2.0 (compatible; MSIE 3.02;
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 6.0;
188.3.2.80	26/Aug/2012	"GET /crossdomain.xml HTTP/1.	404	399	"http://www.yem	"Mozilla/5.0 (Windows NT 6.1)
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 6.0;
87.98.182.241	26/Aug/2012	"GET /component/index.php HT	404	1370	"http://www.yem	"Opera/8.00 (Windows NT 5.1;
176.216.79.21	26/Aug/2012	"GET /templates/restaurante/js	404	449	"http://www.yem	"Mozilla/5.0 (Windows; U; MSIE
176.216.79.21	27/Aug/2012	"GET /tr/plugins/syste-type HT	404	1370	"http://www.yem	"Mozilla/5.0 (Windows; U; MSIE
176.216.79.21	27/Aug/2012	"GET /templates/restau/xa3 HT	404	1370	"http://erp.myme	"Mozilla/5.0 (Windows; U; MSIE
180.76.5.192	27/Aug/2012	"GET /11100-dilr HTTP/1.1"	404	695	"	"Mozilla/5.0 (compatible; Baiduspider/2.0; +http:/
85.110.126.15	27/Aug/2012	"GET /crossdomain.xml HTTP/1.	404	399	"	"Mozilla/5.0 (compatible; MSIE 9.0;
157.56.93.228	28/Aug/2012	"GET /10933-botanica-cafe HTTP	403	699	"	"Mozilla/5.0 (Windows NT 6.1;
69.171.237.10	28/Aug/2012	"GET /http://yemegecikalim.cor	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
180.76.5.113	29/Aug/2012	"GET /11100-dilr HTTP/1.1"	404	695	"	"Mozilla/5.0 (compatible; Baiduspider/2.0; +http:/
88.227.216.41	29/Aug/2012	"GET /undefined HTTP/1.1"	404	695	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 8.0;
66.220.152.3	29/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
66.220.152.1	29/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
69.63.189.249	29/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
66.220.152.3	30/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
66.220.152.7	30/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
62.192.124.17	30/Aug/2012	"GET /http://www.yemegecikal	404	695	"	"facebookexternalhit/1.1 (+http://www.facebook
1370	"http://www.yem	"Mozilla/4.0 (compatible; MSIE 6.0;				

Şekil 5.5. 400 serili durum kodlarını içeren kayıtlar

5.1.1. Arama robotları ve analize etkisi

Birçok web arama robotu web sitelerinin içeriğini elde etmek için sık sık sayfaları ziyaret etmektedir. Bu durum web sitelerine çok sık erişimden kaynaklanan bir yük getirmektedir [64]. Web iz bilgileri analizinde arama robotlarının veri madenciliğinin önışleme aşamasında tespit edilerek filtrelenmesi gerekmektedir.

Bazı arama robotları kısa periyotlar içerisinde düzenli istekler göndererek sayfaları indekslerken bazıları zamandan bağımsız veya farklı periyotlarda istek gönderebilmekte bu durum web sunucu trafiğini gün içerisinde değiştirebilmektedir [65]. Büyük boyutlu trafik normal kullanıcı erişimlerinin analizi konusunda sıkıntı yaratmaktadır. Web sunucu iz bilgilerindeki gerçek bilginin ortaya çıkarılması; analizlerin doğruluğunu artıracığı gibi çok daha hızlı analiz sağlayacaktır [64].

Bu aşamaya kadar yapılan çalışmada ön işleme aşamasında veri temizlemede resim ve diğer bağlı olan dosyalar olmadan kaç ziyaret olduğunu tespit etmek amacıyla .gif, .jpg, .css, .js, .png, javascript uzantılı dosyalar silinmiştir. Veri sayıları Çizelge 5.1’de gösterilmiştir.

Çizelge 5.1. Erişimlerdeki veri sayıları

Erişimler	
Toplam Veri	685.272
Önişlemden Sonra Veri Sayısı	64.427

Ancak burada web robotları erişimleri de bulunmaktadır. Web robotları “robots.txt” dosyasına erişmekte ve yöneticinin verdiği izinlere göre işlem yapmaktadırlar. Web robotları (crawler, spider, bot vb.) ile ilgili yapılan birçok çalışma incelenmiştir [66-69]. Web robotları günün her saati, her dakikası; haftanın her gününde oturum sayıları, sayfanın hitleri, var olan sayfalar vb. birçok bilgiyi çok kısa sürelerde indekslemektedirler [69]. Bu çalışmalarda web robotlarının tespiti ve web iz bilgilerinin incelenmesinde yarattığı gereksiz kayıtlar üzerinde durulmuştur. Web robotlarının tespitinde URL’de geçen “robots.txt” ifadelerinin tespiti ile bizim de çalışmamızda uyguladığımız “robots.txt” dosyalarına uğrayan ve sonrasında diğer sayfaları da indeksleyen IP numaralarının tespiti edilmesi gerekmektedir [66-69].

Web robotlarına ait bu erişimlerin hepsi listelenmiştir. Robots.txt dosyasına istek gönderen toplam 537 farklı IP tespit edilmiştir. Tespit edilen robotların indekslediği sayfalar da IP numaralarına göre belirlenmiştir ve toplam 20153 kayıt olduğu görülmüştür (Şekil 5.6).

300315dbData - asil : Database (Access 2002 - 2003 file format) - Microsoft Access

Database Tools

Calibri 11

Font Rich Text Records Sort & Filter Find

Query3 Query4

	TARİH	URL	DURUM	BOYUT	BASVUR	TARAYICI
157.55.32.186	17/Oct/2012	"GET /11073-therapia HT	200	17220	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
66.249.76.130	17/Oct/2012	"GET /robots.txt HTTP/1.	200	145	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
66.249.76.130	17/Oct/2012	"GET /11015-reyvan-reste	200	19158	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
157.55.35.37	17/Oct/2012	"GET /10920-beeve-steak	200	21693	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
66.249.76.130	17/Oct/2012	"GET /11039-umitkoy-bal	200	18624	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
157.55.34.183	17/Oct/2012	"GET /10942-tike-cayyolu	200	18984	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
208.115.113.91	17/Oct/2012	"GET /11092-kubban-gazi	200	93437	"	"Mozilla/5.0 (compatible; Ezooms/1.0; ezooms.bot
66.249.76.130	17/Oct/2012	"GET /10908-kurucesme-	200	18771	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
66.249.76.130	17/Oct/2012	"POST /index.php HTTP/:	200	766	"http://ww	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
66.249.76.130	17/Oct/2012	"GET /10892-lokal-asmali	200	18169	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
157.55.35.79	17/Oct/2012	"GET /robots.txt HTTP/1.	200	319	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
66.249.76.130	17/Oct/2012	"GET /11044-ankara-cund	200	18645	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
65.55.52.97	17/Oct/2012	"GET /11081-cafe-bahce	200	17266	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
157.56.229.186	17/Oct/2012	"GET /11066-bekri-tunali	200	17379	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
66.249.76.130	17/Oct/2012	"GET /10979-bolka-turk-n	200	18663	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
66.249.76.130	17/Oct/2012	"POST /index.php HTTP/:	200	1325	"http://ww	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
65.55.52.97	17/Oct/2012	"GET /10969-star-fish-bal	200	20469	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
208.115.113.91	17/Oct/2012	"GET /robots.txt HTTP/1.	200	319	"	"Mozilla/5.0 (compatible; Ezooms/1.0; ezooms.bot
175.158.29.208	17/Oct/2012	"GET /robots.txt HTTP/1.	200	319	"	"Yeti/1.0 (NHN Corp.; http://help.naver.com/robots
175.158.29.208	17/Oct/2012	"GET /10996-yonca-resta	200	98276	"	"Yeti/1.0 (NHN Corp.; http://help.naver.com/robots
175.158.29.210	17/Oct/2012	"GET /10995-hakan-et---	200	103591	"	"Yeti/1.0 (NHN Corp.; http://help.naver.com/robots
66.249.76.130	17/Oct/2012	"GET /11069-kafeluka-be	200	17415	"	"Mozilla/5.0 (compatible; Googlebot/2.1; +http://w
65.52.109.36	17/Oct/2012	"GET /robots.txt HTTP/1.	200	145	"	"msnbot-media/1.1 (+http://search.msn.com/msnb
175.158.29.210	17/Oct/2012	"GET /11024-kucukyali-ke	200	87891	"	"Yeti/1.0 (NHN Corp.; http://help.naver.com/robots
157.55.32.115	17/Oct/2012	"GET /robots.txt HTTP/1.	200	319	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www
157.55.32.115	17/Oct/2012	"GET /11029-messt-cafe-	200	19201	"	"Mozilla/5.0 (compatible; bingbot/2.0; +http://www

Record: 1 of 20153 No Filter Search

Şekil 5.6. Filtrelenmesi gereken web robot istekleri

Şekil 5.6'daki sorgu sonucundan da görüldüğü gibi önışlemden geçmiş olan verinin yaklaşık %31'lik gibi büyük bir kısmını arama robotları erişimi oluşturmaktadır. Bu büyük oran hem istatistiksel değerdendirmelerde hem de saldırı tespitinde analizi olumsuz etkileyeceğinden önışleme aşamasına dahil edilerek gerekli veri temizleme işlemi tamamlanmıştır.

5.2. İz Bilgilerinin Analizi

Çalışmada www.ABC.com web sitesine ait web erişim iz bilgisi dosyası ve hata günlüğü dosyaları üzerinde analiz gerçekleştirilmiştir. Web iz bilgisi erişim dosyasının özellikleri Çizelge 5.2'de gösterilmiştir.

Çizelge 5.2. Erişim dosyası özellikleri

Erişim Dosyasının Özellikleri	
Dosyanın Adı	Access_ABC.log.1
Dosyanın Boyutu	313.413.409 KB
Tarih Aralığı	26.08.2012-31.10.2012
Erişim Sayısı	685.272

Hata Günlüğü dosyasının özellikleri Çizelge 5.3’de gösterilmiştir.

Çizelge 5.3. Hata günlüğü dosyası özellikleri

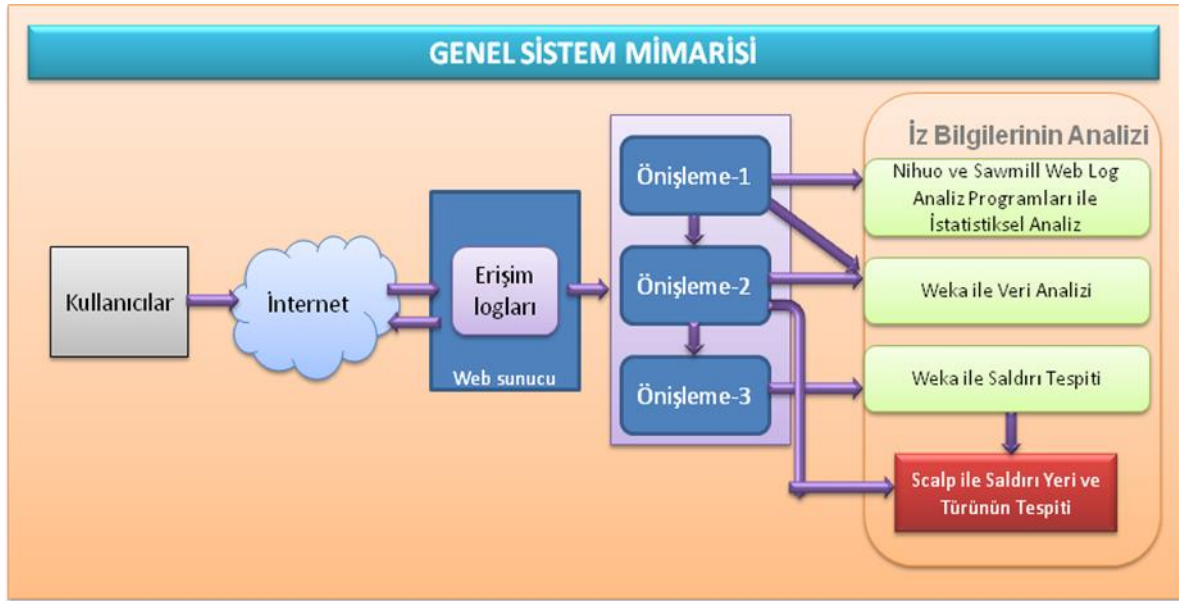
Hata Günlüğü Dosyasının Özellikleri	
Dosyanın Adı	error_ABC.log
Dosyanın Boyutu	86.6 KB
Tarih Aralığı	26.08.2012-31.10.2012
Hata Sayısı	430

İz bilgisi analizi ile ilgili yapılan çalışmalarda geçmişten günümüze farklı yöntemlerin kullanıldığı görülmüştür.

Tez çalışmasında iz bilgileri bundan sonraki aşamada 3 farklı yöntemle analiz edilmiştir.

1. Literatür taramalarında sıkça karşılaşılan ve birçok tez çalışmasında özellikle web kullanım madenciliği amacıyla kullanılan araçlardan Nihuo-4 ve Sawmill 8.7.5 web iz bilgisi analiz ve raporlama programları kullanılarak web erişim iz bilgilerinden istatistiksel analizler yapılmıştır.
2. Web madenciliğinde yaygın olarak kullanılan algoritmalar WEKA yazılımı kullanılarak çalıştırılmış; istatistiksel analiz ve saldırı tespitine yönelik çeşitli çıkarımlarda bulunulmuştur.
3. WEKA’dan elde edilen sonuçlar ışığında filtrelenmiş özel veri kullanılarak web saldırı tespitinde kullanılan Apache Scalp aracında saldırı ve açıklık tespiti gerçekleştirilmiştir.

Genel sistem mimarisi Şekil 5.7’de sunulmuştur.



Şekil 5.7. Genel sistem mimarisi

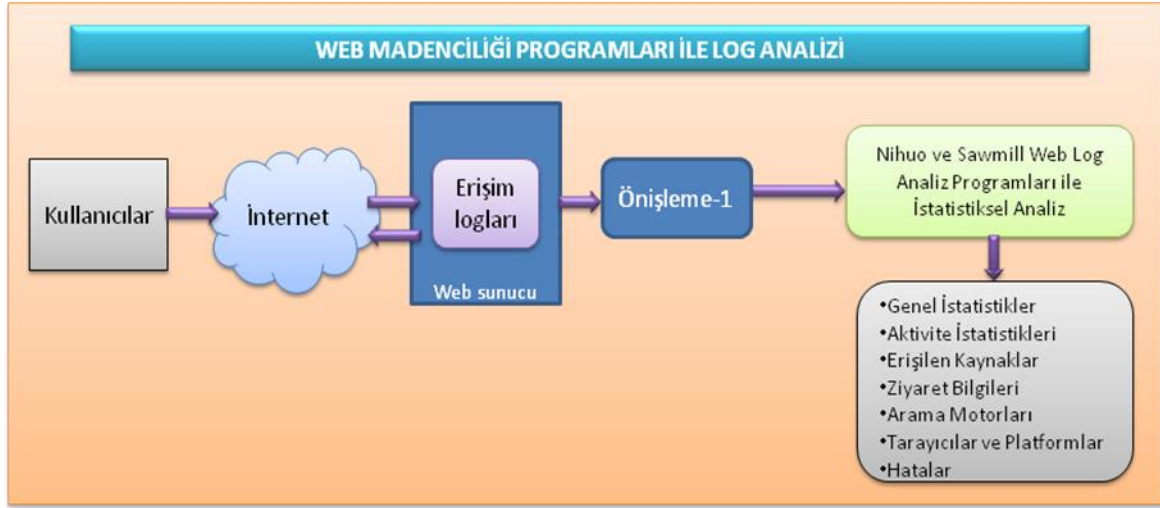
5.2.1. Web madenciliği programları ile iz bilgisi analizi

Önişlemden geçirilmiş olan veriyi, istatistiksel olarak incelemek için web iz bilgisi analizini gerçekleştiren birçok uygulama incelenmiş; çevrimdışı analize olanak sağlayan, raporlama performansı yüksek, yüksek kapasiteli iz bilgisi dosyalarını başarıyla inceleyen Sawmill ve Nihuo Web İz Bilgisi Analiz Programları tercih edilmiştir.

Bu analizlerle sitenin sayfa yapısı, içeriği, kırık linkler vb. konularda elde edilen bulgularla site gelişimine katkı sağlanması hedeflenmektedir. Erişim iz bilgilerinden çıkarılan yararlı istatistiksel bilgiler sayesinde web sitesinin gelişimine katkı sağlanacak ve bu bilgiler web site yöneticisi/tasarımcılarına yol gösterici olacaktır. Aynı zamanda sitenin iyileştirilmesiyle müşteri potansiyelinin de artması hedeflenmektedir.

Sawmill iz bilgisi dosya analizi ve raporlaması için kullanılan ticari bir yazılımdır [70]. Bu yazılımın sınırlı bir süre için sağlanan deneme sürümü kullanılarak analizde neler elde edilebileceği incelenmiştir. Sawmill’in içerisindeki çeşitli raporlamalarda hiyerarşik veri madenciliği analizi yapılabildiği belirtilmektedir [71].

Nihuo Web İz Bilgisi Analiz Programı da yine ticari bir yazılımdır. Nihuo, 80 farklı raporla, 2 ve 3 boyutlu grafiklerle geniş ölçekli bir raporlama ve istatistik sağlamaktadır [72]. Bu tez çalışmasında her iki programın da deneme sürümleri kullanılmış ve elde edilen sonuçlar karşılaştırılarak ve sentezlenerek genel bir rapor elde edilmiştir. Şekil.5.8’de uygulama şeması sunulmuştur.



Şekil 5.8. Web madenciliği programları ile iz bilgisi analizi uygulama şeması

Bu programlar sayesinde aşağıdaki konularla ilgili grafiksel veya çizelge olarak çeşitli raporlar alınabilmektedir.

