T.C
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÖZEL ANLAMLI İFADE İÇEREN VERİLERDE SIZINTI ÖNLEME İÇİN BİR MİMARİ TASARIM VE GERÇEKLEŞTİRİLMESİ

Murat TOPALOĞLU

Tez Danışmanı
Yrd. Doç.Dr. Erdem UÇAR

DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

EDİRNE, 2012

T.C
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÖZEL ANLAMLI İFADE İÇEREN VERİLERDE SIZINTI ÖNLEME İÇİN BİR MİMARİ TASARIM VE GERÇEKLEŞTİRİLMESİ

Murat TOPALOĞLU

DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez 04/05/2012 tarihinde aşağıdaki jüri tarafından kabul edilmiştir.

Yrd. Doç Dr. Erdem UÇAR
(Danışman)

Doç.Dr. Hasan Hüseyin BALIK
(Üye)

Doç.Dr. Yılmaz KILIÇASLAN
(Üye)

Yrd Doç. Dr. Erdinç UZUN
(Üye)

Yrd. Doç. Dr. İlhan UMUT
(Üye)

TEŞEKKÜR

Akademik çalışma gerçekleştirenlerin temel ihtiyacı yardım ve destektir.

Bu konuda çok şanslı olduğumu hissediyorum. Çalışmalarım boyunca her zaman beni destekleyen, rehberliğini, dostluğunu ve bilgilerini esirgemeyen kişilerle bir arada oldum.

Doktora tez danışmanım Yrd. Doç. Dr. Erdem UÇAR'a çalışmalarıma verdiği desteklerden dolayı teşekkür ederim.

Çalışmalarımın istatistiksel kısımlarında tecrübelerinden yararlandığım ve ilk aramamda yardımını esirgemeyen Yrd. Doç. Dr. M. Fatih AMASYALI, Öğr. Gör. Fatih AYDIN ve Arş. Gör. Edip Serdar GÜNER'e teşekkür ederim.

İstatistik çalışmasında ve kodlama esnasında tavsiyelerini esirgemeyen İlkay TAŞKIRAN, Alican GÜNDÜZ ve Rasim SAVCI'ya teşekkür ederim.

Ayrıca hayatım boyunca desteklerini hiçbir zamana esirgemeyen ve her zaman yanımda olan eşim, oğlum ve aileme teşekkür ederim.

İÇİNDEKİLER

1. GİRİŞ 1
1.1. Amaç 2
1.2. Literatür Özetleri 2
1.3. Materyal ve Yöntem 5
1.4. Tez Organizasyonu 6
2. ÖZEL ANLAMLI VERİLER 8
2.1. Gizlilik Sınıfları 9
2.2. Sınıflandırmanın Tamamlanması 10
2.3. Veri Erişim Kontrol Sistemleri 11
2.3.1. İsteğe Bağlı Erişim (Discretionary Access Control - DAC) 11
2.3.2. Zorunlu Erişim Kontrolü (Mandatory Access Control – MAC) 11
2.3.3. Rol Tabanlı Erişim Kontrolü (Role Based Access Control – RBAC) 11
3. VSE SİSTEMLERİNE GENEL BAKIŞ 12
3.1. VSE Tanımlaması 12
3.2. VSE Büyüyen İhtiyaç 13
3.3. Risklerin Tanımlanması 14
3.3.1. Kullanımdaki Veriler 16
3.3.2. Durağan Veriler 16
3.3.3. Hareket Halindeki Veriler 17
3.4. Tehditlerin Belirlenmesi 17
3.5. Bilginin Kurumsal Yapısı 18
3.5.1. Harici Depolama Ortamları 19
3.5.2. İnternet Tabanlı Yollar 19
3.6. VSE Politikalarının Önemi 19
3.7. Şirket Politikaları ve Örgütsel Önlemler 20
3.8. Yasal Uygunluklar 21
4. VSE SİSTEMLERİNDE SIZINTI ENGELLEME 22
4.1. VSE Çözümleri 22
4.2. İçerik Tabanlı Sınıflandırma 23
4.3. İstemci Tabanlı VSE Sistemi 23
4.3.1. Ağ Tabanlı VSE Sistemi 24

4.4. Kalıp Tepkileri 25
5. TEZDE KULLANILAN DİĞER YÖNTEMLER VE PROTOKOLLER 26
5.1. VSE Metin Sınıflandırma İçin Kullanılan Yöntemler 26
5.1.1. Naïve Bayes 27
5.1.2. Karar Ağaçları 27
5.1.3. Destek Vektör Makinesi 28
5.1.4. K-En Yakın Komşu Algoritması 28
5.1.5. K Ortalamalar Algoritması 29
5.1.6. N-Gram 29
5.1.7. Kelime Kökleri 29
5.1.8. Ağırlıklandırma Yöntemleri 30
5.1.8.1. Frekansa Göre Ağırlıklandırma 30
5.1.8.2. Terim Frekansı x Ters Doküman Frekansı 30
5.2. Ağ Tabanlı VSE Sisteminde Kullanılan Protokoller 31
5.3. HTTP Protokolü 31
5.3.1. URI (Uniform Resources Identifier) 31
5.3.2. Varlık 32
5.3.3. İçerik Kodlamaları 32
5.3.4. Transfer Kodlamaları 32
5.3.5. Medya Türleri 33
5.3.6. HTTP Mesajları 34
5.3.7. HTTP İstekleri 34
5.3.8. HTTP Cevapları 35
5.3.9. Kalıcı Bağlantılar 35
5.4. ICAP Protokolü 36
5.4.1. REQMOD 36
5.4.2. RESPMOD 38
5.4.3. HTTP Mesajlarının Kapsüllenmesi 39
5.4.4. Örnekler 39
5.4.5. ICAP Protokolünün VSE Amaçlı Kullanımı 42
5.5. HTTPS 43
5.5.1. HTTPS Trafiğinin Denetlenmesi 43

5.6. TLS 45
6. İSTEMCİ TABANLI VSE MİMARİSİ TASARIMI 47
6.1. Çekirdek (Kernel) Sürücüsü Uygulaması 51
6.2. Sistem Çağrı Tablosu Manipülasyonu (SSDT Hooking) 51
6.3. VSE İstemci Mimarisinin Özellikleri 55
7. VSE SİSTEM MİMARİSİ İÇİN METİN SINIFLANDIRMA 56
7.1. Deneysel Çevre 57
7.3. Deneysel Sonuçlar 59
8. AĞ TABANLI VSE SİSTEM MİMARİSİ 73
8.1.1. Kullanılan Ek Kütüphaneler 73
8.1.2. Kullanılan Yardımcı Program 74
8.1.3. Thread Mimarisi 74
8.2. M-ICAP Programının Çalışma Şekli 75
8.2.1. Md-5 Parmak İzi Kontrolü 77
8.2.2. Düzenli İfadeler Aracılığıyla Desen Eşleştirme 78
8.2.3. N-Gram Tabanlı Doküman Sınıflandırma 78
8.3. Hassas Veriye Karşı Uygulanacak Davranış 82
9. DEĞERLENDİRME 83
10. KAYNAKLAR 86

ŞEKİL LİSTESİ

Şekil 5.1. Örnek bir URL 32

Şekil 5.2. Parçacıklı Transfer Kodlaması Kullanan http Mesajı 33

Şekil 5.3. REQMOD İçin Tipik Veri Akışı 37

Şekil 5.4. RESPMOD İçin Tipik Veri Akışı 38

Şekil 5.5. ICAP Reqmod İstek Mesajı 40

Şekil 5.6. ICAP Reqmod Cevap Mesajı 40

Şekil 5.7. ICAP Respmod İstek Mesajı 41

Şekil 5.8. ICAP Respmod Cevap Mesajı 42

Şekil 5.9. Transport Layer Security İletişimi 45

Şekil 6.1. VSE Organizasyon Şeması 47

Şekil 6.2. VSE İstemci Mimari Yapısı 49

Şekil 6.3. VSE Sürücüsü Kernel Şematiği 50

Şekil 6.4. SSDT Hook Tablosu 53

Şekil 7.1. Naïve Bayes sınıflandırıcının 1. deney f-measure sonuçları 61

Şekil 7.2. J48 sınıflandırıcının 1. deney f-measure sonuçları 61

Şekil 7.3. IBK sınıflandırıcının 1. deney f-measure sonuçları 62

Şekil 7.4. SVM sınıflandırıcının 1. deney f-measure sonuçları 62

Şekil 7.5. NaïveBayes sınıflandırıcının 2. deney f-measure sonuçları 64

Şekil 7.6. J48 sınıflandırıcının 2. deney f-measure sonuçları 64

Şekil 7.7. IBK sınıflandırıcının 2. deney f-measure sonuçları 65

Şekil 7.8. SVM sınıflandırıcının 2. deney f-measure sonuçları 65

Şekil 7.9. Naïve Bayes sınıflandırıcının 3. deney f-measure sonuçları 66

Şekil 7.10. J48 sınıflandırıcının 3. deney f-measure sonuçları 67

Şekil 7.11. IBK sınıflandırıcının 3. deney f-measure sonuçları 67

Şekil 7.12. SVM sınıflandırıcının 3. deney f-measure sonuçları 68

Şekil 7.13. Naïve Bayes sınıflandırıcının 4. deney f-measure sonuçları 69

Şekil 7.14. J48 sınıflandırıcının 4. deney f-measure sonuçları 69

Şekil 7.15. IBK sınıflandırıcının 4.deney f-measure sonuçları 70

Şekil 7.16. SVM sınıflandırıcının 4.deney f-measure sonuçları 70

Şekil 8.1. M-ICAP Tasarım Şeması 75

Şekil 8.2. M-ICAP Respmod izinsiz içerik engelleme 77

Şekil 8.3. Benzerlik değerinin hesaplanması 80

Şekil 8.4. N-gram tabanlı doküman sınıflandırma için veri akışı 81

TABLO LİSTESİ

Tablo 6.1. VSE istemci mimari özellikleri..... 55

Tablo 7.1. Yapılan deneylerin parametre seçimleri 60

Tablo 7.2. Birinci deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları 63