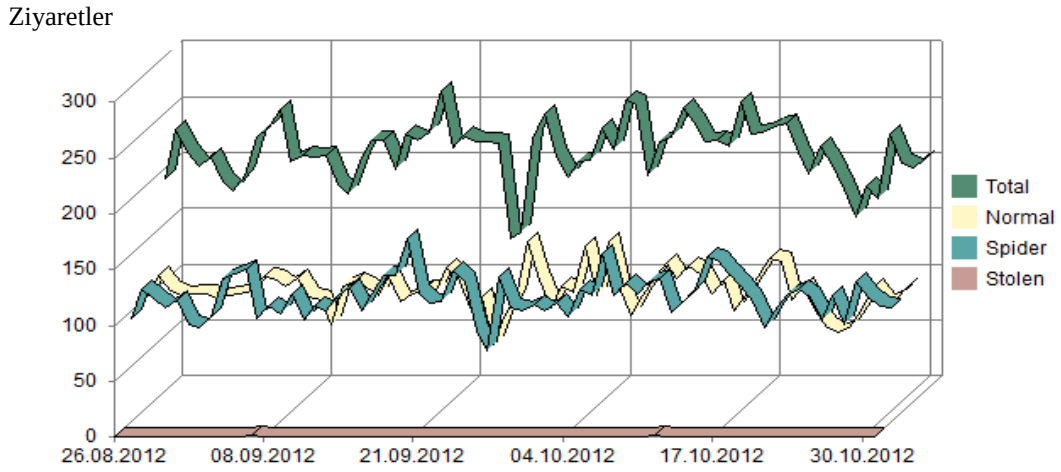
- Genel İstatistikler: Sayfa görüntüleme, ziyaretler, bant genişliği vb. hakkında bilgi verilmektedir.
- Aktivite İstatistikleri: Aktivite bilgileriyle ilgili günlük, günün saatleri, haftanın günleri, aylık olarak analiz yapılabilir.
- Erişilen Kaynaklar: URL’ler, sayfalar, indirilen dosyalar, dizinler, giriş yapılan sayfalar, çıkış yapılan sayfalar, dosya tipleri vb. hakkında analiz yapılabilir.
- Ziyaret Bilgileri: Ziyaretler, bant genişliği, ülkeler, şehirler vb. hakkında analiz yapılabilir.
- Arama Motorları: En çok kullanılan arama motorları, aranan kelimeler, aranan dosyalar vb. hakkında analiz yapılabilir.
- Tarayıcılar ve Platformlar: En çok kullanılan tarayıcılar, platformlar, mobil cihazlar, erişim dili vb. hakkında analiz yapılabilir.

- Hatalar: 404 hataları, sunucu hataları, diğer istemci hataları hakkında bilgi alınabilmektedir.

Aşağıda bu analizlerin bir kısmına yer verilmiştir.

Genel istatistikler

Genel ziyaretçi istekleri ile ilgili bilgi Şekil.5.9’da sunulmuştur. Mavi renkle web robotlarının göndermiş olduğu istekler, sarı renkle ise normal ziyaretçi istatistikleri gösterilmektedir. Her iki istek sayılarının analizin gerçekleştirildiği 26.08.2012 – 30.10.2012 tarihleri arasında birbirine çok yakın olduğu görülmektedir. Bu durumda kullanıcılarla ilgili yapılacak istatistiklerde web robotları olmadan analiz yapmak daha doğru sonuçlar verecektir.



Şekil 5.9. Ziyaretlerle ilgili genel istatistikler grafiği

Sayfa görüntüleme, sayfa ziyaretleri, bant genişliği kullanımı ile ilgili genel istatistiksel bilgiler Şekil 5.10’da sunulmuştur.

General Statistics	
Hits	
Total Hits	64.426
Normal Hits	42.357
Spider Hits	22.062
Stolen Object Hits	7
Average Hits per Day	961
Average Hits per Visit	4,51
Cached Requests	6
Failed Requests	5.433
Page Views	
Total Page Views	62.062
Average Page Views per Day	926
Average Page Views per Visit	4,35
Unique Page Views	39.091
Visits	
Total Visits	14.276
Normal Visits	6.876
Spider Visits	7.398
Stolen Object Visits	2
Average Visits per Day	213
Total Unique Visitors	5.925
Total Visitor Stay Length	1.583:52:15
Average Visitor Stay Length	6:39
Bandwidth	
Total Bandwidth	1,15 GB
Normal Bandwidth	561,25 MB
Spider Bandwidth	605,54 MB
Stolen Object Bandwidth	6,49 MB
Average Bandwidth per Day	17,51 MB
Average Bandwidth per Hit	18,65 KB
Average Bandwidth per Visit	84,16 KB

Şekil 5.10. Ziyaret genel istatistikleri ile ilgili detaylı bilgi

Aktivite istatistikleri

Aktivite istatistiklerine göre haftanın en yoğun günü Pazar günü iken en az aktivite olan günü ise Perşembe'dir (Şekil 5.11).

Summary	
Summary of Activity	
Average Number of Visits per Day on Weekdays	215
Average Number of Hits per Day on Weekdays	957
Average Number of Visits per Weekend	207
Average Number of Hits per Weekend	972
Most Active Day of the Week	Pazar
Least Active Day of the Week	Perşembe
Most Active Date	Salı, 04 Eylül, 2012
Number of Hits on Most Active Date	1.414
Number of Visits on Most Active Date	267
BandWidth on Most Active Date	56,68 MB
Least Active Date	Çarşamba, 24 Ekim, 2012
Number of Hits on Least Active Date	508
Number of Visits on Least Active Date	135
BandWidth on Least Active Date	6,38 MB
Most Active Hour of the Day	16:00 - 16:59
Least Active Hour of the Day	05:00 - 05:59

Şekil 5.11. Aktivite istatistikleri özet bilgisi

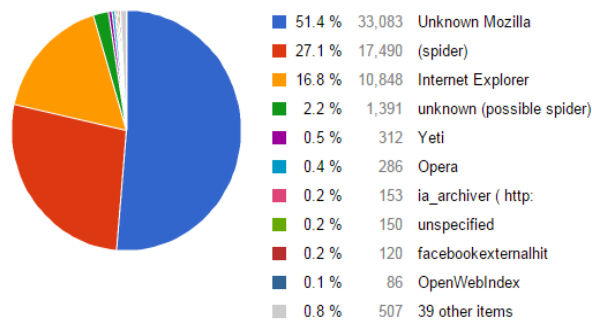
En çok kullanılan işletim sistemleri ve tarayıcılar

Şekil 5.12’de en çok kullanılan işletim sistemleri listesi gösterilmiştir. İşletim sistemi bilinen en çok Windows 7 kullanılan bilgisayarlardan erişim sağlandığı ve yaygın olarak Microsoft tabanlı işletim sistemlerinin tercih edildiği görülmektedir.

Operating system	↓ Hits	Visitors	Size	Page views	Sessions	Session duration	Bounces	Bounce Rate
1 unknown	46,776 72.6 %	3,916	903.63 M	45,507	11,067	61d 21:40:30	6,085	54.98%
2 Windows 7	8,753 13.6 %	1,324	121.32 M	8,709	1,742	1d 10:22:17	557	31.97%
3 Windows XP	4,466 6.9 %	611	57.71 M	4,436	650	15:53:12	59	9.08%
4 Android	1,092 1.7 %	220	11.88 M	1,092	251	04:09:37	50	19.92%
5 Linux	918 1.4 %	130	19.00 M	900	438	01:51:28	289	65.98%
6 Windows Vista	860 1.3 %	121	35.94 M	854	126	03:23:06	5	3.97%
7 Macintosh	651 1.0 %	80	6.79 M	651	95	03:06:51	14	14.74%
8 Windows	361 0.6 %	42	4.53 M	344	44	00:43:55	5	11.36%
9 BlackBerry 6/7	178 0.3 %	26	1.93 M	178	115	00:22:41	78	67.83%
10 unspecified	150 0.2 %	10	2.15 M	150	18	00:09:39	6	33.33%
11 other items	221 0.3 %	56	8.39 M	219	–	00:33:13	10	155.24%
Total	64,426 100.0 %	6,536	1.15 G	63,040	–	64d 14:16:29	7,158	–

Şekil 5.12. En çok kullanılan işletim sistemleri

Şekil 5.13’de ise en çok kullanılan tarayıcıların listesi gösterilmiştir. Burada kesin olarak tespit edilemeyen ancak Mozilla’yı da içeren tarayıcı oranı %51.4 iken, bunu %27.1 ile web robotları ve %16.8 ile Internet Explorer tarayıcısının takip ettiği görülmektedir.



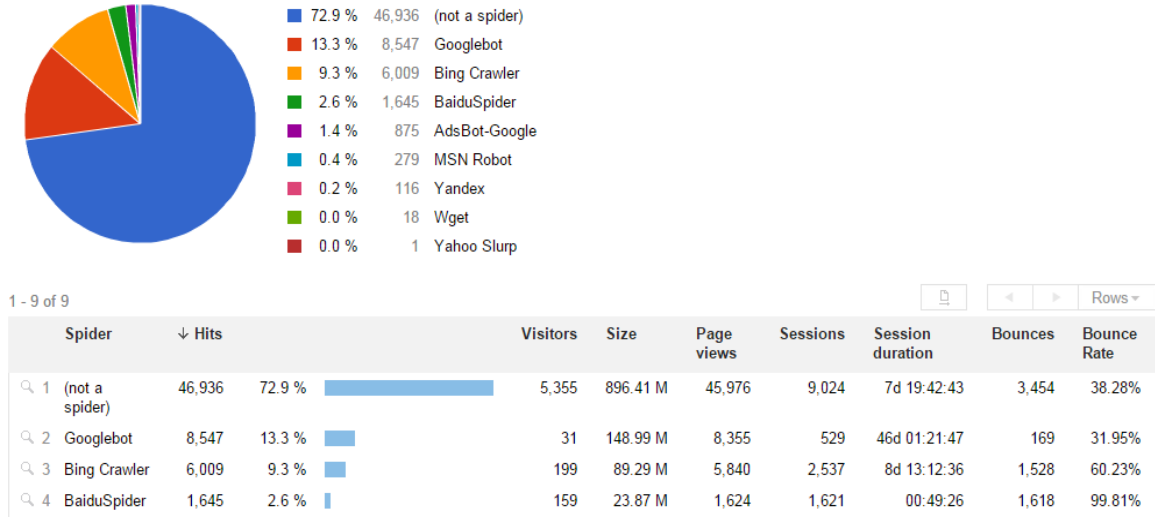
1 - 10 of 49

Web browser	↓ Hits	Visitors	Size	Page views	Sessions	Session duration	Bounces	Bounce Rate
1 Unknown Mozilla	33,083 51.4 %	3,728	642.11 M	32,526	6,477	4d 11:05:31	2,777	42.87%
2 (spider)	17,490 27.1 %	605	276.87 M	17,064	5,225	56d 18:33:46	3,704	70.89%
3 Internet Explorer	10,848 16.8 %	1,467	165.69 M	10,567	1,860	1d 18:46:39	255	13.71%

Şekil 5.13. En çok kullanılan tarayıcılar

Arama robotları

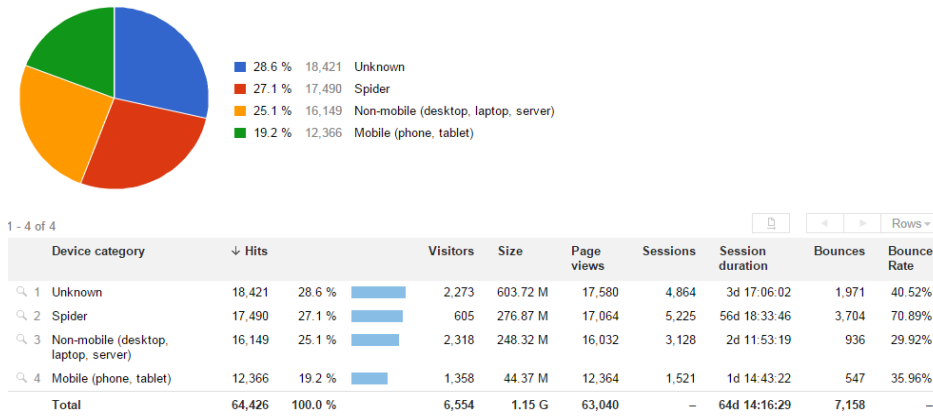
Şekil.5.14’de arama robotları (crawler/spider/bot) grafiği ve listesi gösterilmiştir. Googlebot arama robotları içinde ilk sırada yer almaktadır.



Şekil 5.14. En çok kullanılan arama robotları

En çok erişimin sağlandığı araç tipleri ve en çok kullanılan mobil cihazlar

Şekil 5.15’de en çok erişimin sağlandığı araç tipleri grafiği ve listesi gösterilmiştir. Burada mobil olmayan cihazlarla erişim oranı %25.1 iken mobil cihazlarla erişim oranının %19.2 olduğu görülmektedir.



Şekil 5.15. En çok erişimin sağlandığı araç tipleri

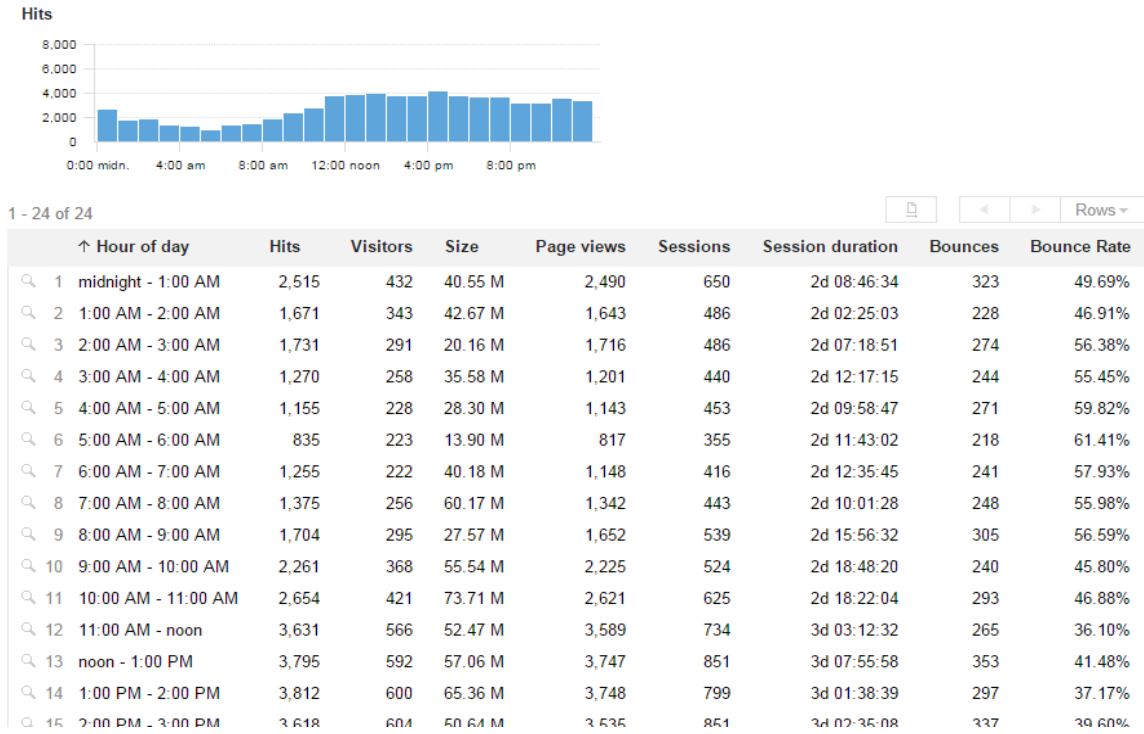
Şekil 5.16’da en çok kullanılan mobil cihazların listesi gösterilmiştir. %80 oranla iPhone kullanımının çok yaygın olduğu görülmektedir.



Şekil 5.16. En çok kullanılan mobil cihazlar

Günün saatlerine göre erişim analizi

Şekil 5.17’de sunulan grafik ve listede özellikle öğleden sonra artan bir erişim grafiği görülmektedir. Web sayfalarının bakımıyla ilgili çalışmalar erişimin en az sağlandığı saat aralığı olan 04:00 ve 08:00 arasında yapılabilir.



Şekil 5.17. Günü saatlerine göre erişim analizi

Erişimin sağlandığı IP numaralarına göre analiz

Şekil 5.18’de erişimin sağlandığı IP numaraları, kaç kez erişim sağladıkları ve oturum sayıları gibi bilgiler görülmektedir.

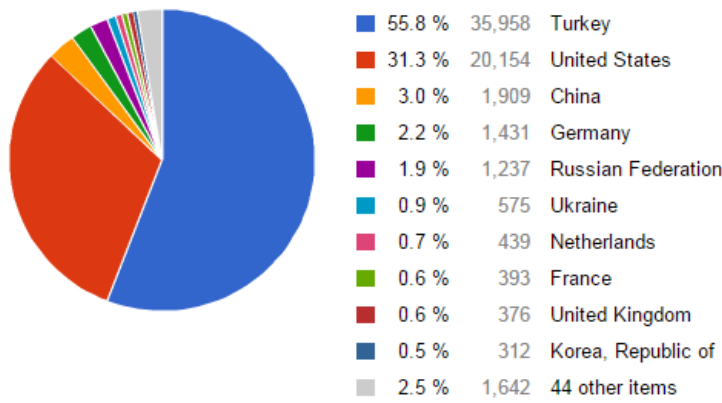
1 - 10 of 5925

Hostname	↓ Hits	Visitors	Size	Page views	Sessions	Session duration	Bounces	Bounce Rate
1 66.249.72.6	2,495 3.9 %	1	35.55 M	2,492	93	13d 22:37:26	25	26.88%
2 66.249.76.130	1,477 2.3 %	1	26.95 M	1,424	115	8d 07:03:58	32	27.83%
3 66.249.73.209	1,435 2.2 %	1	20.22 M	1,398	92	7d 14:46:40	26	28.26%
4 208.115.113.91	1,100 1.7 %	1	50.76 M	1,096	656	02:36:01	436	66.46%
5 66.249.71.140	996 1.5 %	1	14.49 M	963	69	5d 07:21:44	24	34.78%
6 66.249.72.76	931 1.4 %	1	20.42 M	924	27	4d 14:19:46	6	22.22%
7 217.69.133.28	826 1.3 %	1	61.30 M	545	29	00:36:42	20	68.97%
8 208.115.111.75	674 1.0 %	1	27.51 M	670	480	02:31:10	366	76.25%
9 72.14.199.216	444 0.7 %	1	5.93 M	423	108	1d 04:40:16	48	44.44%
10 157.55.32.115	430 0.7 %	1	12.16 M	417	174	13:55:36	92	52.87%
5915 other items	53,618 83.2 %	5,915	897.99 M	52,688	-	22d 19:47:10	6,083	166,763.33%
Total	64,426 100.0 %	5,925	1.15 G	63,040	-	64d 14:16:29	7,158	-

Şekil 5.18. Erişimin sağlandığı IP numaralarına göre analiz

En çok ziyaret eden ülkeler ve şehirler

Erişimin en çok sağlandığı ülkeler Şekil 5.19’da ve erişimin en çok sağlandığı şehirler de Şekil 5.20’de gösterilmiştir. Hem ülke hem de şehir bazında Türkiye’den sonra Amerika Birleşik Devletlerinin büyük bir orana sahip olduğu görülmektedir. Bu durum arama motorlarının büyük oranda Amerika’dan erişim sağlıyor olmalarından kaynaklanmaktadır.