Tablo 7.3. İkinci deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları 66

Tablo 7.4. Üçüncü deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları 68

Tablo 7.5. Dördüncü deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları 71

KISALTMA LİSTESİ

DLP: Data Leak Prevention (Veri Sızıntısı Engelleme)

DRM: Digital Rights Management (Dijital Haklar Yönetimi)

HTTP: Hypertext Transfer Protocol (Hipermetin Aktarma İletişim Kuralı)

HTTPS: Hypertext Transfer Protocol Secure (Güvenli Hipermetin Aktarım İletişim Kuralı)

IANA: The Internet Assigned Numbers Authority (Internet Atanmış Numaralar Yetkilisi)

IBK: Örnek Tabanlı Sınıflandırma Algoritması

ICAP: Internet Content Adaption Protocol (İnternet İçerik Uyarlama Protokolü)

J48: C4.5 Karar Ağaçları

KNN: K – En Yakın Komşuluk Algoritması

MD5: Message-Digest Algorithm 5 (Mesaj Özetleri Algoritması)

MIME: Multipurpose Internet Mail Extensions (Çok Amaçlı İnternet Posta Uzantıları)

REQMOD: Request Modification Mode (İstek Uyarlama Modu)

RESPMOD: Response Modification Mode (Cevap Uyarlama Modu)

RTP: Real – Time Transport Protocol (Gerçek Zamanlı İletişim Protokolü)

SIP: Session Initiation Protocol (Oturum Başlatma Protokolü)

SSDT: System Service Description Table (Sistem Çağrı Tablosu Manipülasyonu)

SVM: Support Vector Machines (DVM: Destek Vektör Makinesi)

TCP/IP: Transmission Control Protocol/ Internet Protocol

Tf x idf: Terim Frekansı x Ters Doküman Frekansı

Tf: Frekansa Göre Ağırlıklandırma

TLS: Transport Layer Security (Taşıma Katmanı Güvenliği)

URI: Uniform Resources Identifier (Uniform Kaynak Tanımlayıcı)

ÖZET

DOKTORA TEZİ

ÖZEL ANLAMLI İFADE İÇEREN VERİLERDE SIZINTI ÖNLEME İÇİN BİR MİMARİ TASARIM VE GERÇEKLEŞTİRİLMESİ

Kurumsal yapının sürekliliğini sağlamak verinin korunmasından geçmektedir. Kullanılan ve yeni oluşturulan verilerin tamamının belli bir düzen içerisinde sınıflandırılması bilgi çıkarımı ve kurumsal işleyiş açısından önemlidir. Kullanıcıların hassas bilginin farkında olması gerekmektedir. Veri korunması için iyi niyetli kullanıcı düşüncesi önemlidir. Ancak veri sızıntısının gerçekleşmesi durumunda kurumsal zararların mali boyutlarının büyüklüğü kaçınılmaz olacaktır. Verilerin korunmasını katı kurallar içerisinde yapmak sistemin verimli işleyişine engel olacaktır. Kullanıcıların daha az kısıtlandığı, hassas olan verilerin akışının kontrol edildiği, sızdırma yapabilecek kullanıcıların engellendiği ve kullanıcıların farkında olmadan yapılabileceği hataların en aza indirgenmesini sağlayacak çözümlere ihtiyaç duyulmaktadır. Bu çözüm kurumumuza zarar verebilecek eylemleri engellemeyi hedef almaktadır. Kurumsal bilgilerin korunması mobil, uyarlanabilir ve en az müdahaleye ihtiyaç duyan yapıda olmalıdır. Riskleri azaltmaya yardımcı olabilecek en umut verici çözüm veri sızıntısı önleme sistemi (VSE) olarak adlandırılmaktadır.

Tez veri sızıntısı engelleme konusunda yeni bir mimari önermektedir.

Bu mimari üç kısımdan oluşmaktadır. Bunlar sırası ile İstemci tabanlı veri sızıntısı engelleme, ağ tabanlı veri sızıntısı engelleme ve sızıntı tespitinde makine öğrenmesi yöntemlerinin kullanılarak yapılan deneysel sonuçların incelenmesidir.