Şekil 5.19. En çok ziyaret eden ülkeler

City	↓ Hits	Visitors	Size	Page views	Sessions	Session duration	Bounces	Bounce Rate
1 (unknown region)	16,729 26.0 %	1,786	360.39 M	16,223	2,865	2d 13:36:57	1,060	37.00%
2 İstanbul (İstanbul, Turkey)	12,496 19.4 %	1,547	121.97 M	12,445	1,678	1d 18:00:50	237	14.12%
3 Mountain View (CA, United States)	9,645 15.0 %	47	164.48 M	9,408	767	48d 23:36:25	286	37.29%
4 Ankara (Ankara, Turkey)	7,190 11.2 %	900	75.70 M	7,156	972	1d 02:19:12	103	10.60%
5 Redmond (WA, United States)	6,340 9.8 %	559	94.20 M	6,181	3,123	7d 22:40:43	2,164	69.29%
6 Seattle (WA, United States)	1,774 2.8 %	2	78.27 M	1,766	1,136	05:07:11	802	70.60%
7 Beijing (Beijing, China)	1,749 2.7 %	155	25.07 M	1,727	1,692	01:53:57	1,666	98.46%
8 İzmir (İzmir, Turkey)	658 1.0 %	67	3.47 M	655	69	01:14:11	5	7.25%
9 Berlin (Berlin, Germany)	435 0.7 %	47	4.01 M	435	233	00:27:39	136	58.37%
10 Ashburn (VA, United States)	333 0.5 %	9	11.58 M	331	62	06:14:48	9	14.52%
262 other items	7,077 11.0 %	806	234.14 M	6,713	–	1d 15:04:36	690	5,384.58%
Total	64,426 100.0 %	5,925	1.15 G	63,040	–	64d 14:16:29	7,158	–

Şekil 5.20. En çok erişimin sağlandığı şehirler

Sayfaya erişimde en çok kullanılan arama motorları

Şekil 5.21’de sunulan rapora göre tespit edilen arama robotları arasında sayfaya en çok Google arama motoru aracılığı ile ulaşılmıştır.

	Search engine	↓ Hits		Visitors	Size	Page views	Sessions	Session duration	Bounces	Bounce Rate
1	(empty)	59,324	92.1 %	5,476	1.03 G	57,940	13,631	64d 02:57:33	6,683	49.03%
2	Google	5,065	7.9 %	3,634	115.86 M	5,063	3,978	11:15:35	470	11.81%
3	Bing	30	0.0 %	21	644.47 K	30	25	00:03:15	1	4.00%
4	Yahoo	7	0.0 %	7	131.13 K	7	7	00:00:06	4	57.14%
	Total	64,426	100.0 %	9,138	1.15 G	63,040	–	64d 14:16:29	7,158	–

Şekil 5.21. Sayfaya erişimde en çok kullanılan arama motorları

5.2.2. WEKA ile örüntü keşfi ve analiz

WEKA hakkında genel bilgi

WEKA makine öğrenme ve veri madenciliği için kapsamlı bir araçtır [73]. WEKA, Zelanda WBuaikato Üniversitesi'nde bir proje olarak geliştirilmeye başlanmış olup bugün dünyada birçok kişi tarafından kullanılan açık kaynak kodlu ve java platformu üzerinde geliştirilmiş bir programdır [74].

WEKA birçok sınıflandırma tekniği içermekle birlikte uygulamaların komut girilerek gerçekleştirilmesine de olanak tanımaktadır [75]. WEKA geniş bir öğrenme algoritmaları yelpazesine sahip olduğu gibi veri ön işleme için birçok filtreleme seçeneği de içermektedir. WEKA bünyesinde 4 temel uygulamayı barındırmaktadır [76]:

- Explorer; çeşitli veri madenciliği algoritmalarını barındıran, veri ön işleminin yapılabileceği, kullanımı kolay bir arayüzdür.
- Experimenter; eldeki probleme hangi metodun, hangi parametrelerle uygulandığında daha iyi bir sonuç alınabileceğini analiz etmeye yaramaktadır.
- Knowledge Flow; grafiksel olarak taşı-bırak metodu ile çeşitli veri işleme metodları ikonlarının ekrana taşınarak bir veri akış diyagramı ile veri madenciliği uygulamalarının yapılmasını sağlamaktadır.
- Simple CLI; metin tabanlı bir konsoldan komut satırlarının girilerek veri madenciliğinin uygulanabilmesine imkan vermektedir.

WEKA'ya Verilerin Yüklenmesi

WEKA'ya verilerin yüklenmesi için çeşitli yöntemler bulunmaktadır:

- Verileri dosyadan açma
- Verileri URL'den açma
- Verileri veritabanından açma

WEKA yazılımında ARFF (Attribute-Relation File Format) olarak adlandırılan değişken tanımlanmasına izin veren dosyalar kullanılmaktadır. ARFF dosyaları, ASCII metin dosyalarıdır [77]. ARFF, veriler için özellik adı, özellik tipi, özellik değeri ve verinin kendisi için özel etiketler içeren bir dosya formatıdır [78]. Excel formatında hazırlanmış olan veri setinden ARFF dosyalarına dönüşüm yapılarak WEKA'ya eklenmektedir.

Bu tez çalışmasında ilk yöntem olarak verilerin ARFF formatına dönüştürülerek dosya olarak WEKA'da açılması sağlanmıştır. Ancak karşılaşılan zorluklar nedeniyle çalışmaların devamında veriler veritabanı aracılığı ile WEKA'ya aktarılmıştır. Aşağıdaki bölümlerde yapılan işlemler ve karşılaşılan problemlerle ilgili detaylı bilgi verilmiştir.

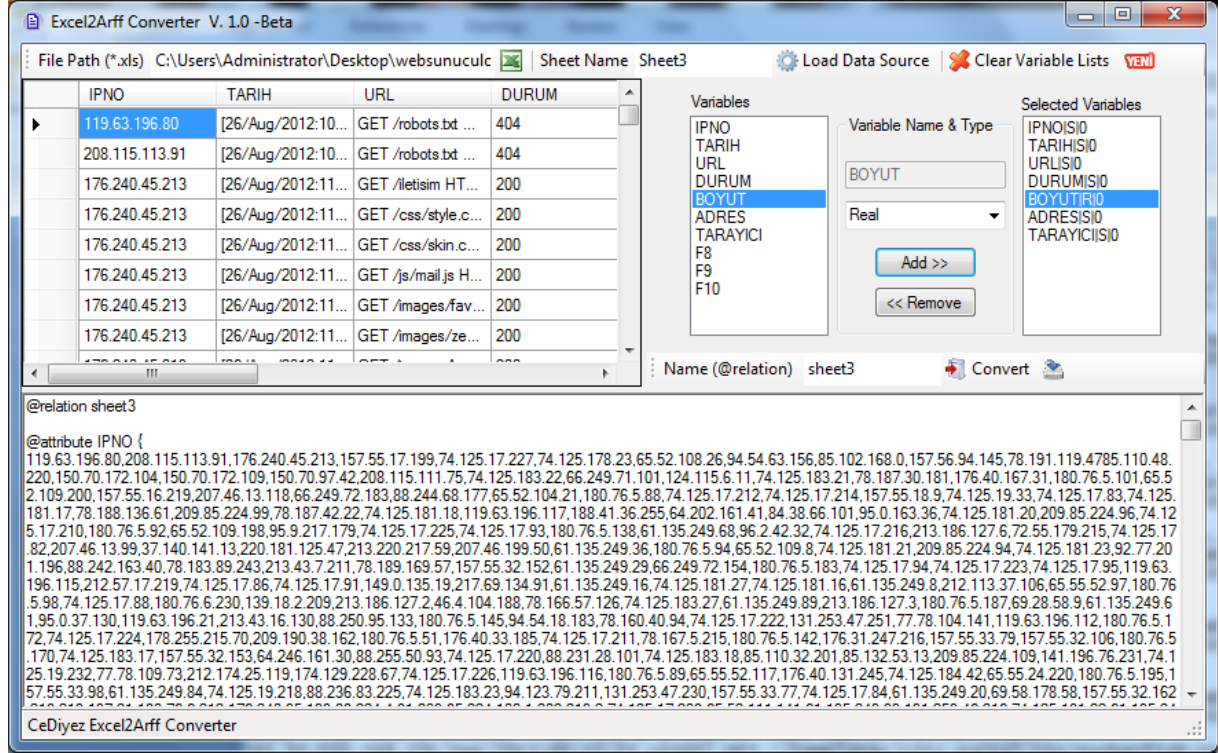
Verilerin ARFF Formatına Dönüştürülerek Dosya Olarak Eklenmesi

İncelenen çalışmaların birçoğunda verileri WEKA'ya yüklemek için ARFF formatına dönüşüm işlemi gerçekleştirilmiştir [77-79, 91-93]. Önişlemden geçirilen veri setinin WEKA programına yüklenebilmesi için ARFF formatına dönüştürülmesi gerekmektedir.

WEKA'da çalışma yapmak için izlenecek adımlar aşağıdaki gibi belirlemiştir [79]:

- Öncelikli olarak çalışma yapılacak olan veri seti hazırlanır.
- Hazırlanan veri seti ARFF dosyasına dönüştürülür.
- WEKA yazılımına ARFF dosyası yüklenir.
- WEKA'da gerekli olan ilgili modüller yüklenir.
- Parametreler belirlenir ve gerekli ayarlar yapılır.
- Sonuçlar alınır ve değerlendirilir.

Excel dosyasını ARRF formatına dönüştürmek için Şekil 5.22’de görülmekte olan Excel2Arff Converter [80] adlı program kullanılmıştır.



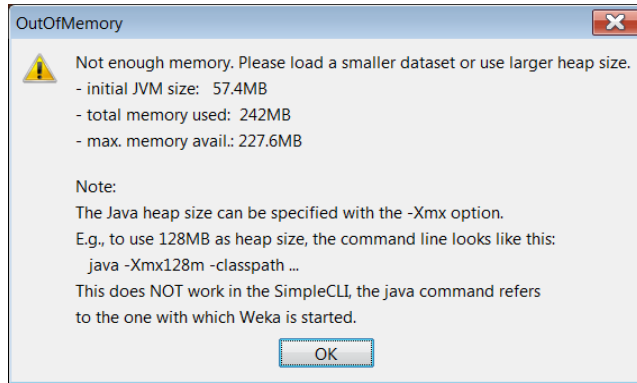
Şekil 5.22. ARRF dönüştürücü program

Bu programda dönüştürülmek istenen işlemiş olan iz bilgisi dosyası yüklendikten sonra değişkenlerin tipleri belirlenmekte ve son olarak dönüşüm işlemi tamamlanmaktadır. Önişlemeden geçirilmiş olan verinin ARRF Converter programında dönüştürülme işleminden sonra WEKA’da ARRF dosyası açılmak istendiğinde birçok hata ile karşılaşmıştır.

ARRF Converter programı ile dönüşüm yapıldığı sırada dosyanın içeriğine ARRF formatına uymayan “/par” gibi yabancı karakterler eklendiği görülmüştür ve bu karakterler temizlenmiştir.

Hataların diğer sebepleri arasında ARRF formatında tek tırnak (‘), noktalı virgül (;), boşluk() gibi karakterlerin bulunması yer almaktadır. Bu karakterler Excel ortamında temizlendikten sonra yeniden ARRF formatına dönüşüm sağlanmıştır. Bu kez işlenen

verinin çok büyük olmasından kaynaklanan bellek yetersizliği hatası alınmıştır (Şekil 5.23).



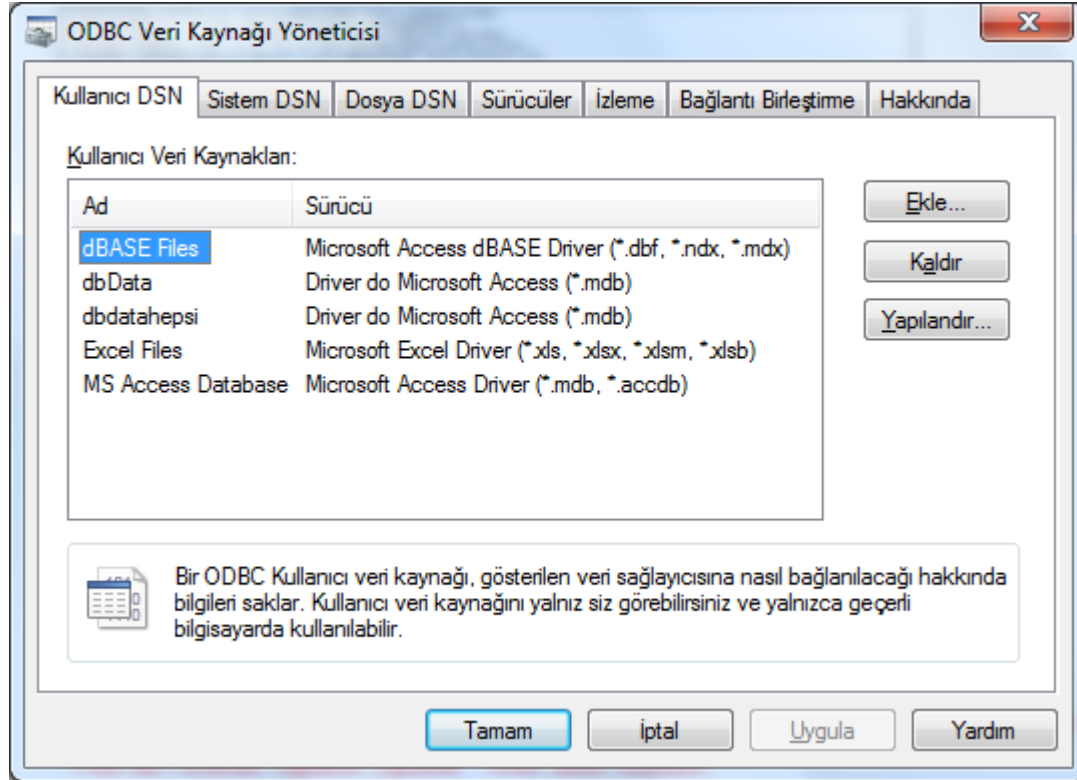
Şekil 5.23. Bellek yetersizliği hatası

Bu hatayı gidermek için WEKA'nın kullandığı bellek miktarı artırılmıştır.

Verilerin Veritabanı Bağlantısı Kurularak Yüklenmesi

WEKA'da ARRF formatının kullanımı sırasında karşılaşılan zorluklardan dolayı daha etkin bir yöntem olan "WEKA'da veritabanı bağlantısı yapılarak" veriler analiz edilmiştir. WEKA'da veritabanı bağlantısı kurmak oldukça karmaşık bir yapıdadır ancak gerekli ayarlar yapıldıktan sonraki aşamada veritabanında değişiklikler yapılsa bile WEKA'ya verilerin yüklenmesi konusunda problem yaşanmamıştır.

1. Adımda; Denetim Masası → Yönetimsel Araçlar → Veri Kaynağı Yöneticisi (ODBC) adımları izlenerek Kullanıcı DSN sekmesinden MS Access sürücüsü eklenmiştir ve DSN (Data Source Name) alanında veri kaynağı adı verilmiştir. Daha sonra veritabanı dosyası bulunarak seçilmiştir. Böylece kullanıcı veri kaynakları listesinde yeni oluşturulan DSN görülmüştür (Şekil 5.24).



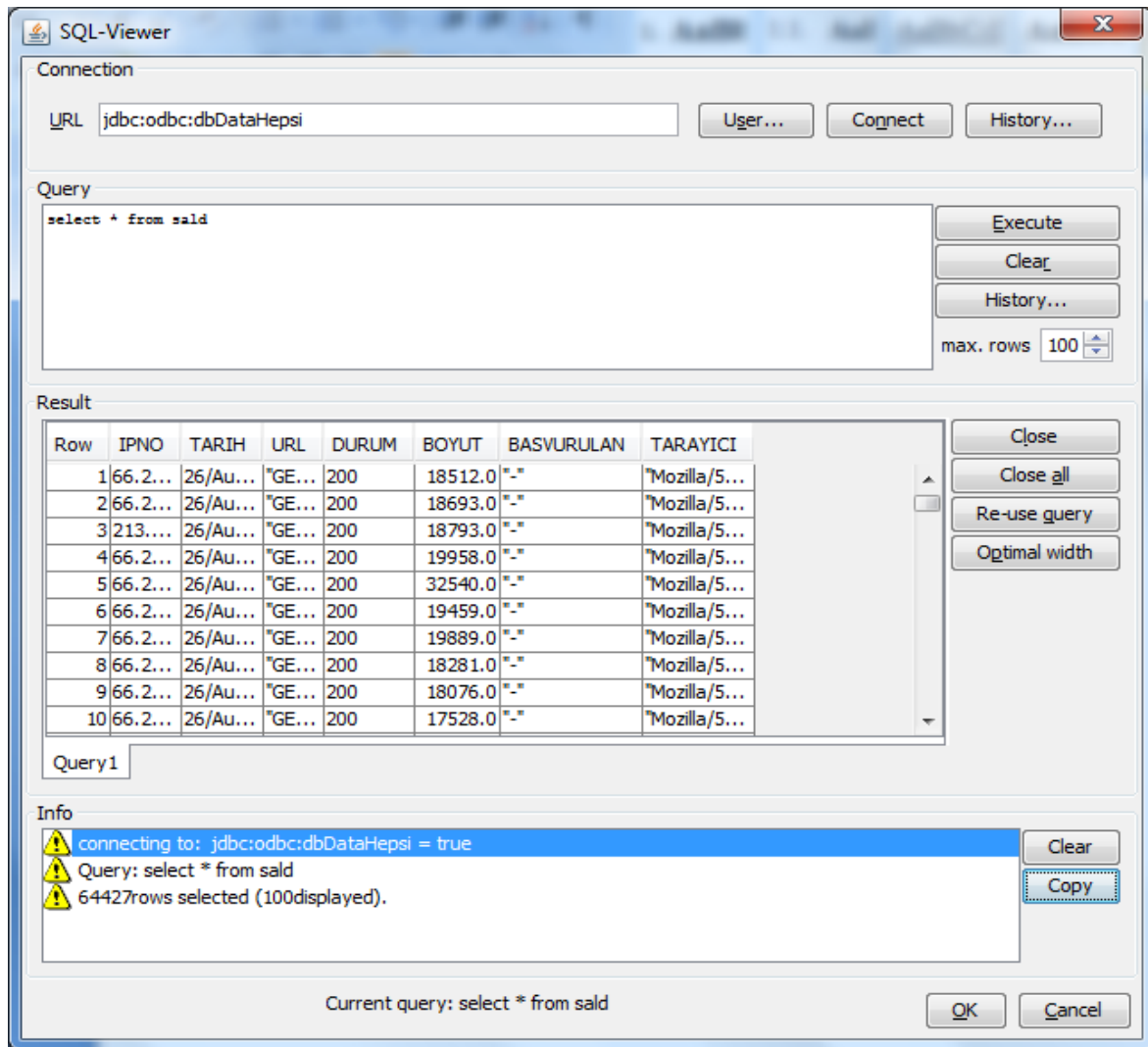
Şekil 5.24. DSN oluşturulması

- Adımda; WEKA.jar dosyasında yer alan DatabaseUtils.props dosyasındaki CLASSPATH'e aşağıdaki satırlar eklenmiştir. Buradaki dbname DSN'e verilmiş olan isimdir.

```
jdbcDriver=sun.jdbc.odbc.JdbcOdbcDriver
```

```
jdbcURL=jdbc:odbc:dbname
```

- sAdımda; WEKA çalıştırılarak "Open DB" sekmesi seçilmiştir. Açılan pencerede DSN'e verilen veritabanı adı yazılmıştır. "connecting to: jdbc:odbc:dbDataHepsi = true" uyarısı alınmıştır. Burada sorgu penceresinden "select" cümlecisi ile tüm kayıtlar çekilerek sonuç bölümünde 100 satır için tablo görüntülenerek veriler başarıyla WEKA'ya yüklenmiştir (Şekil 5.25).

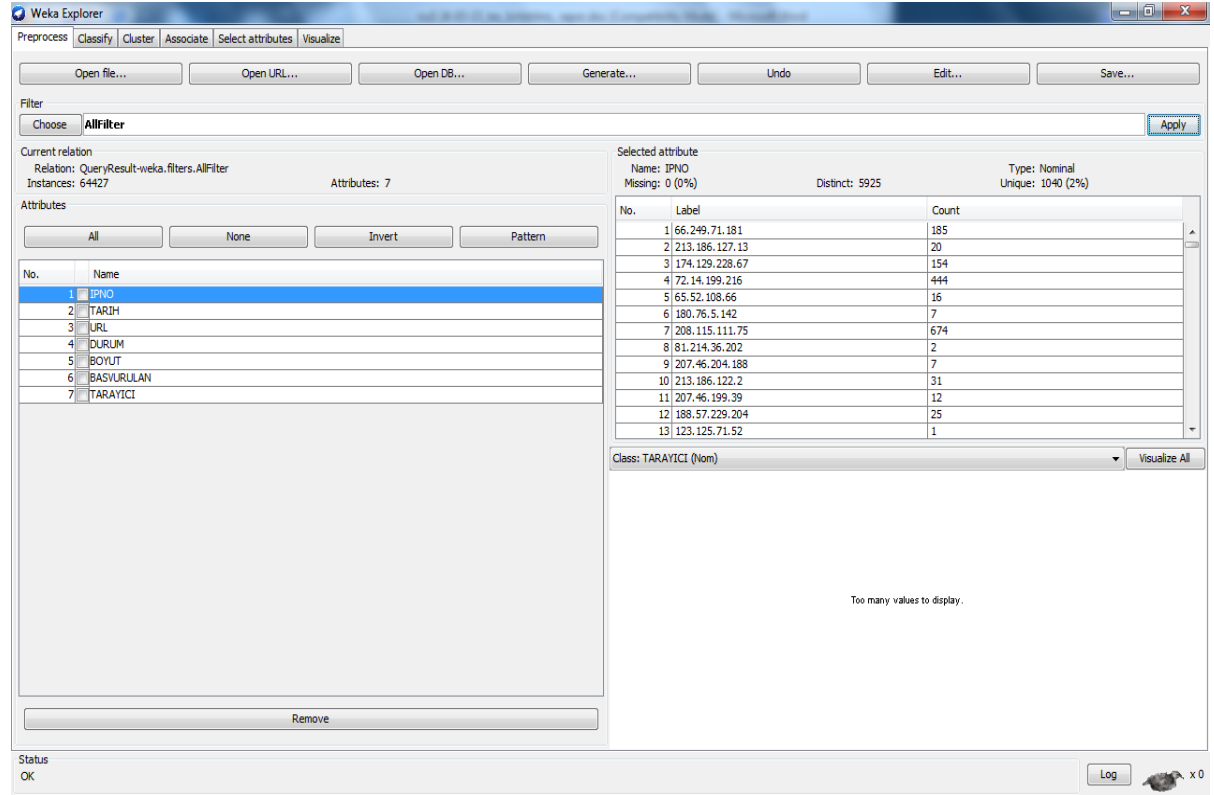


Şekil 5.25. WEKA'dan MS Access veritabanına yapılan bağlantı ekranı

Bu tez çalışmasında, WEKA ile daha önce bahsedilen çeşitli amaçlara yönelik olarak işlenmiş olan 3 veri dosyasıyla da işlem yapılmış ve sonuçlar ilerleyen bölümlerde değerlendirilerek karşılaştırılmıştır.

Önişleme-1 verisi ile analiz

WEKA ile örüntü keşfi ve analiz için önişleme-1 adımıdan elde edilen veri tablosu, veritabanı bağlantısıyla yüklenmiştir. WEKA'ya veriler yüklendikten sonra Şekil 5.26'da WEKA içerisinde yer alan önişlem adımı görüntülenmektedir. Bu adımda çeşitli filtreleme işlemleri yapılabileceği gibi istatistiksel sonuçlar da elde edilebilmektedir.



Şekil 5.26. WEKA’da önişlem penceresi

Erişim iz bilgisi dosyasından elde edilen istatistiksel sonuçlar Çizelge 5.4’de gösterilmiştir.

Çizelge 5.4. Erişim iz bilgisi dosyası istatistiksel sonuçlar

Erişimler (Önişleme-1 için)	
Toplam Veri	685.272
Önişlemden Sonra Veri Sayısı	64.427
Günlük Ortalama Erişim	961.59
Ziyaretçi Başına Ortalama Erişim	10.87
Başarılı İstek Sayısı (200 serili durum kodu)	57.561
Hatalı İstek Sayısı (400 serili durum kodu)	1393
Ziyaretçi	
Toplam Ziyaretçi	5925
Ortalama Günlük Ziyaretçi	88.43

Hata iz bilgisi dosyasından elde edilen istatistiksel sonuçlar Çizelge 5.5’de gösterilmiştir.

Çizelge 5.5. Hata iz bilgisi dosyası istatistikler

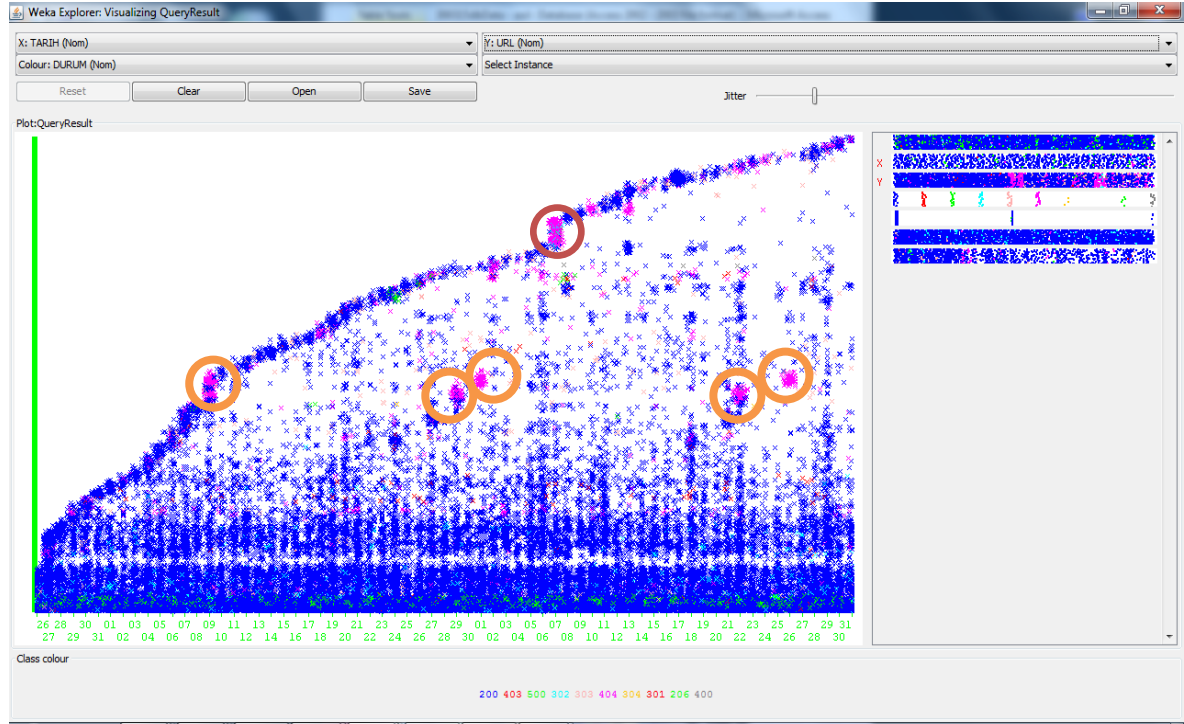
Hata İz Bilgisi Dosyası	
Toplam Veri	430
Alınan Hatalar	Dosya bulunamadı- 420 kez
	Sunucu ayarlarıyla istemci reddedildi- 1 kez
	İstek içerisinde geçersiz URI- 9 kez

İşlenmiş olan erişim iz bilgilerinin detaylı analizi amacı ile web iz bilgilerinin incelenmesinde sık kullanılan algoritmalar uygulanarak çeşitli sonuçlar elde edilmiştir.

WEKA’nın Visualize sekmesinde analiz edilen verinin istenilen alanları arasındaki ilişki yayılım grafiğinde görülebilmektedir. Bu sekme sayesinde veri analiziyle ilgili çok ilginç çıkarımlar yapılabilir. Bu sekmede alanların dağılımlarına göre kesişim noktaları görülebildiğinden kümeleme analizine olanak sağlamaktadır [81].

Bu sekmenin jitter isimli özelliği kullanılarak grafiğin yoğunluğu artırılıp azaltılabilmekte böylece oluşan kümeler daha net görülebilmektedir.

İncelenen iz bilgisi verisi için Tarih, URL ve durum alanlarına göre işlem yapıldığında Şekil 5.27’deki sonuç elde edilmiştir. Burada X eksenini tarih alanını, Y eksenini URL alanını, ekranın en altında görüldüğü gibi farklı renkler de durum alanını temsil etmektedir. Bu 3 alana göre pembe renk ile temsil edilen 404 hata durum kodu için aşağıda işaretlenen 6 yerde küme oluştuğu görülmektedir. Grafikte yapılan isteklerin başarılı olduğunu gösteren 200 durum kodunu temsil eden mavi rengin yoğun olduğu görülmektedir.



Şekil 5.27. WEKA'nın Visualize sekmesinde analiz edilen verinin görüntüsü

Visualize ekranında koordinat düzleminde yer alan her bir kaydın üstüne tıklandığında detaylı bilgi görüntülenmektedir. Bu kümelerin bulunduğu alanlar bu şekilde detaylı incelendiğinde çoğu kümenin arama robotlarının kısa sürede arka arkaya gerçekleştirdiği indeks kayıtlarından oluştuğu görülmektedir. Şekil 5.28'de arama robotlarının yaptığı isteklerden örnek sunulmuştur. Tarayıcı alanları incelendiğinde isteklerin web robotlarından geldiği görülmektedir.

```

BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; Mail.RU_Bot/2.0)"

Plot : Master Plot
Instance: 56728
IPNO: 217.69.133.67
TARİH: 22/Oct/2012
URL: "GET /tr/10910-the-wrap?format=raw&restoranId=10910&view=restoran_resimler HTTP/1.0"
DURUM: 404
BOYUT: 1370.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; Mail.RU_Bot/2.0)"

Plot : Master Plot
Instance: 56732
IPNO: 217.69.133.67
TARİH: 22/Oct/2012
URL: "GET /tr/10928-mehtap-cafe?format=raw&restoranId=10928&view=restoran_resimler HTTP/1.0"
DURUM: 404
BOYUT: 1370.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; Mail.RU_Bot/2.0)"

Plot : Master Plot
Instance: 56738
IPNO: 217.69.133.67
TARİH: 22/Oct/2012
URL: "GET /tr/10890-vera-restaurant?format=raw&restoranId=10890&view=restoran_resimler HTTP/1.0"
DURUM: 404
BOYUT: 1370.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; Mail.RU_Bot/2.0)"

Plot : Master Plot
Instance: 56741
IPNO: 217.69.133.67
TARİH: 22/Oct/2012
URL: "GET /tr/10887-afrodit-restaurant?format=raw&restoranId=10887&view=restoran_resimler HTTP/1.0"
DURUM: 404
BOYUT: 1370.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; Mail.RU_Bot/2.0)"

```

Şekil 5.28. WEKA'nın Visualize sekmesinde analiz edilen veriyle ilgili detaylı bilgi

Sadece kırmızı renkle işaretlenen küme detaylı incelendiğinde robot olmadığı anlaşılan bir IP numarasından, çok kısa süre içerisinde fazla sayıda istekte bulunduğu ve istek yapılan URL alanı incelendiğinde saldırı denemelerinde bulunduğu tespit edilmiştir. Şekil 5.29'da tarayıcı alanlarından görüldüğü gibi istek herhangi bir web robotundan gelmemektedir. Arka arkaya aynı IP 'den gelmektedir. En önemlisi de istek yapılan URL'de saldırı girişiminde bulunduğu görülmektedir.

```

        URL: "GET /phpMyAdmin/main.php HTTP/1.1"
        DURUM: 404
        BOYUT: 1175.0
3ASVURULAN: "-"
        TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 41750
        IPNO: 65.111.180.205
        TARİH: 07/Oct/2012
        URL: "GET /pma/main.php HTTP/1.1"
        DURUM: 404
        BOYUT: 1175.0
3ASVURULAN: "-"
        TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 41754
        IPNO: 65.111.180.205
        TARİH: 07/Oct/2012
        URL: "GET //WEB-INF/web.xml HTTP/1.1"
        DURUM: 404
        BOYUT: 1175.0
3ASVURULAN: "-"
        TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 41756
        IPNO: 65.111.180.205
        TARİH: 07/Oct/2012
        URL: "GET /db/main.php HTTP/1.1"
        DURUM: 404
        BOYUT: 1175.0
3ASVURULAN: "-"
        TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 41757
        IPNO: 65.111.180.205
        TARİH: 07/Oct/2012
        URL: "GET /?page=../../../../../../../../etc/passwd%00.jpg HTTP/1.1"
        DURUM: 302

```

Şekil 5.29. Saldırı denemelerinde bulunan isteklerle ilgili görüntü

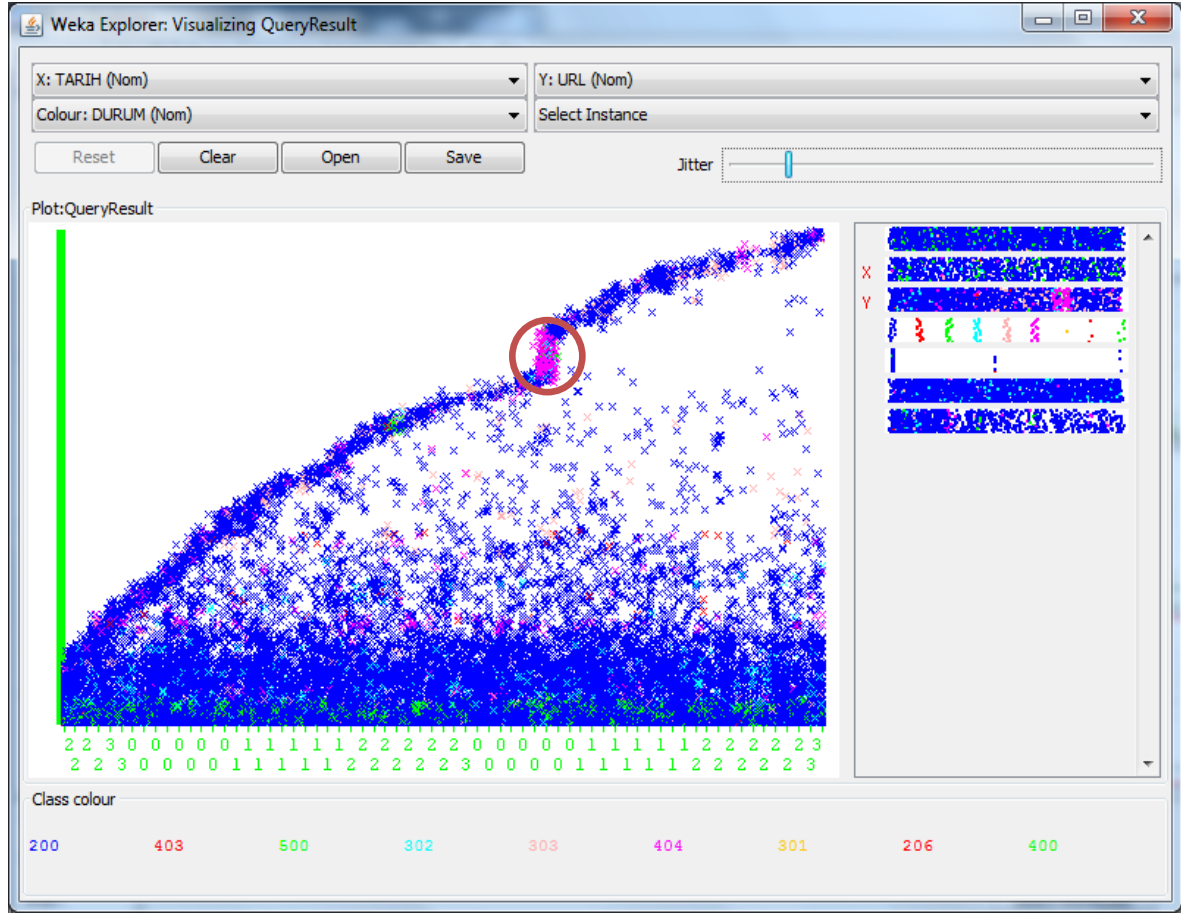
Önişleme-2 verisine göre analiz

Önişleme-1 verisine kümeleme ve birliktelik algoritmaları uygulandığında web robotlarının kayıtlarının çok fazla olmasından dolayı; çıkan sonuçlar kullanıcı, sayfa, saldırı vb. analizlerinde yanılmalara ve bazı durumların gözden kaçmasına sebep olabilmektedir. Dolayısıyla web robotlarının indekslediği sayfaların yer aldığı iz bilgisi satırları temizlenerek elde edilen Önişleme-2 verisine kümeleme ve birliktelik algoritmaları uygulanarak daha anlamlı sonuçlar elde edilmiştir. Çizelge 5.6’da önişleme-2 verisi için genel istatistiksel sonuçlar gösterilmiştir.