Anahtar Kelimeler: *DLP, Veri Sızıntısı Engelleme, Metin Sınıflandırma, ICAP, İstatistiksel İçerik Analizi*

ABSTRACT

PHD THESIS

DESIGN AND IMPLEMENTATION OF AN ARCHITECTURE FOR DATA LEAK PREVENTION IN DATA CONTAINING SPECIFIC MEANING

The continuity of the institutional structure is ensured through the protection of the data. Classification of all the data being used or newly created in a specific order is important for data collection and institutional operations. Users should be aware of sensitive information. It is important for users to be well intentioned for data protection. However, the realization of data leakage may create some problems especially when corporate losses are taken into account. It is not possible to perform data protection within strict rules. Solutions are needed that are less restricting, control flow of sensitive data prevent the users from causing leakage and minimize errors. The solution we propose is intended to block out actions that could be harmful to our institution. Protection of information should be mobile, adaptable, and of a natüre requiring least intervention. The most promising solution that can help reduce the risks is labeled as data leakage prevention (DLP).

The thesis proposes a new approach to data leak prevention. This architectural design consists of three parts; these are client-based data leakage prevention, network-based data leakage prevention and analyzing experimental results using machine learning methods when leakage is detected.

Keywords: *DLP, Data Leak Prevention, Data Loss Prevention, Text Classification, ICAP, Internet Content Adaption Protocol, Statistical Content Analysis*

1. GİRİŞ

Günümüzde çeşitli bilgi güvenliği yöntemleri, modelleri ve bunlara ilişkin yazılımlar vardır. Kurumsal güvenlik ihtiyaçları her geçen gün artan bir ihtiyaç olmakta ve yeni oluşan sorunlara karşı yeni stratejiler geliştirilmektedir.

Veri (data), bir kavram, varlık veya konunun herhangi bir simge öbeği ile ifade edilmesidir. Bilgi (information) ise herhangi bir konu veya varlık ile ilgili verilerin bir araya gelmesi ile ortaya çıkan açıklayıcı bir bütündür. Veri, gerçeklik üzerinde yapılan gözlemlerin sonucu ve bu anlamda bilginin üretildiği hammaddedir.

Veri, kullanıcılar için herhangi bir anlam ifade etmeyen olgulardır ve sayısal değerlerin yanı sıra sayısal olmayan değerleri de kapsayabilir (Gökçen, 2007). Bilgi, kişiler veya kurumlar için anlamlı olması yanı sıra belli maddi veya manevi değerleri de içermektedir. Belli süreçler ya da işlemler sonunda bu bilgiler kazanılmış bilgiye (knowledge) dönüşmektedir. Günümüzün rekabetçi ekonomileri ve teknolojileri düşünüldüğünde; verileri bilgi ve kazanılmış bilgiye dönüştürebilen kurumlar, çok daha başarılı, hızlı ve etkin stratejik kararlar alabilmektedir (Vercellis, 2009).

Günümüzde bilişimde yaşanan müthiş gelişim hızı ile beraber kişiler ve kurumlar; yazılımlar ve bilgisayarların kullanılmasıyla yapılan sahtekârlıklar, bilgi hırsızlığı, bilgisayar korsanları, elektronik saldırılar, bilgi sızdırma ve ilgili kuruluşların kendi çalışanlarınca oluşturulabilecek potansiyel iç saldırılar gibi çok çeşitli tehditlerle karşı karşıya kalmaktadır (Rost, 2011).

VSE (Veri Sızıntısı Engelleme – Data Leak Prevention) kurumların sahip oldukları verilerin, kurum içi nedenlerle dışarı sızdırılmasını engellemeyi amaçlayan birden fazla aşamadan oluşan güvenlik ifadesidir.

Tez amacı tüm VSE çözümlerini içine alan, tüm sorunlarına cevap arayan mimari geliştirmektir. Bu tez çalışmasında geliştirilen mimari VSE için metin sınıflandırma, istemci tabanlı VSE sürücüsü ve sunucu tabanlı VSE sistemi tasarımını ve yazılımsal kodlama işlemlerini içermektedir.

1.1. Amaç

Kurumsal yapının bir parçası olan güvenlik verilerinin korunması ile ilgili çözümler, her zaman değişen ve güncellenen versiyonları ile sürdürülmektedir. Veri bütünlüğü; finansal bilgiler, stratejik hedefler, pazar planları, müşteri portföyü, bütçe uygulamaları, kişisel bilgiler gibi alanların birlikteliğini sağlamaktadır. Bu bilgiler kurumların geleceği için önem taşımaktadır. Son yıllarda Wikileaks olayından itibaren veri güvenliğini etkileyen sızıntı olayları başlamış ve günümüzde artarak devam etmektedir. Veri sızıntısını engelleme sistemleri ilk olarak adını bu olay ile duyurmaya başlamıştır. Bu durum mevcut bir problemi gün yüzüne çıkarmıştır. Kurumsal verilerin çeşitli kaynaklardan kurum dışına sızdırılmasını engellemenin önemi anlaşılmaya başlanmıştır. Kurumsal yapı içerisinde kullanıcıların veri akışını kontrol edecek ve veri güvenliğini sağlayacak çözümlere ihtiyaç her geçen gün artmaktadır. Geliştirilecek yeni teknolojiler veri ihlallerini en aza indirgemek için bazı çalışmalar sunacaktır. Bu tez kurum içerisinde veri akışını kontrol edecek ve hassas verilerin akışını engelleyecek çözüm geliştirmek için hazırlanmıştır. Tezin amacı veri sızıntısından korunma amacıyla bütünleşik bir sistem geliştirmektir. Geliştirilen sistem ve yazılımlar veri sızıntı engelleme, uç nokta yönetimi (data in use) ve ağ üzerindeki veriler (data in motion) olmak üzere kritik derecelendirmeyi saptayacak, izleyebilecek ve gerektiğinde müdahale ederek verilerin korunmasını sağlayacak sistemdir. Veri sızıntısını engellemenin yöntemi anlamlı veri içeren ifadelerin kesilmesinden geçmektedir. Geliştirilen veri sızıntısı engelleme sistemlerinin genel sorunlarına bakılarak teze yön verilirken, istemci tabanlı veri akışında ve ağ tabanlı içerik analizinde verilerin sınıflandırılması problemlerinin önüne geçilmiştir. Bu tezde düzenli ifadeler, istatistiksel içerik analizi yöntemleri ve algoritmaları kullanılmaktadır. Düzenli ifadeler sadece standart içeriği etkileyeceği için korunma imkânları kısıtlı kalmaktadır. Bu tezde makine öğrenmesi yöntemleri kullanılarak öğrenme sağlanmış ve metin sınıflandırma metotları ile koruma üst seviyelere yükseltilmiştir. Veri sızıntısı engelleme sistemi için etkili ve verimli bir mimari tasarlanmıştır.

1.2. Literatür Özetleri

Kasparick tarafından yüksek lisans tezi olarak sunulan çalışma VSE konusunda genel ve özel tanımlamalara sahiptir. Uç nokta yönetimi alanında kernel hooking

yöntemlerinin tanıtıldığı ve burada ki yöntemlerin nasıl kullanılacağının anlatıldığı yararlı bir tezdir (Kasparick, 2008).

Backes ve diğerleri, veri akışı ve analizi ile hangi verinin sızıntı olduğuna karar vermek için metot geliştirmiştir (Backes, 2009).

Borders ve Prakash ise kurum dışına çıkan web trafiğinin analizinde (http protokolü) veri sızıntısını engellemek için geliştirdikleri algoritmayı konu almışlardır (Borders, 2009).

Lee ve diğerleri, dolaylı gizli kanal saldırılarında hassas verilerin yetkisiz kişilerin eline geçmemesi için paylaşılan kaynakların kullanılmasını inceleyen çalışma yapmışlardır. Ancak bu tür mekanizmalar gizli kanallar aracılığı ile veri sızıntılarına karşı güçlü bir koruma sağlamamıştır. Bu çalışmada tespit ve renklendirme anlayışlı istemci dosya sistemleri ve potansiyel veri sızıntısını örneklendirmeyi amaçlamıştır. Bunun için renkli dosya izlemi süreci otomatikleştirilmiş ve mobil ajan tabanlı bir yaklaşım önerilmiştir (Lee, 2009).

Dijital suçların kanıtlarını toplamak amacı ile geliştirilmiş bilgisayar tabanlı sistemdir. Lee ve diğerleri, çalışmada registry analizi, günlük dosya analizi, cd/dvd yazıcı log analizi gibi dosya analizlerini yaparak veri sızıntılarını engellemeye çalışmaktadır (Lee, 2009).

Luft, lisans tezinde genel VSE tanımlamalarını konu almış, prosedür açıklamalarını yapmıştır. Mcafee ve Websense güvenlik ürünlerinin çözümleri incelenmiştir. Çıkan sonuçlara da tezinde yer vermiştir (Luft, 2009).

Papadimitriou ve Garcia, sızıntı tespit olasılıklarının hesaplandığı veri dağılımı stratejileri önermişlerdir. Yayımlanan verilerin değişikliği sırasında bu metodun geçerliliğine güvenin nasıl değişeceği belirtilmemiştir (Papadimitriou, 2009).

Xiaosong ve diğerleri, kurumsal ortamda bilerek ya da bilmeyerek yapılan hassas verilerdeki sızıntıyı engellemeye çalışmaktadır. Uygulama zorunlu ve şeffaf filtreleme özelliği sunmaktadır. Kullanıcı esnekliği ve veri güvenliğine verilen önem çalışmanın konusu olmuştur (Xiaosong, 2009).

Bhaskar ve diğerleri, makalelerinde hassas veri içeren veri kümesinin anlam desenlerinin çıkarılması ve veri keşfinin sağlanması konusunu işlemişlerdir. Veri setinde bulunan hassas kayıtları keşfetmek için "k" sıklık desenleri ile algoritma kullanılmıştır. Algoritmalar gizliliği tahmin etmeye çalışmakta, dış verinin varlığında anlamlı gizlilik garantisi tartışılmaktadır (Bhaskar, 2010).