Çizelge 5.6. Erişim iz bilgisi dosyası istatistiksel sonuçlar

Erişimler (Önişleme-2 için)	
Toplam Veri	685.272
Önişlemden Sonra Veri Sayısı	44.274
Günlük Ortalama Erişim	660.80
Ziyaretçi Başına Ortalama Erişim	8.21
Başarılı İstek Sayısı (200 serili durum kodu)	38.443
Hatalı İstek Sayısı (400 serili durum kodu)	550
Ziyaretçi	
Toplam Ziyaretçi	5388
Ortalama Günlük Ziyaretçi	80.41

WEKA'nın Vizualize sekmesinden tarih, URL ve durum alanları için yayılım dağılımına bakıldığında Şekil 5.30'da görüldüğü gibi sadece 1 küme oluştuğu ve bu kümenin de daha önce önişleme-1 verisinden elde edilen saldırı girişiminde bulunulduğu tespit edilen kümeyle aynı küme olduğu görülmektedir. Burada X eksenini tarih alanını, Y eksenini URL alanını, ekranın en altında görüldüğü gibi farklı renkler de durum alanını temsil etmektedir. Bu 3 alana göre sadece pembe renk ile temsil edilen 404 hata durum kodu için küme oluşmuştur.



Şekil 5.30. Önışleme-2 verisi yayılım dağılımı

Analizde oluşan küme ile ilgili detaylı bilgi Şekil 5.31’de sunulmuştur. Burada özellikle URL alanları incelendiğinde OWASP’ın listesinde yer alan SQL Enjeksiyon, işlev seviyesi erişim kontrol eksikliği, emniyetsiz doğrudan nesne referansı gibi risklerle ilgili saldırı girişimlerinde bulunduğu görülmektedir. İstekler aynı IP’den ve aynı tarihte arka arkaya yapılmıştır ve durum kodları incelendiğinde başarılı olunamadığı görülmüştür.

```

BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 28888
IPNO: 65.111.180.205
TARİH: 07/Oct/2012
URL: "GET /?param=-1+UNION+SELECT+GROUP_CONCAT(table_name)+FROM+information_schema.tables HTTP/1.1"
DURUM: 302
BOYUT: 22988.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 28893
IPNO: 65.111.180.205
TARİH: 07/Oct/2012
URL: "GET //WEB-INF%20-%20Copy/web.xml HTTP/1.1"
DURUM: 404
BOYUT: 1175.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 28897
IPNO: 65.111.180.205
TARİH: 07/Oct/2012
URL: "GET //WEB-INF/jboss-web.xml_ HTTP/1.1"
DURUM: 404
BOYUT: 1175.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

Plot : Master Plot
Instance: 28902
IPNO: 65.111.180.205
TARİH: 07/Oct/2012
URL: "GET /stats/ HTTP/1.1"
DURUM: 404
BOYUT: 1175.0
BASVURULAN: "-"
TARAYICI: "Mozilla/5.0 (compatible; MSIE 9.0;

```

Şekil 5.31. Önışleme-2 verisi yayılım dağılımından elde edilen detaylı bilgi

Böylece Önışleme-2 verisinde web robotlarının temizlenmesiyle daha etkin sonuç elde edilmiştir.

Yine aynı veriye kümeleme algoritmalarından Kmeans uygulanmıştır. Kümeleme, verideki benzer kayıtların gruplandırılmasını sağlayan veri madenciliği tekniğidir. Kümeleme yönteminde her kayıt var olan kümelerle karşılaştırılmakta ve kendisine en yakın kümeye atanmaktadır. Bu yeni gelen kayıtla kümeyi tanımlayan değer de değişmektedir. Optimum çözüm bulununcaya kadar kayıtlar yeniden atanmakta ve küme merkezleri ayarlanmaktadır [82]. Kmeans; URL, tarih ve durum alanları için uygulanmıştır. Çapraz geçerleme (cross validation) veya yüzde ayırma (percentage split) seçenekleri kullanılarak kurulan modellerin doğruluğu ölçülebilmektedir. Yüzde ayırma bölümünde veri kümesinin belirli bir yüzdesi modelin öğrenilmesi için kalan kısmı ise modelin testi için kullanılmaktadır. Burada programın varsayılan değeri olan %66 kullanılmıştır. Şekil 5.32’de elde edilen sonuç sunulmuştur.

Cluster#	1
0	(4820)
(24400)	
04/Sep/2012	29/Aug/2012
"POST /iphone/restoran_bilgi.php HTTP/1.1" "GET /index.php?option=com_egbcaptcha&width=85&height=35&characters=4 HTTP/1.1"	
200	200

Şekil 5.32. Kmeans kümeleme algoritması kullanılarak yapılan analiz

Burada başarılı (200 durum kodu) 2 istek için küme oluştuğu görülmektedir ve istek yapılan sayfaların da en çok ziyaret edilen sayfalardan oldukları bilinmektedir. 404 hata durum koduyla ilgili Visualize sekmesinde yayılım dağılımında küme oluştuğu halde burada Kmeans kümeleme algoritmasıyla oluşmamıştır. Çünkü web iz bilgilerinde genel olarak, özellikle geniş tarih aralığında çalışma yapıldığı durumlarda, 200 serili durum kodlarının yer aldığı istek sayısına göre 400 serili hata durum kodlarının yer aldığı istek sayısı oldukça azdır. Şekil 5.33’de WEKA önışlem penceresinde durum kodlarının sayıları görülmektedir.

Selected attribute		
Name: DURUM		
Missing: 0 (0%)		
Distinct: 9		
Type: Nominal		
Unique: 1 (0%)		
No.	Label	Count
1	200	38437
2	403	59
3	500	4036
4	302	753
5	303	491
6	404	483
7	301	1
8	206	6
9	400	8

Şekil 5.33. WEKA’da işlenen verinin durum kodları ve sayıları

Dolayısıyla özellikle saldırı tespitine yönelik analiz yapılacağıında 400 serili hata durum kodları haricindeki kayıtların önışleme aşamasında silinmesi gerekmektedir. Ancak WEKA Visualize sekmesi sayesinde veri yayılım dağılımları görüntülenebildiği için analiz edilen bu veride saldırı tespiti yapılabilmiştir.

Apriori algoritması, genel anlamda birliktelik kuralı (association rule) çıkarımında kullanılan bir algoritmadır. Algoritmanın amacı, veri tabanında bulunan satırlar arasındaki

bağlantıyı ortaya çıkarmaktır. Bu algoritmada sık geçen kümeleri tespit etmek üzere veritabanını sürekli taramak gerekmektedir. İlk taramada bir elemanlı minimum destek değerini sağlayan sık geçen öge kümeleri bulunmakta; sonraki taramalarda bir önceki taramada tespit edilen sık geçen öge kümeleri aday kümeler olarak ele alınmakta ve yeni sık geçen öge kümelerini bulmak üzere kullanılmaktadır. Destek değerleri tarama yapıldığı sırada hesaplanmakta ve aday kümelerinden minimum destek değerini sağlayan kümeler o taramada belirlenen sık geçen öge kümelerini oluşturmaktadır. Sık geçen öge kümeleri sonraki taramalar için aday küme olmaktadır. Bu işlemler yeni bir sık geçen öge kümesi tespit edilmediği sürece devam etmektedir [37].

Önişleme-2 verisinde tarih, URL ve durum alanları için minimum destek değeri 0.1 ve minimum güven değeri 0.9 için Apriori algoritmasıyla oluşan birliktelik durumları Şekil 5.34’de sunulmuştur.

```

Associator output

=== Run information ===

Scheme:      weka.associations.Apriori -N 10 -I 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:    QueryResult-weka.filters.AllFilter-weka.filters.unsupervised.attribute.Remove-R1,5-7
Instances:   44274
Attributes:  3
              TARIH
              URL
              DURUM

=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.1 (4427 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 5

Size of set of large itemsets L(2): 4

Best rules found:

1. URL="GET /index.php?option=com_egbcaptcha&width=85&height=35&characters=4 HTTP/1.1" 6991 ==> DURUM=200 6990    conf:(1)
2. URL="GET / HTTP/1.1" 6271 ==> DURUM=200 6183    conf:(0.99)
3. URL="POST /index.php HTTP/1.1" 4905 ==> DURUM=200 4828    conf:(0.98)

```

Şekil 5.34. Apriori algoritması çıktısı

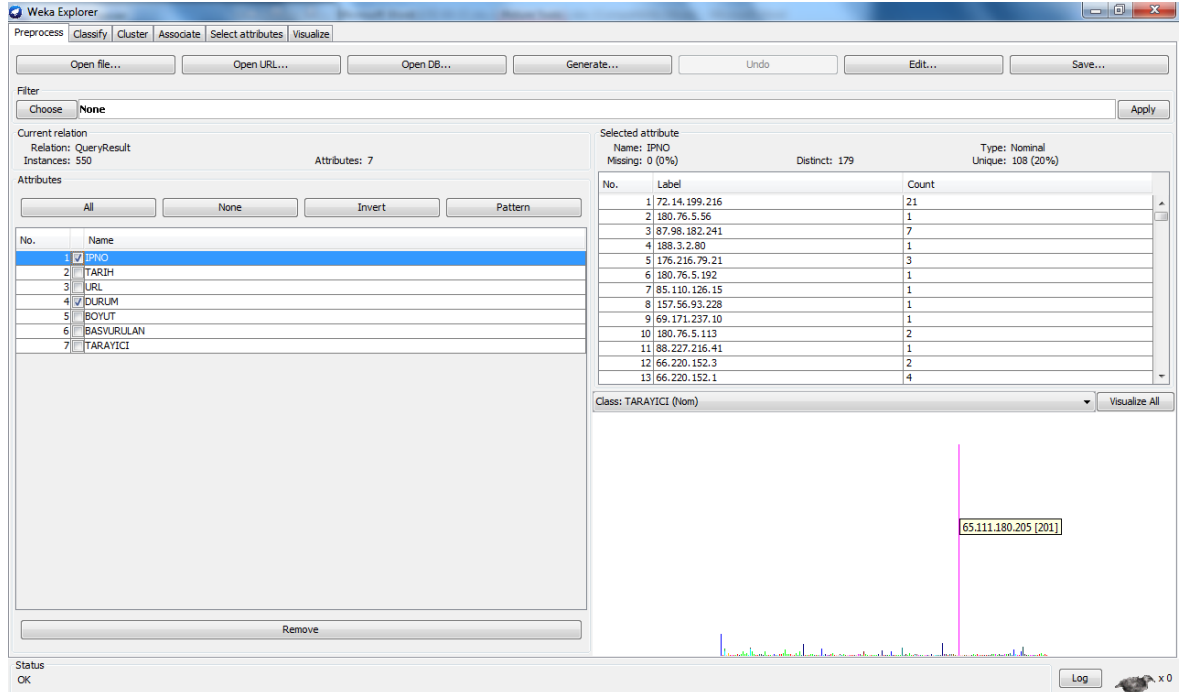
Burada başarılı (200 durum kodu) ve en sık ziyaret edilen sayfalar arasında birliktelik kuralları olduğu görülmüştür. Bu sayfalar anasayfa, yönlendirme sayfası ve “captcha”

olarak isimlendirilen güvenlik resmindeki harflerin girildiği sayfadır. Bu sayfa her rezervasyon işlemi sırasında kullanıldığından Apriori ile kural oluştuğu düşünülmektedir. Kümelemede olduğu gibi burada da 400 serili hata durum kodlarıyla ilgili kural oluşmamıştır.

5.2.3. WEKA ile saldırı tespiti

Bir önceki bölümde WEKA ile istatistiksel analiz yapılırken kümeleme ve birliktelik kurallarına göre herhangi bir saldırı girişimi tespit edilememiştir. Sadece WEKA'nın Visualize sekmesinden saldırı girişiminde bulunulduğu tahmin edilmiştir.

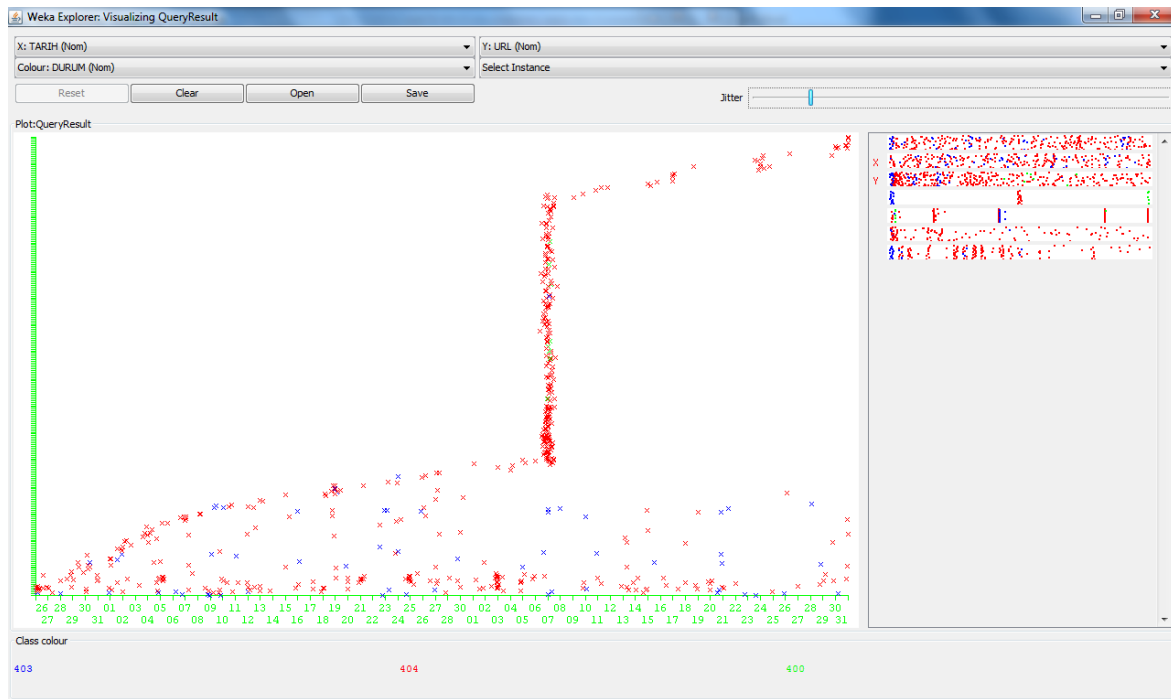
Bu bölümde doğrudan saldırı tespitine yönelik sadece 400 serili durum kodlarının bulunduğu iz bilgisi satırlarını içeren önişleme-3 verisi kullanılarak analiz yapılmıştır. Veri WEKA'ya yüklendikten sonra Şekil 5.35'de gösterilen önişleme sekmesinde, 65.111.180.205 numaralı IP'den çok fazla sayıda hatalı istek alındığı hemen dikkat çekmektedir.



Şekil 5.35. Önişleme-3 verisinin analizi

Şekil 5.36’da visualize sekmesinde X eksenini tarih alanını, Y eksenini URL alanını, ekranın en altında görüldüğü gibi farklı renkler de durum alanını temsil etmektedir. Bu aşamada sadece hata durum kodlarına göre işlem yapıldığı için 400, 403 ve 404 durum kodları grafikte yer almakta ve yoğun olarak 404 ile sayfanın görüntülenemediği görülmektedir.

Burada tarih, URL ve durum alanları için yayılım dağılımı incelendiğinde aynı tarihte çok farklı URL’lere istek yapıldığı ve çok sayıda hata durum kodu alındığı görülmektedir. Bu alandaki verilerin detaylı bilgileri incelendiğinde saldırı girişiminde bulunulduğu açıkça görülmektedir.