Kim ve Kim, gizlilik ihlalleri seviyelerini göz önüne alarak veri sızıntısı engelleme sistemi önermektedir. Veri birimlerine dayalı gizlilik ihlali dereceleri için nicel bir yöntem önerilmektedir. Çalışanların veri işlemleri takip edilerek hangilerinin buna maruz kaldığı belirlenmeye çalışılmaktadır. Veri sızıntısının önem derecesini belirlemek için yöntem önerilmektedir (Kim, 2010).

Marecki ve diğerleri, veri sızıntısını engellemede veri kaynağı ve birçok alıcı arasındaki çeşitli nedenlerle oluşabilecek veri kayıplarını anlatılmaktadır. Veri sızıntısı engellemede problemin çözümü için markov decision process ile izleme mekanizması yaratılmaktadır. Gönderici için en uygun veri paylaşımı türetilir ve kötü niyetli alıcılar içinde en uygun veri sızıntı engelleme teknikleri incelenmektedir (Marecki, 2010).

Al-Fedaghi çalışmasında veri sızıntısı engelleme sistemleri, izleme ve hassas veriyi koruyan, yönetim politikaları, veri akışını güçlendiren ve sızıntı denetimini gerçekleştiren merkezi veri yönetim sistemini konu almaktadır. VSE tasarımındaki birçok geçerli yaklaşım sistem bağımlı ve belirli bir donanım içeren ticari yazılım sistemleridir. Bu çalışmada VSE sisteminde bilginin yaşam döngüsü ve iş süresince farklı süreçler kullanarak temel bir VSE sisteminin tasarımını konu alınmıştır. Yaklaşımların gerçekçi olduğunu göstermek için bazı testler uygulanmıştır (Al-Fedaghi, 2011).

Hart ve diğerleri, çalışmalarında kurumsal belgelerin hassas veri içerdiğini anlamak için metin sınıflandırma algoritması kullanmışlardır. Sınıflandırıcı veri setinde %97 başarı sağlanmıştır (Hart, 2011).

Raman ve diğerleri, bu çalışmada teşvik girişimi ile ilişkili alan soruları ve alanların incelenmesini yapmaktadır. Veri sızıntısı engelleme sorununu tanıtmak, mevcut yaklaşımları tanımlamak ve alanında potansiyel arama yönleri özetlenmektedir.

Saldırı tespit teknikleri ve veri sızıntısı engelleme sorunları farklıdır. Çalışma genel amaçlı bazı çözümleri anlatmaktadır (Raman, 2011).

Kurumsal ağlardaki miras uygulamalarda pratik veri sızıntısı engelleme ile alakalı çalışma soy ağacı şeklinde bir sunumda bulunmakta, ağ üzerindeki miras uygulamalar için veri akışının kontrolünü yürütülmektedir. Mundada ve diğerleri çalışmalarında veri akışı kontrol sistemleri, etiketleme yöntemi ve istemci ile ağ arasında verinin seyahat ve işlemlerinde veri akış kontrol aşamaları anlatılmaktadır (Mundada, 2011).

Uç nokta yönetiminde veri dağıtıcısının güvenilir olduğu varsayılmıştır. Yetkisiz ortamlarda bulunan ve sızdırılmaya çalışılan bilgiler mevcuttur. Papadimitriou ve Garcia, çalışmalarında sızıntı tanımlaması için veri dağılım stratejilerini açıklamaktadır. Sızıntının gerçekten olduğunu ya da sızıntı gibi görünen fakat aldatıcı olan bir durumdan ibaret olup olmadığına karar verilmeye çalışılmaktadır (Papadimitriou, 2011).

1.3. Materyal ve Yöntem

Tezin gerçekleştirimini üç aşamaya ayırmak mümkündür.

Birinci aşamada veri sızıntısı engelleme sisteminin uç nokta yönetiminde SSDT Hooking yöntemi kullanılarak önemli veri içeren dokümanlarının düzenli ifadeler ile aranması sağlanmıştır. Önceden tanımlanmış ve yönetim sunucusu tarafından güncellenebilen konfigürasyon dosyasından alınan tanımlamalar ile kullanıcının kopyalama, yapıştırma ve yeni dosya yaratma gibi işlemlerinde yazılan sürücünün işletim sistemine müdahalesi sağlanmaktadır. VSE uç nokta yönetim sürücüsüne yönlendirilen işlem ile veri sızıntısını engelleyen birinci basamak C programlama dili ile çözümlenmiştir.

Veri sızıntısını engellemenin ikinci kısmı, kurumsal ağ üzerinde taşınan verinin incelenmesi ile yapılmaktadır. Düzenli ifadeler verilerin incelenmesi için bu kısımda yeterli olmamaktadır. Makine öğrenmesi ile metin sınıflandırılması ve dokümanlar için otomatik öğrenme sisteminin sağlanması gerekmektedir. Metin sınıflandırma işlemi için kurumsal dokümanların sayısal forma dönüştürülmesi gerekmektedir. En sık kullanılan

özellik çıkarım tekniklerinden words ve n-gram, kelime kümeleme yöntemlerinden K-Means algoritması kullanılmıştır. Ağırlıklandırma yöntemlerinden frekansa göre ağırlıklandırma, terim frekansı x ters doküman frekansı yöntemleri tercih edilmiştir. Bu özellik çıkarım ve ağırlıklandırma yöntemleri ile birlikte 16 farklı özellik vektörü kullanılmıştır. Bu vektörler üzerinde NaïveBayes, K-En Yakın Komşuluk, Karar Ağaçları ve Destek Vektör Makinesi sınıflandırma algoritmaları uygulanmıştır.

Veri sızıntısı engellemenin son kısmında incelenen ağ tarafı kurumsal yapının çıkışlarını kontrol etmektedir. ICAP client üzerine yazılan İnternet İçerik Uyarlama Protokolü (ICAP) sunucusu ağ akışının tamamından sorumlu olmakta, öğrenebilen algoritmalar ile doküman analizleri yapılmaktadır. Sunucu üzerinde makine öğrenmesi teknikleri kullanılarak otomatik doküman analizi yapılmakta hassas dokümanların akışı engellenerek kurumsal ağın güvenliği sağlanmaktadır. Ayrıca yönetilebilir ortam sunulabilmesi için yönetim konsolu tasarlanmıştır. Tüm sistem ayrıntıları buradan incelenebilmekte ve tüm ayarlar buradan yapılandırılıp, yönetilebilmektedir.

1.4. Tez Organizasyonu

Belirtilen amaçlar doğrultusunda veri sızıntısı engelleme mimari tasarımı için gerekli olan bileşenler tez incelenmesini oluşturulmuştur. Sistem bölümlere ayrılarak incelenmiş, geliştirilen uygulamalar ve istatistiksel çalışmalar neticesinde sonuçlar değerlendirilmiştir.

Tezin organizasyonu da belirtilen duruma uygun olarak ortaya çıkmıştır.

İlk bölümde tez amacı, VSE sistemleri ile ilgili yapılmış çalışmalar, materyal ve yöntem incelenmiştir. Yapılan incelemeler ayrıntıları ile bölümde anlatılmıştır.

İkinci bölüm tez için temel bilgiler içermektedir. Özel anlamlı verinin ne olduğu açıklanmaktadır. Özel anlam içeren verilerin gizlilik sınıfları anlatılmış ve sınıflandırmanın tamamlanması için değerlendirilecek yöntemler konu alınmıştır.

Üçüncü bölüm, tezin materyal kısmını oluşturmaktadır. Veri sızıntısının tanımı, nasıl oluşabileceği, ne iş yaptığı ve nasıl kullanıldığı anlatılmaktadır. VSE sistemlerinin genel tanımlamaları burada yapılmaktadır.

Dördüncü bölüm literatür taramasının geniş özetini içermektedir. VSE sistemlerinde sızıntı engellemenin nasıl yapıldığı ve hangi basamakları kapsadığı konu alınmıştır. Veri sızıntısı engelleme sistemlerinin oluşma basamakları ve bileşenleri anlatılmaktadır.

Beşinci bölüm özel anlamlı ifade içeren verilerde sızıntısı engellemeyi amaçlayan tez için tasarlanan sistem mimarisinin genel tanımlı yöntem ve protokol ihtiyaçlarının tanımlamalarını kapsamaktadır. Bu bölümden sonra yapılan bütün çalışmalar tezde geliştirilen mimariyi anlatmaktadır.

Altıncı bölüm geliştirilen VSE mimarisinin istemci tarafını kapsamaktadır. İstemci tarafında ki mimarinin açıklamalarını içermektedir. Bu mimari SSDT hooking yöntemini içeren kompleks istemci arayüzlerini açıklamaktadır.

Yedinci bölüm geliştirilen VSE mimarisi için metin sınıflandırmanın anlatıldığı, hassas dokümanların otomatik olarak bulunmasını sağlayacak, makine öğrenmesi yöntemlerinin kullanıldığı bölümdür. İstatistiksel çalışma Weka [1] programı kullanılarak yapılmış ve otomatik sınıflandırmanın ayrıntıları ve deneysel sonuçları paylaşılmıştır.

Sekizinci bölümde geliştirilen VSE mimarisinin kurumsal ağı paylaşan tüm kullanıcıların veri sızıntı takibini sağlayacak sistem tanıtılmıştır. Sistem ICAP sunucusu üzerine detaylı bir kodlama ürünü olarak veri takibini sağlamaktadır.