Şekil 5.36. Önişleme-3 verisinin visualize ekranından analizi

Önişleme-3 verisine Apriori algoritmasını çalıştırarak minimum destek değeri 0.1 ve minimum güven değeri 0.9 için birliktelik kurallarını uyguladığımızda Şekil 5.37’deki sonuç elde edilmiştir.

```

Attributes: 4
            IPNO
            TARİH
            URL
            DURUM
=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.1 (55 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 5
Size of set of large itemsets L(2): 4
Size of set of large itemsets L(3): 1

Best rules found:

1. IPNO=65.111.180.205 201 ==> TARİH=07/Oct/2012 201   conf:(1)
2. IPNO=65.111.180.205 DURUM=404 192 ==> TARİH=07/Oct/2012 192   conf:(1)
3. URL="GET /undefined HTTP/1.1" 77 ==> DURUM=404 77   conf:(1)
4. TARİH=07/Oct/2012 DURUM=404 193 ==> IPNO=65.111.180.205 192   conf:(0.99)
5. TARİH=07/Oct/2012 207 ==> IPNO=65.111.180.205 201   conf:(0.97)
6. IPNO=65.111.180.205 201 ==> DURUM=404 192   conf:(0.96)
7. IPNO=65.111.180.205 TARİH=07/Oct/2012 201 ==> DURUM=404 192   conf:(0.96)
8. IPNO=65.111.180.205 201 ==> TARİH=07/Oct/2012 DURUM=404 192   conf:(0.96)
9. TARİH=07/Oct/2012 207 ==> DURUM=404 193   conf:(0.93)
10. TARİH=07/Oct/2012 207 ==> IPNO=65.111.180.205 DURUM=404 192   conf:(0.93)

```

Şekil 5.37. Apriori algoritması ile elde edilen sonuçlar

Burada güven değeri 1 için aşağıdaki kurallar oluşmuştur ;

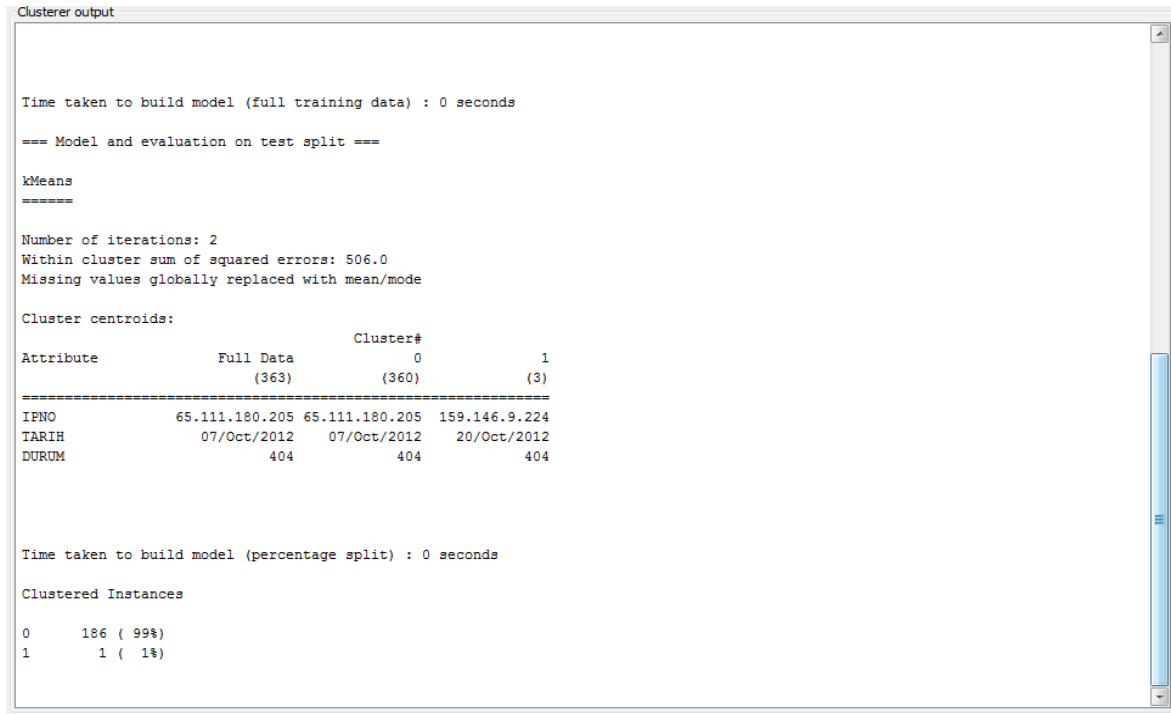
1. IPNO=65.111.180.205 201 ==> TARİH=07/Oct/2012 201 conf:(1)
2. IPNO=65.111.180.205 DURUM=404 192 ==> TARİH=07/Oct/2012 192 conf:(1)

Bu sonuca göre; 65.111.180.205 201 numaralı IP numarasının 07 Ekim 2012 tarihinde sayfayı ziyaret etmiş olduğu ve ilk satırda 201 kural oluşurken ikinci satırda durum kodu 404 olan istekler için 192 satır oluştuğu görülmektedir. Yani aynı IP'den gönderilen isteklerin hepsi ilk kez 07 Ekim 2012 tarihinde gönderilmiş ve neredeyse isteklerin tamamında (%96'sında) 404 ile sayfa görüntülenememiştir.

Tarih, URL, IP ve durum arasında ikili, üçlü diğer oluşan kurallarla da aynı sonuca varılmaktadır.

Tarih, IP numarası ve durum alanları için Kmeans kümeleme algoritması çalıştırılmış ve iki küme oluştuğu görülmüştür. Yüzde ayırma bölümünde programın varsayılan değeri olan %66 kullanılmıştır. Şekil 5.38'de elde edilen sonuç sunulmuştur. Kümelere biri

daha önce saldırı girişiminde bulunduğunu tespit ettiğimiz IP numarasını içermektedir. Diğer küme ise çok az sayıda veri içermektedir.



```
Clusterer output

Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on test split ===

KMeans
=====

Number of iterations: 2
Within cluster sum of squared errors: 506.0
Missing values globally replaced with mean/mode

Cluster centroids:

Attribute          Full Data          Cluster#
                   (363)              0              1
                   (360)              (3)
-----
IPNO                65.111.180.205 65.111.180.205 159.146.9.224
TARİH               07/Oct/2012   07/Oct/2012   20/Oct/2012
DURUM               404           404           404

Time taken to build model (percentage split) : 0 seconds

Clustered Instances

0      186 ( 99%)
1       1 ( 1%)
```

Şekil 5.38. Kmeans ile elde edilen kümeler

Bu bölümde WEKA ile saldırı girişiminde bulunduğu kesin olarak tespit edilmiştir. Her ne kadar saldırı girişimiyle ilgili detaylı bilgiler görülebilse de saldırıların türü çok çeşitli olabilmektedir. Bu analizlerden elde edilen saldırı girişiminin tespit edildiği tarih, durum kodu, URL gibi bilgiler web iz bilgileri saldırı analiz aracı olan Apache Scalp ile saldırı türünü ve saldırının başarılı olup olmadığını tespit etmek üzere kullanılmıştır.

5.2.4. Apache Scalp ile iz bilgisi analizi

Scalp, güvenlik uzmanlarının kullandığı, Apache iz bilgisi analiz aracıdır. Scalp, Apache sunucunun güvenliğini iz bilgilerini analiz ederek değerlendiren ve bunu gerçekleştirmek üzere düzenli ifadeler (regular expressions) kullanan küçük bir python scripttir. Web uygulamalarının SQL Enjeksiyon, XSS, CSRF vb. bilinen saldırıların hedefinde olup olmadığını analiz etmektedir [83]. Bu script PHPIDS düzenli ifadelerini kullanmaktadır; bu ifadeler filtreleme seçeneğini sağlayan “default_filter.xml” dosyasında yer almaktadır [84].

PHPIDS, açık kaynak PHP web uygulamaları saldırı tespit sistemidir. Her saldırı için bir etki değeri belirlemektedir. Saldırı tespiti için düzenli ifadeleri kullanmaktadır. PHPIDS karaliste yaklaşımını uygulamakta ve var olduğu bilinen saldırı çeşitleri için kurallar içermektedir. Dolayısıyla yeni saldırı vektörleri için PHPIDS kullanan uygulamaların güncellemelerini takip etmesi gerekmektedir [86].

Düzenli ifadeleri içeren “default_filter.xml” dosyasıyla birlikte scalp phyton 0.4 versiyonu olan “scalp-0.4.py” dosyası Apache Scalp’ı çalıştırmak için gerekmektedir [85].

Literatürdeki çalışmalar incelendiğinde Scalp aracıyla ilgili çalışmaların çok az olduğu ve iz bilgisi dosyaları analizinde genellikle SQL Enjeksiyon ataklarını tespit etmek üzere kullanıldığı görülmüştür [87-89].

Bu tez çalışmasında Scalp’ın tercih edilme sebebi Apache web iz bilgilerinin analiz ediliyor olması ve Scalp’ın doğrudan Apache aracı olup web iz bilgileri saldırı tespitine yönelik geliştirilmiş bir araç olmasıdır.

Elia, Fonseca ve Vieira (2010) yaptıkları çalışmada içinde Scalp’ın da bulunduğu birçok saldırı tespit aracını incelemişlerdir. Birçok saldırı analiz aracında olduğu gibi Scalp’ın da yanlış alarm verebildiğini belirtmişlerdir. Yanlış alarmların azaltılması ve analizcinin daha etkin sonuç alabilmesi için Scalp ile analiz edilecek verinin öncelikle hedefe yönelik filtrelenmesi gerekmektedir. Bu filtreleme WEKA’dan elde edilen analiz sonucuna göre gerçekleştirilmiştir [90].

Apache Scalp’ı çalıştırmak ve çeşitli Linux komutlarıyla da analiz yapabilmek için bir sanallaştırma yazılımı olan Vmware Workstation 9.0. kurularak üzerine Linux tabanlı bir güvenlik kontrol işletim sistemi olan BackTrack 5-r3 kurulmuştur.

Bu aşamada Scalp’da önışleme-2 verisi analiz edilecektir. WEKA’dan web madenciliği işlemleri sonucunda elde edilen analiz sonuçları Scalp’da filtreleme yapmaya yardımcı olacaktır. Son işlenen veri olan önışleme-3 verisi yerine önışleme-2 verisinin kullanılma sebebi; saldırı girişimlerinin başarılı olup olmadığını tespit etmek içindir. Çünkü 400 serili durum kodlarının yer aldığı önışleme-3 verisinde saldırı girişimleri tespit edilebilmekte

ancak saldırının başarılı olup olmadığını anlamak için 200 serili durum kodlarının yer aldığı işlenmiş olan önışleme-2 verisinin kullanılması gerekmektedir.

Önışleme-2 verisini analiz etmek üzere, Backtrack kullanılarak Şekil 5.39'da gösterilen script çalıştırılmıştır.

```

Processed 64348 lines over 64426
Found 10691 attack patterns in 75.682437 s
Generating output in ./access_log_scalp_*
root@bt:~/Desktop# python scalp-0.4.py -l access_log -f default_filter.xml -o ./ -h
Loading XML file 'default_filter.xml'...
Processing the file 'access_log'...
Scalp results:
Processed 64348 lines over 64426
Found 10691 attack patterns in 71.034453 s
Generating output in ./access_log_scalp_*
root@bt:~/Desktop#

```

Şekil 5.39. Scalp script çalıştırma ve sonuçları

Toplam 64348 satır için işlem yapılmıştır ve 10691 saldırı girişimi 71 saniye içerisinde tespit edilmiştir.

Kullanılan scriptte;

- l: iz bilgisi dosyasını belirtmektedir.
- f: filtreleme dosyasını belirtmektedir.
- o: çıktı dizinini göstermektedir.
- h: çıktı formatı HTML'dir.

Scalp varsayılan olarak çıktıyı iz bilgisi dosyasıyla aynı dizine koymaktadır. Varsayılan olarak çıktı text dosyasıdır ve XML olarak da çıktı üretebilmektedir. Bu işlemde kullanılan filtreleme dosyasından bir kesit aşağıdaki gibidir:

<filter>

<id>17</id>

```

<rule><![CDATA[([^\s\w,.-\/?+]\s*)?(?<![a-z]\s)(?<![a-z\_\@-
])\s*return\s*)?(?:hash|name|href|source|pathname|close|constructor|port|protocol|replace|b
ack|forward|go|document|window|self|parent|frames|_?content|date|cookie|innerHTML|innert
ext|outerhtml|print|moveby|resizeto)(?(1)[^\w%"])(?:\s*["^\s\w%",".+-]))]]>
</rule>
<description>Detects JavaScript object properties and methods</description>
<tags>
<tag>xss</tag>
<tag>csrf</tag>
<tag>id</tag>
<tag>rfe</tag>
</tags>
<impact>4</impact>
</filter>

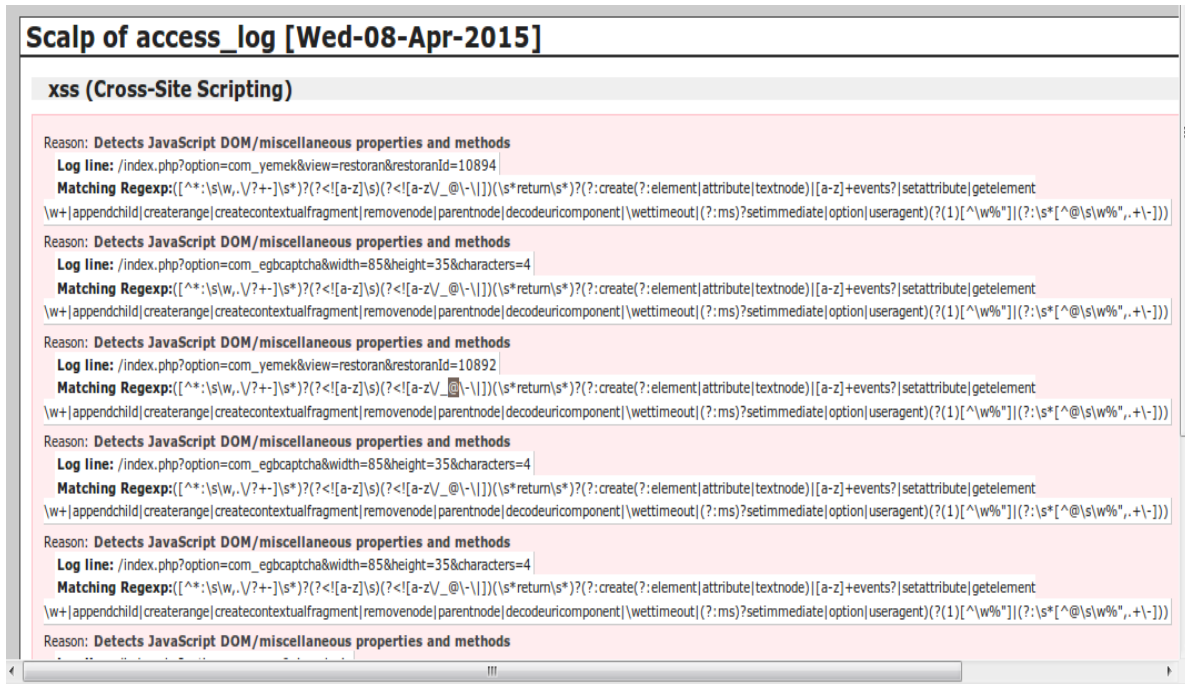
```

Rule: düzenli ifadeleri vermektedir.

Description: saldırı örüntüsü ile ilgili açıklama yapılmaktadır.

Tags: yapılacak testin sınırlandırılmasını sağlamaktadır. Örneğin sadece XSS ve CSRF saldırıları ile ilgileniliyorsa yapılan test -a seçeneği ile (-a XSS, CSRF) şeklinde sınırlandırılabilir.

Script çalıştırıldıktan sonra sonuçların yer aldığı “access_log_scalp” adlı dosya üretilmiştir. Bu dosyada saldırı olabilecek örüntü ile ilgili açıklama, tespit edilen ilgili iz bilgisi satırı ve “default_filter.xml” dosyasındaki düzenli ifade ile eşleşen satır gösterilmektedir. Dosyanın içeriğinden bir kesit Şekil 5.40’da sunulmuştur:



Şekil 5.40. Sonuç dosyasından bir kesit

Bir sonraki adımda önce sadece XSS saldırıları için script çalıştırılmıştır. Bu durumda 28 saniye içerisinde 10683 saldırı girişimi tespit edilmiştir. Sonrasında sadece SQL Enjeksiyon için script çalıştırılmıştır. Bu durumda ise 19 saniye içerisinde 2 saldırı girişimi tespit edilmiştir. Şekil 5.41’de çalıştırılan scriptler ve sonuçları görülmektedir.



Şekil 5.41. XSS ve SQL Enjeksiyon için çalıştırılan scriptler

Scalp ile üretilen sonuçların yaklaşık %15’i yanlış alarm (false positive) olmaktadır. Dolayısıyla sonuçları analiz eden kişinin dikkatli bir incelemeden geçirmesi gerekmektedir. Bu ise veri boyutunun çok büyük olduğu durumlarda 100.000 lerce satır

arasından saldırı tespiti yapmayı gerektirmektedir. Bu noktada bir önceki bölümde web madenciliği ile analiz edilmiş olan veri sonuçlarına göre filtreleme yapılarak Scalp'da gerçekleştirilecek saldırı analizinde veri boyutu ciddi oranda azaltılıp; analizi çok daha kısa sürede tamamlamak hedeflenmiştir.

WEKA'da gerçekleştirilen örüntü keşfinde hem visualize hem de kümeleme ve birliktelik kurallarıyla saldırı girişimi olduğu tespit edilen isteklerin tarihi 7 Ekim 2012'dir. Bununla birlikte IPNO ve çeşitli istek yapılan URL lerle ilgili de detaylı bilgi bulunmaktadır. Scalp'da 7 Ekim 2012 tarihine göre filtreleme yapılarak analiz tekrar yapılmıştır ve Şekil 5.42'deki sonuçlar elde edilmiştir. Toplam 1211 satır işlenmiştir ve 7 saniye içerisinde 166 saldırı girişimi tespit edilmiştir.



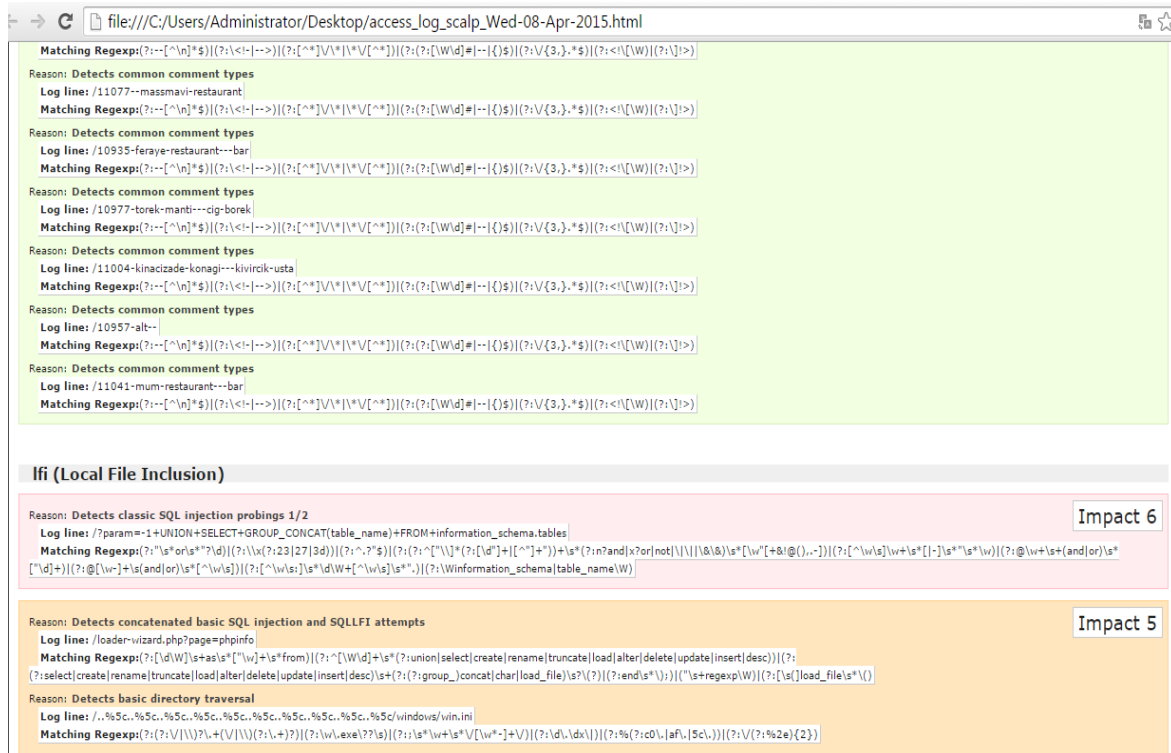
```

Generating output in ../access_log_scalp *
root@bt:~/Desktop# python scalp-0.4.py -l access_log -f default_filter.xml -o ./ -h -p 07/0ct/2012.07/0ct/2012
Loading XML file 'default_filter.xml'...
Processing the file 'access_log'...
Scalp results:
  Processed 1211 lines over 64426
  Found 166 attack patterns in 6.993619 s
Generating output in ../access_log_scalp *
root@bt:~/Desktop#

```

Şekil 5.42. Tarihe göre filtrelenerek çalıştırılan script

Tespit edilen saldırılarla ilgili bir kesit Şekil 5.43'de sunulmuştur. Örneğin Sunucudan Dosya Çağırma (Local File Inclusion) başlığı altında SQL Enjeksiyon tespit edildiği görülmektedir.



Şekil 5.43. Filtreleme sonrası Scalp'dan elde edilen saldırı sonuçları kesiti

Scalp'ın ürettiği HTML çıktısında saldırı türü ana başlığı altında saldırının nedeni, etki değeri, tespit edildiği iz bilgisi satırı ve tespiti sağlayan düzenli ifadelerle ilgili bilgiler yer almaktadır. Ancak saldırı girişiminin tespit edildiği iz bilgisi satırındaki durum kodu, IP numarası ve tarih bilgileri görüntülenmemektedir. Scalp'ın açık kaynak olmasından yararlanarak bu tez çalışmasına özel, kaynak kodunda çeşitli değişiklikler yapılmıştır. Özellikle saldırı girişimini incelerken durum kodunun bilinmesi saldırının başarılı olup olmadığını gösterdiği için çok önemlidir. Yine saldırı girişiminde bulunan IP numaraları tespit edilerek bu IP ler bloklanabilir. Scalp'ın çıktısına bu alanlarla ilgili bilgiler de eklendikten sonra sonuçlar daha detaylı analiz edilmiştir.

Scalp kaynak koduna yapılan eklemeler sonucunda elde edilen rapor sonucundan bir kesit Şekil 5.44’de sunulmuştur.

Reason: Detects JavaScript DOM/manipulation properties and methods

Log Kayiti: /index.php?option=com_egbcaptcha&width=85&height=35&characters=4
IP NO:92.45.145.122 - **Tarih:**07/Oct/2012:23:29:44

Durum Kodu:400

Esilesen Regexp:([^*\\$\\w_\./+?<|([a-z])(<|[a-z_@|\-\/])(\\$*return*)?(?:create(?:element|attribute|textnode)|([a-z]+events)?setattribute|getelementbyid|appendchild|createrange|createcontextualfragment|removenode|parentNode|decodeURIComponent|setTimeout(?:ms)?setImmediate|option|useragent)(?{1}[\^\\w%"])(?:\\$*[^\\$\\w%","+\-\]])

Reason: Detects nullbytes and other dangerous characters

Log Kayiti: /horde/util/barcode.php?type=../..../..../etc/passwd%00
IP NO:65.111.180.205 - **Tarih:**07/Oct/2012:23:13:05

Durum Kodu:404

Esilesen Regexp:(?:\\x[01fe][\db-ce-fj])(?:%[01fe][\db-ce-fj])(?:&#[01fe][\db-ce-fj])(?:\\[01fe][\db-ce-fj])(?:&#x[01fe][\db-ce-fj])

Reason: Detects self-executing JavaScript functions

Log Kayiti: /roller-ui/login rol?pageTitle=\${new%20java.lang.Integer(99035%2b99156)}
IP NO:65.111.180.205 - **Tarih:**07/Oct/2012:23:13:05

Durum Kodu:404

Esilesen Regexp:(?:\\w\\w*s*)(\\s*)(?:\\((\\w[s+]|(\\w[s+]|\\w[s+]))(?:<(?!:(mozilla|d\\.d\\.)))([^)]+[\\^[^]]+|^)*\\))(?:[^s!]{([^(|([[(|(^)))})+(}|)\\)})|\\s+\\\\d*}\\}))

s*["';:] + s*[{([(| (^))) } +(}| \\)}) | \\s + \\\\ d *] \\}) :

Reason: Detects nullbytes and other dangerous characters

Log Kayiti: /util/barcode.php?type=../..../..../etc/passwd%00
IP NO:65.111.180.205 - **Tarih:**07/Oct/2012:23:13:10

Durum Kodu:404

Esilesen Regexp:(?:\\x[01fe][\db-ce-fj])(?:%[01fe][\db-ce-fj])(?:&#[01fe][\db-ce-fj])(?:\\[01fe][\db-ce-fj])(?:&#x[01fe][\db-ce-fj])

Reason: Detects self-executing JavaScript functions

Log Kayiti: /roller-ui/login rol?pageTitle=\${new%20java.lang.Integer(99035%2b99156)}
IP NO:65.111.180.205 - **Tarih:**07/Oct/2012:23:13:05

Durum Kodu:404

Esilesen Regexp:(?:\\w\\w*s*)(\\s*)(?:\\((\\w[s+]|(\\w[s+]|\\w[s+]))(?:<(?!:(mozilla|d\\.d\\.)))([^)]+[\\^[^]]+|^)*\\))(?:[^s!]{([^(|([[(|(^)))})+(}|)\\)})|\\s+\\\\d*}\\}))

s*["';:] + s*[{([(| (^))) } +(}| \\)}) | \\s + \\\\ d *] \\}) :

Reason: Detects possible includes, VBScript/JScript encoded and packed functions

Log Kayiti: /servlet/%OARefresh:0;URL=javascript:prompt(1)%OA1
IP NO:65.111.180.205 - **Tarih:**07/Oct/2012:23:13:10

Durum Kodu:404

Esilesen Regexp:(?:#@~\\^\\w+)((?:\\wscript:[@import["\\w"];base64|base64,)|(?:\\w*s*(\\w[s+],[\\w[s+],[\\w[s+],[\\w[s+],[\\w[s+],[\\w[s+],[\\w[s+]]]

Şekil 5.44. Düzenlenmiş Scalp raporu çıktısı

Çizelge 5.7’de Scalp analiz sonuçlarının karşılaştırılması yapılmıştır.

Çizelge 5.7. Scalp analiz sonuçlarının karşılaştırılması

Apache Scalp ile Gerçekleştirilen Analiz Sonuçları		
	Önişleme-2 verisi analiz sonuçları	Örüntü keşfi sonucuna göre filtrelenen veri analizi
İşlenen Satır Sayısı	64348	1211
Veri İşleme Süresi	71 saniye	7 saniye
Saldırı Girişimi Olabilecek Satır Sayısı	10691	166

İşlem süresi %90.1 oranında kısılırken, incelenecek saldırı girişimleri sayısı da %88.7 oranında azalmıştır.

Apache Scalp ile tespit edilen saldırı girişimleri ile ilgili bilgiler Çizelge 5.8’de sunulmuştur.

Çizelge 5.8. Apache Scalp ile tespit edilen saldırı girişimleri

Apache Scalp ile Tespit Edilen Saldırı Girişimleri		
Saldırı Türleri	Saldırı Sayısı	Etki Değerleri
XSS	163	3,5,6
SQL Enjeksiyon	2	5,6
Sunucudan Dosya Çağırma (LFI)	10	5,6

Bazı XSS ve SQL Enjeksiyon saldırıları “Sunucudan Dosya Çağırma” saldırıları başlığı altında da raporlanmıştır.

6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, web madenciliği, web iz bilgilerinin yapısı ve içeriği ile ilgili detaylı bilgi verilmiş, güvenlik açısından gerçekleştirilecek analizi desteklemek üzere web saldırıları hakkında ayrıntılı bilgiler verilerek uygulamalı örnekler sunulmuştur.

Tezin uygulama bölümünde ise analiz hem istatistiksel hem de güvenlik açısından gerçekleştirilmiştir. Önişleme aşamasında web kullanım madenciliği ve web saldırı tespiti için yapılan işlemlerin farkları ortaya konulmuştur. Her bir amaca yönelik ayrı önişlemeler gerçekleştirilerek analizler yapılmıştır. Adım adım çeşitli amaçlara yönelik olarak gerçekleştirilen önişlemeler ile web madenciliği işleminin kalitesi artırılmış, madencilik işlemi kolaylaştırılmış ve verimlilik artmıştır.

6.1. İstatistiksel Sonuçların Değerlendirilmesi

Gerçekleştirilen istatistiksel analizler neticesinde kullanıcı eğilimleri ve genel site kullanımı hakkında elde edilen bulgularla site geliştiricilerine web sitesinin iyileştirilmesine ve geliştirilmesine yönelik katkıda bulunulacağı düşünülmektedir.

Bu tez çalışmasında Nihuo ve Sawmill programları ile gerçekleştirilen istatistiksel analiz için önişleme aşamasında web robotlarının istekleri korunmuştur. Böylece web robotları ile ilgili de çıkarımlarda bulunulmuştur. WEKA’da gerçekleştirilen analizlerde hem web robotlarıyla hem de web robotları olmadan analiz gerçekleştirilerek ziyaretçi sayısı, başarılı istek sayısı, hatalı istek sayısı gibi sonuçlar elde edilmiştir. Kümeleme ve birliktelik kuralları ile belirli tarihlerde durum kodlarına göre en çok ziyaret edilen sayfalar belirlenmiştir.

Elde edilen başlıca istatistiksel sonuçlar ve öneriler aşağıda sunulmuştur:

1. Genel olarak sayfalara yapılan ziyaretler, sayfa görüntüleme ve bant genişliği hakkında istatistiksel bilgiler verilmiştir. Sayfaya yapılan ziyaretlerde web robotlarının gönderdiği isteklerle kullanıcı isteklerinin birbirine çok yakın sayıda olduğu

görülmüştür. Özellikle kullanıcılarla ilgili yapılacak analizlerde yanıltıcı olmaması için web robotlarının kayıtlarının temizlenmesi gerektiği sonucuna varılmıştır.

2. Aktivite istatistiklerine göre haftanın en yoğun günü Pazar günü olmuştur. Özellikle reklam, tanıtım vb. faaliyetler haftanın bu gününde yoğun olarak yapılabilir. Yoğunluğun en az olduğu gün ise Perşembe günüdür. Perşembe günleri web sitesi ile ilgili bakım, yenileme çalışmaları yapılması uygun olacaktır.
3. En çok kullanılan işletim sistemleri, tarayıcılar, arama robotları, erişimin en çok sağlandığı araç tipleri ve en çok kullanılan mobil cihazlar belirlenmiştir. Hem WEKA'dan hem de web iz bilgisi analiz araçlarından elde edilen sonuçlarda mobil cihazlarla ciddi oranda erişim olduğu görülmüştür. Mobil cihazların kullanımının gün geçtikçe arttığı da dikkate alınarak web geliştiriciler bu mobil teknolojilerle uyumlu sayfa yapısı üzerine çalışabilirler.
4. Günün saatlerine göre erişim analizi incelendiğinde özellikle öğleden sonraları artan bir erişim grafiği görülmektedir. Web sayfalarının bakımıyla ilgili çalışmalar erişimin en az sağlandığı saat aralığı olan 04:00 ve 08:00 arasında yapılabilir.
5. Erişimin en çok sağlandığı hem ülke hem de şehir bazında Amerika Birleşik Devletlerinin büyük bir orana sahip olduğu görülmektedir. Bu durumun arama motorlarından kaynaklandığı düşünülmektedir.

6.2. Güvenlik Açısından Sonuçların Değerlendirilmesi

Elde edilen sonuçlar ve öneriler aşağıda sunulmuştur:

1. Web robotları sayfaya kısa süre içerisinde çok fazla ve arka arkaya istek gönderdiğinden web robotlarının da yer aldığı önışleme verisi için analiz gerçekleştirildiğinde hata serili durum kodlarıyla visualize ekranında kümeler oluştuğu görülmüştür. Toplam 6 kümeden 5'i web robotlarına ait iken sadece 1 kümenin gerçek saldırı girişimi olduğu tespit edilmiştir. İkinci önışleme verisinden web robotları çıkarıldıktan sonra sadece tek bir küme oluştuğu görülmüştür. Özellikle web

madenciliği ile saldırı analizi yapılacağında web robotlarının önışleme aşamasında belirlenip indekslediği sayfaların temizlenmesi gerektiği sonucuna varılmıştır.

2. Web robotlarının temizlenmiş olduđu önışleme-2 verisi web madenciliği ile analiz edildiğinde 200 (başarılı) durum koduyla en çok ziyaret edilen sayfalar tespit edilmiştir. Web iz bilgilerinde geniş tarih aralığında işlem yapıldığında çođu zaman işlemin başarılı olduğuna dair durum kodu alınacağından visualize ekranında oluşan saldırı örüntüleri kümeleme veya birliktelik kuralları ile görülememiştir. Bu nedenle web madenciliğinde özellikle saldırı tespitine yönelik analiz gerçekleştirileceğinde 400 serili hata durum kodlarının haricindeki iz bilgisi satırlarının temizlenmesi gerekmektedir. Bu şekilde elde edilen önışlem-3 verisine web madenciliği algoritmaları uygulandığında doğrudan saldırı tespiti yapılmıştır.
3. Web madenciliği ile elde edilen saldırı örüntüleri ile ilgili bilgiler, Apache Scalp aracında filtreleme yapmak üzere kullanılmış ve saldırıların türü ve yeri ile ilgili detaylı analizler yapılmıştır. İşlem süresi %90.1 oranında kısılırken, incelenecek saldırı girişimleri sayısı da %88.7 oranında azalmıştır.
4. Scalp'ın sonuçlarına göre en çok XSS ve SQLI atakları yapıldığı görülmüştür. Saldırganlar gizlenebilmek amacıyla çeşitli kodlama (encoding) yöntemlerini kullanarak aranacak kelime gruplarının farklı şekillerde iz bilgisi dosyasında tutulmasını sağlayabilmektedir. Site geliştiricilerin bu atakların bertaraf edilmesine yönelik kaynak kodunda gerekli kontrolleri yapmaları ve varsa saldırı tespit sisteminde gerekli konfigürasyonları uygulamaları gerekmektedir. Saldırı girişimleri daha çok bilinen açıklıklar üzerinden gerçekleştirilmektedir, dolayısıyla yazılımların güncel tutulması, yamaların yapılması da oldukça önemlidir.
5. Web saldırıları bölümünde bahsedilen OWASP projesi olan SQL Inject Me ve ZAP gibi zafiyet tespit etmeye yarayan araçlarla düzenli periyotlarla web sitelerinin taranması gerekmektedir.

KAYNAKLAR

1. Adamov, A. (2014). Data mining and analysis in depth. case study of Qafqaz University HTTP server log analysis. *In Application of Information and Communication Technologies (AICT), 2014 IEEE 8th International Conference on*, 1-4.
2. Özdemir, S. (2012). Veri Madenciliği Ders Notları, *Gazi Üniversitesi*.
3. Dener, M., Dörterler, M., Orman, A. (2009). Açık Kaynak Kodlu Veri Madenciliği Programları: Weka'da Örnek Uygulama. *Akademik Bilişim*, 9, 11-13.
4. Yumnam, A. S., Chaitanya Sreeram, Y., Naeem, S. A. (2014). Overview: Web log mining, privacy issues and application of Web Log mining. *In Computing for Sustainable Global Development (INDIACom), 2014 International Conference on*, 638-641.
5. Khairkar, A. D., Kshirsagar, D. D., Kumar, S. (2013). Ontology for Detection of Web Attacks. *In Communication Systems and Network Technologies (CSNT), 2013 International Conference on*, 612-615.
6. Mobasher, B., Cooley, R., Srivastava, J. (2000). Automatic Personalization based on Web Usage Mining, *Communications of the ACM*, 43(8),142-151.
7. Vahaplar, A., İnceoğlu, M. M. (2001). Veri Madenciliği ve Elektronik Ticaret, *Türkiye'de İnternet Konferansları VII*.
8. Daş, R. (2008). Web Kullanıcı Erişim Kütüklerinden Bilgi Çıkarımı, Doktora Tezi, *Fırat Üniversitesi Fen Bilimleri Enstitüsü*, Elazığ.
9. Leite, J. P. (2011). Analysis of log files as a security aid. *In Information Systems and Technologies (CISTI), IEEE 2011 6th Iberian Conference on*, 1-6.
10. Nagappan, M., Vouk, M. A. (2010). Abstracting log lines to log event types for mining software system logs. *In Mining Software Repositories (MSR), 2010 7th IEEE Working Conference on*, 114-117.
11. Burçak, K.C. (2012). Kırıkkale Üniversitesi Web Sitesinin Kullanıcı Örüntülerinin Web Madenciliği ile Analizi, Yüksek Lisans Tezi, *Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü*, Kırıkkale.
12. Vernekar, S. S., Buchade, A. (2013). MapReduce based log file analysis for system threats and problem identification. *In Advance Computing Conference (IACC), 2013 IEEE 3rd International*, 831-835.
13. Ozulku, O., Fadhel, N. F., Argles, D., Wills, G. B. (2014). Anomaly detection system: Towards a framework for enterprise log management of security services. *In Internet Security (WorldCIS), 2014 World Congress on*, 97-102.

14. Haberal, İ.(2007). Veri Madenciliği Algoritmaları Kullanılarak Web Günlük Erişimlerinin Analizi, Yüksek Lisans Tezi, *Başkent Üniversitesi Fen Bilimleri Enstitüsü*, Ankara.
15. Hussain, T., Asghar, S., Masood, N. (2010). Web usage mining: A survey on preprocessing of web log file. *In International Conference on Information and Emerging Technologies (ICIET)*, 1-6.
16. Özakar, B., Püskülcü, H. (2002). Web içerik ve web kullanım madenciliği tekniklerinin entegrasyonu ile oluşmuş bir veri tabanından nasıl yararlanılabilir.
17. Sisodia, D. S., Verma, S. (2012). Web usage pattern analysis through web logs: A review. *In Computer Science and Software Engineering (JCSSE), 2012 International Joint Conference on*, 49-53.
18. Yadav, M. P., Keserwani, P. K., Samaddar, S. G. (2012). An efficient web mining algorithm for Web Log analysis: E-Web Miner. *(In Recent Advances in Information Technology (RAIT), 2012 1st International Conference on*, 607-613.
19. Yevale, R., Dhage, M., Nalawade, T., Kaule, T. (2014). Unauthorized Terror Attack Tracking Using Web Usage Mining. *(IJCSIT) International Journal of Computer Science and Information Technologies*, 5 (2), 1210-1212.
20. Gržinic, T., Kišasondi, T., Šaban, J., (2013). Detecting anomalous Web server usage through mining access logs, *Central European Conference on Information and Intelligent Systems*, 228-296.
21. Turan, M. (2011). Web Mining: Pattern Discovery On The World Wide Web, Yüksek Lisans Tezi, *Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü*, İzmir.
22. Sarıman, G. (2011). Paralel Programlama ile Web Madenciliğinde Log Analizi, Yüksek Lisans Tezi, *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü*, Isparta.
23. İnternet: Önal, H. Web Sunucu Loglarından Saldırı Analizi. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fblog.bga.com.tr%2F2013%2F01%2Fweb-sunucu-loglardan-saldr-analizi.html>. Son Erişim Tarihi: 20.11.2014.
24. Vigna, G., Robertson, W., Kher, V., Kemmerer, R. A. (2003). A stateful intrusion detection system for world-wide web servers. *In Computer Security Applications Conference, 2003. Proceedings. 19th Annual*, 34-43.
25. Auxilia, M., Tamilselvan, D. (2010). Anomaly detection using negative security model in web application. *In Computer Information Systems and Industrial Management Applications (CISIM), 2010 International Conference on*, 481-486.
26. Sun, Z., Sheng, H., Wei, M., Yang, J., Zhang, H., Wang, L. (2011). Application of web log mining in local area network security. *In Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, 8, 3897-3900.
27. Yıldız, E. (2010). Veri Madenciliği Teknikleriyle Saldırı Tespiti ve Bir Uygulama, Yüksek Lisans Tezi, *Gazi Üniversitesi Bilişim Enstitüsü*, Ankara.

28. Salama, S. E., Marie, M. I., El-Fangary, L. M., Helmy, Y. K. (2011). Web Server Logs Preprocessing for Web Intrusion Detection. *Computer and Information Science*, 4(4), 123.
29. Patil, P. V., Patil, D. R.(2012) Preprocessing Web Logs For Web Intrusion Detection, *International Journal of Applied Information Systems (IJAIS)*.
30. Mabzool, M., Lighvan, M. Z. (2014). INTRUSION DETECTION SYSTEM BASED ON WEB USAGE MINING. *International Journal of Computer Science, Engineering & Applications*, 4(1).
31. Etzioni, O. (1996). The World-Wide Web: quagmire or gold mine?. *Communications of the ACM*, 39(11), 65-68.
32. İnternet: Web Characterization Terminology & Defininition Sheet. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.w3.org%2F1999%2F05%2FWCA-terms%2F>. Son Erişim Tarihi: 24.04.2014.
33. Gezer, M., Erol, Ç., Gülseçen, S. (2007). Bir Web Sayfasının Veri Madenciliği İle Analizi. *Akademik Bilişim*.
34. Duan, J., Liu, S. (2012). Research on web log mining analysis. *In Instrumentation & Measurement, Sensor Network and Automation (IMSNA), 2012 International Symposium on*, 2, 515-519.
35. Daş, R.(2008). Bir web sitesine ait kullanıcı erişim kayıtlarının web kullanım madenciliği yöntemiyle analizi: Fırat Üniversitesi Örneği. *NWSA: Engineering Sciences*, 3(2).
36. Hernandez, P., Garrigos, I., Mazón, J. (2010). modeling web logs to enhance the analysis of Web usage data. *In Database and Expert Systems Applications (DEXA), IEEE 2010 Workshop on*, 297-301.
37. Karabatak, M., İnce, M. C. (2004). Apriori Algoritması ile Öğrenci Başarısı Analizi. *Elektrik Elektronik Bilgisayar Mühendisliği Sempozyumu*, 8-12.
38. Meyer, R. (2008). Detecting Attacks on Web Applications from Log Files. *SANS Institute InfoSec Reading Room*.
39. İnternet: OWASP Top Ten Project, URL: http://www.webcitation.org/query?url=https%3A%2F%2Fwww.owasp.org%2Findex.php%2FCategory%3AOWASP_Top_Ten_Project. Son Erişim Tarihi: 08.12.2014.
40. İnternet: OWASP Top Ten Project. URL: http://www.webcitation.org/query?url=https%3A%2F%2Fwww.owasp.org%2Findex.php%2FTop_10_2013-Top_10. Son Erişim Tarihi: 09.12.2014.
41. İnternet: Common Vulnerabilities and Exposures (The Standard for Information Security Vulnerability Names). URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fcwe.mitre.org%2F>. Son Erişim Tarihi: 08.12.2014.

42. İnternet: Zlatkovski, N. Š. D., and Mileva, A. (2013). Evaluation And Testing Of Several Free/Open Source Web Vulnerability Scanners. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fciit.finki.ukim.mk%2Fdata%2Fpapers%2F10CiiT%2F10CiiT-48.pdf>. Son Erişim Tarihi: 21.12.2014.
43. İnternet: Web Application Security Consortium, “Web Application Security Scanner Threat Classification”. URL: http://www.webcitation.org/query?url=http%3A%2F%2Fprojects.webappsec.org%2F%2FWASC-TC-v2_0.pdf. Son Erişim Tarihi: 21.12.2014.
44. Johari, R. and Sharma, P. (2012). A survey on web application vulnerabilities (SQLIA, XSS) exploitation and security engine for SQL Enjeksiyon. *Communication Systems and Network Technologies (CSNT), 2012 International Conference on*, 453-458.
45. İnternet: Infosec Institute. OWASP Top 10 Tools and Tactics. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fresources.infosecinstitute.com%2Fowasp-top-10-tools-and-tactics%2F>. Son Erişim Tarihi: 08.12.2014.
46. Win, W., Htun, H. H. (2013). A Simple and Efficient Framework for Detection of SQL Enjeksiyon Attack. *IJCER*, 1(2), 26-30.
47. Mali, Y. M., JV, R. M., Raj, M., Gaykar, A. T. (2014). Honeypot: a tool to track hackers. *IRACST – Engineering Science and Technology: An International Journal (ESTIJ)*,(4), 2250-3498.
48. İnternet: URL: http://www.webcitation.org/query?url=http%3A%2F%2Ftr.wikipedia.org%2Fwiki%2FCross_site_scripting. Son Erişim Tarihi: 15.12.2014.
49. Hydara, I., Sultan, A. B. M., Zulzalil, H., Admodisastro, N. (2015). Current state of research on cross-site scripting (XSS)–A systematic literature review. *Information and Software Technology*, 58, 170-186.
50. İnternet: Aydın M. OWASP Top10 Açıklık ve İlgili Araçlar. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwug.blogspot.com.tr%2F2011%2F03%2Fowasp-top10-acklk-ve-ilgili-araclar.html>. Son Erişim Tarihi: 08.12.2014.
51. İnternet: Top 10 2013-A2-Broken Authentication and Session Management. URL: http://www.webcitation.org/query?url=https%3A%2F%2Fwww.owasp.org%2Findex.php%2FTop_10_2013-A2-Broken_Authentication_and_Session_Management. Son Erişim Tarihi: 27.12.2014.
52. İnternet: Yeşilyurt Z. Web Uygulama Güvenliği ve Güvenli Kod Geliştirme Kaynak Belgesi. URL: http://www.webcitation.org/query?url=http%3A%2F%2Fwww.kadinyazilimci.com%2Fwp-content%2Fuploads%2F2014%2F09%2FWeb_Guvenligi_lyk2014_1zinnur9.pdf. Son Erişim Tarihi: 27.12.2014.
53. İnternet: Web Güvenliği Çalışma Grubu. En Kritik 10 Web Uygulaması Güvenlik Zayıflıkları. URL: http://www.webcitation.org/query?url=http%3A%2F%2Fcirt.ulakbim.gov.tr%2Fdokumanlar%2FCeviri_OWASP.pdf. Son Erişim Tarihi: 28.12.2014.

54. İnternet: Using Components with Known Vulnerabilities. URL: http://www.webcitation.org/query?url=https%3A%2F%2Fwww.owasp.org%2Findex.php%2FTop_10_2013-A9-Using_Components_with_Known_Vulnerabilities. Son Erişim Tarihi: 28.12.2014.
55. İnternet: Demir, B. Özgür Uygulamalar ile Web Güvenliği. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fseminer.linux.org.tr%2Fwp-content%2Fuploads%2FOzgurWebUygulamaGuvenciligi.odp>. Son Erişim Tarihi: 12.11.2014.
56. İnternet: Güvenlik Uzmanlarının Test Keçisi, WebGoat 5.1, URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.olympus.net%2Fblog%2Fhomeros%2Fguvenlik-uzmanlarinin-test-keçisi-webgoat-51-507662.html%23axzz3M9wEGQEs>. Son Erişim Tarihi: 13.11.2014.
57. İnternet: OWASP WebGoat Project. URL: http://www.webcitation.org/query?url=https%3A%2F%2Fwww.owasp.org%2Findex.php%2FCategory%3AOWASP_WebGoat_Project. Son Erişim Tarihi: 12.11.2014.
58. Guerbas, A., Addam, O., Zaarour, O., Nagi, M., Elhajj, A., Ridley, M., Alhajj, R. (2013). Effective web log mining and online navigational pattern prediction. *Knowledge-Based Systems*, 49, 50-62.
59. Arslan, H. (2008). Sakarya Üniversitesi Web Sitesi Erişim Kayıtlarının Web Madenciliği ile Analizi, Yüksek Lisans Tezi, *Sakarya Üniversitesi Fen Bilimleri Enstitüsü*, Sakarya.
60. Aye, T. T. (2011). Web log cleaning for mining of web usage patterns. *In Computer Research and Development (ICCRD), IEEE 2011 3rd International Conference on*, 2, 490-494.
61. Shu-yue, M., Wen-cai, L., Shuo, W. (2011). The Study on the Preprocessing in Web Log Mining. *In Knowledge Acquisition and Modeling (KAM), IEEE 2011 Fourth International Symposium on*, 315-317.
62. AL-HAMAMI, A., MOHAMMAD, A., HASSAN, S. (2006). Applying data mining techniques in intrusion detection system on web and analysis of web usage. *Information Technology Journal*, 5(1), 1-4.
63. Ezeife, C. I., Dong, J., Aggarwal, A. K. (2008). SensorWebIDS: a web mining intrusion detection system. *International Journal of Web Information Systems*, 4(1), 97-120.
64. Chen, Z., Feng, W. (2013). Detecting Impolite Crawler by Using Time Series Analysis. *In Tools with Artificial Intelligence (ICTAI), 2013 IEEE 25th International Conference on, IEEE*. 123-126.
65. Calzarossa, M. C., Massari, L., Tessera, D. (2013). An extensive study of Web robots traffic. *In Proceedings of International Conference on Information Integration and Web-based Applications & Services (410)*. ACM.

66. Velkoski, M., Martinovska, C., (2014). Analyzing Web Server Access Log Files Using Data Mining Techniques, *International Conference on Applied Internet and Information Technologies*, 321-326.
67. Dikaiakos, M. D., Stassopoulou, A., Papageorgiou, L. (2005). An investigation of web crawler behavior: characterization and metrics. *Computer Communications*, 28(8), 880-897.
68. Munk, M., Kapusta, J., Švec, P. (2010). Data preprocessing evaluation for web log mining: reconstruction of activities of a web visitor. *Procedia Computer Science*, 1(1), 2273-2280.
69. Kohavi, R. (2001). Mining e-commerce data: The good, the bad, and the ugly. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 8-13.
70. İnternet: Sawmill. URL: <http://www.webcitation.org/query?url=https%3A%2F%2Fwww.sawmill.net%2Findex.html>. Son Erişim Tarihi: 31.12.2014.
71. İnternet: Sawmill Kurulum. URL: http://www.webcitation.org/query?url=http%3A%2F%2Fdownload.cnet.com%2FSawmill-32-bit%2F3000-2651_4-10029810.html. Son Erişim Tarihi: 06.01.2015.
72. İnternet: Nihuo Web Log Analyzer (NWLA). URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.nihuo.com>. Son Erişim Tarihi: 27.03.2015.
73. İnternet: WEKA. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fweka.wikispaces.com%2FPri mer>. Son Erişim Tarihi: 06.01.2012.
74. Dener, M., Dörterler, M., Orman, A. (2009). Açık Kaynak Kodlu Veri Madenciliği Programları: Weka'da Örnek Uygulama. *Akademik Bilişim*, 9, 11-13.
75. İnternet: Veri Madenciliği-Weka Yazılımı. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.sevdanurgenc.com%2Farchives%2F840>. Son Erişim Tarihi: 24.04.2015.
76. Coşkun, C. (2010). Veri Madenciliği Algoritmaları Karşılaştırmaları, Yüksek Lisans Tezi, *Dicle Üniversitesi Fen Bilimleri Enstitüsü*, Diyarbakır.
77. İnternet: Bilginin Keşfi. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.gurunlu.com%2Fpost%2FiszekamNet-Bilginin-Kesfi.aspx>. Son Erişim Tarihi: 24.04.2015.
78. İnternet: My Weka Page. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.hakank.org%2Fweka%2F>. Son Erişim Tarihi: 24.04.2015.

79. İnternet: WEKA ile SVM. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fbilgisayarkavramlari.sadievrenseker.com%2F2011%2F09%2F19%2Fweka-ile-svm%2F>. Son Erişim Tarihi: 25.04.2015.
80. İnternet: ExceltoArffConvertor. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fsourceforge.net%2Fprojects%2Fexceltoarffconv%2F>. Son Erişim Tarihi: 15.01.2015.
81. İnternet: WEKA. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fresearch.cs.queensu.ca%2FHome%2Fcisc333%2Ftutorial%2FWeka.html>. Son Erişim Tarihi: 03.04.2015.
82. Kayaalp, K.(2007). Asenkron Motorlarda Veri Madenciliği ile Hata Tespiti, Yüksek Lisans Tezi, *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü*, Isparta.
83. İnternet: Apache Log Analyser for Security Specialists, URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fwww.infosec.lk%2F2011%2F02%2Fscalp-apache-log-analyser-for-security-specialists.html>. Son Erişim Tarihi: 24.08.2014.
84. İnternet: Apache Log Analyser for Security apache-scalp. URL: http://www.webcitation.org/query?url=http%3A%2F%2Fcode.google.com%2Fp%2Fapache-scalp%2Fsource%2Fbrowse%2Ftrunk%2Fdefault_filter.xml. Son Erişim Tarihi: 24.08.2014.
85. İnternet: Apache-scalp. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fcode.google.com%2Fp%2Fapache-scalp%2Fdownloads%2Flist>. Son Erişim Tarihi: 24.08.2014.
86. İnternet: PHPIDS. URL: <http://www.webcitation.org/query?url=http%3A%2F%2Fen.wikipedia.org%2Fwiki%2FPHPIDS>. Son Erişim Tarihi: 07.04.2015.
87. Choraś, M., Kozik, R. (2013). Evaluation of Various Techniques for SQL Enjeksiyon Attack Detection. *In Proceedings of the 8th International Conference on Computer Recognition Systems CORES 2013*, 753-762.
88. Doshi, J. C., Christian, M., Trivedi, B. H. (2014). SQL FILTER–SQL Enjeksiyon Prevention and Logging Using Dynamic Network Filter. *In Security in Computing and Communications*, 400-406.
89. Kozik, R., Choraś, M., Hołubowicz, W. (2015). Hardening Web Applications against SQL Enjeksiyon Attacks Using Anomaly Detection Approach. *In Image Processing & Communications Challenges*, 285-292.
90. Elia, I. A., Fonseca, J., Vieira, M. (2010). Comparing SQL Enjeksiyon detection tools using attack Enjeksiyon: an experimental study. *In Software Reliability Engineering (ISSRE), 2010 IEEE 21st International Symposium on*, 289-298.

91. Robu, R., Stoicu-Tivadar, V. (2010). Arff convertor tool for WEKA data mining software. In *Computational Cybernetics and Technical Informatics (ICCC-CONTI), 2010 International Joint Conference on*, 247-251.
92. Khamphakdee, N., Benjamas, N., Saiyod, S. (2014). Network traffic data to ARFF converter for association rules technique of data mining. In *Open Systems (ICOS), 2014 IEEE Conference on*, 89-93.
93. Fowler, C. A., Hammel, R. J. (2014). Converting PCAPs into Weka mineable data. In *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2014 15th IEEE/ACIS International Conference on*, 1-6.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ÇİNAR, Işıl
 Uyruğu : T.C.
 Doğum tarihi ve yeri : 03.07.1985, Niğde
 Medeni hali : Evli
 Telefon : -
 Faks : -
 e-mail : isilscinar@gmail.com



Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Üniversitesi	Devam Ediyor
Lisans	Selçuk Üniversitesi/Bilgisayar Müh.	2007
Lise	Aksaray Fen Lisesi	2003

İş Deneyimi

Yıl	Yer	Görev
2009-Halen	Genelkurmay Başkanlığı	Bilgisayar Mühendisi
2007-2009	Teknokom A.Ş./İzmir Teknopark	Yazılım Mühendisi

Yabancı Dil

İngilizce

Yayınlar

1. Çınar, I., Çınar, M.S. ve Bilge, H.Ş., (2014). Web Sunucu Loglarının Web Madenciliği Yöntemleri ile Analizi, *Akademik Bilişim 2014*, Mersin.

2. Çınar, I., Çınar, M.S. ve Bilge, H.Ş., (2014). Etkileşimli Televizyon: IPTV, *Akademik Bilişim 2014*, Mersin.
3. Akın, O., Çınar, I., Karaman, M., Bilekyigit, F., (2013). Siber Durum Farkındalığını Artırmada Etkili Bir Yöntem: Bayrağı Yakala (Capture the Flag), *ISCTurkey, 6th International Information Security & Cryptology Conference*, Ankara.
4. Çınar, M.S., Çınar, I. ve Bilge, H.Ş., (2013). Özgün Bir Şifreleme Algoritması Kullanan Güvenli Elektronik Posta Uygulaması. *ISCTurkey, 6th International Information Security & Cryptology Conference*, Ankara.

Hobiler

Gezi, kayak, müzik



GAZİ GELECEKTİR...