Son bölümde ise yapılan uygulama ile elde edilen sonuçlar ve değerlendirmeler toparlanarak tartışılmakta ve gelecekte yapılabilecek çalışmalar konusunda bilgi verilerek tez son bulmaktadır.

2. ÖZEL ANLAMLI VERİLER

Kurumlar iş alanları ile ilgili kullandıkları verilerini depolama ortamlarında saklamaktadır. Zaman ile kurum içerisinde toplanan verilerin tamamı çeşitli değerlere sahip olur. Verilerin sahip oldukları bu değerler zamanla kurumların özelini oluşturmaktadır. İş hayatının sürekliliği açısından kurum içi veya dışında sahip olunan verilerin hareketi söz konusu olmaktadır. İyi niyetli çalışma şartlarından kaynaklanan bu verilerin paylaşılma eğilimi, beraberinde riskleri de getirmektedir. Bu riskler hassas kurumsal verilerin başkalarınca öğrenilmesi veya istenmeyen birimlerin eline geçmesi ile oluşmaktadır. Bu durum kurumlar için arzu edilmeyeceğinden, yayımlanan verilerden, hassas verilerin süzülmesini mecbur kılmaktadır. Özel anlamlı veri, organizasyon için "değeri olan" anlamını taşımaktadır.

Veriler kurumlar için hayati değerlere sahiptir. Hassas içerikli verilerin sınıflandırılması, korunması için ilk adımdır. Kurumun sahip olduğu müşterileri bilgileri, kredi kart bilgileri, finans verileri, çalışanların kişisel verileri, dokümanları ve veritabanları hassas içerikli ortamları oluşturmaktadır. Bu bilgilerin e-posta, harici depolama üniteleri gibi birimler ile dışarıya sızdırılmasını engellemek için sınıflandırma kurallarını oluşturmak gerekmektedir. Sınıflandırma, sızıntı engelleme amacı ile kurumsal verilerin değerlendirilmesi için gereklidir. Çünkü bu veriler hamdır ve düzensizdir. Bu nedenle verilerin bir düzene koyulması, sınıflandırılması ve gruplandırılması çalışmalarının yapılması gerekmektedir. Kurumsal verilerin çözümlenmesinde kullanılacak sınıflandırma tekniklerinin ve bunların neden işleneceği konusu, koruma yöntemlerinin geliştirilmesinde kullanılabilir. Sınıflandırma yöntemleri bilgi toplama yöntemi ve teknikleri yanında verilerin nasıl korunacağını da açıklamaktadır. Verilerin, sınıflandırma amaçları doğrultusunda, temel öğelerini ve karakterlerini belirleme ve ayırt etme işlemlerine, verilerin çözümü ya da çözümlenmesi denir [3]. Veri sınıflandırma şu amaçlarla yapılmaktadır:

- Verileri özetlemek,
- Korunacak bilgilere karar vermek,
- Varsayımlara dayalı olarak anlamlılık çalışmaları yapmak,
- Model oluşturmak.

2.1. Gizlilik Sınıfları

Kişisel iletişim yolları olarak adlandırılabilir ve e-posta, telefon, faks bunların arasında sayılabilir. Kurum için verilerin gizlilik sınıflandırılması aşağıdaki gibidir:

- Sınıflandırılmamış,
- Kişisel,
- Gizli,
- Çok gizli,

İşte bu listede aşağıdaki gibi hiyerarşik bir yapı mevcuttur (Kasparick, 2008):

Sınıflandırılmamış < Kişisel < Gizli < Çok gizli

Kurumlar var oldukları süre boyunca veri ile çalışmaya mecbur oldukları için verilerini yönetmek zorundadır. Bilgi yaşam döngüsünün amacı kurumun iş hedefleri, gizliliği, pazar ihtiyaçları karşılanırken etkili süreçler ile uygun teknolojilerin kullanılması için kurumsal bilgi yönetim kuralları geliştirilmesidir. Bilgi yaşam döngüsü, karşınıza çıkabilecek her türlü ihtiyaçta; veri yönetimi maliyetlerini düşürmenizi, kurumsal uygulama performansınızı artırmanızı ve yasal riskleri en aza indirgemenizi sağlamaktadır. Bilgi bir kurum tarafından oluşturulur ya da çeşitli yöntemler ile dışarından alınır. Kurum bilgi yönetim politikaları bilgi koruması konusunda net bir yön tayin etmek zorundadır. Kurum bilgi sınıflandırma yöntemleri, hızlı ve uygun bir şekilde veri korumasına yardımcı olacaktır. Kurum açısından hayati önem taşıyan ve bu isimle etiketlenmiş verilerini güvenli yöntemler ile yaratmalı ve koruma altına almalıdır. Kurum dışarıdan aldığı (sözleşme vb.) verileri çeşitli standartlar kullanarak korumalı, veri sızıntısını engelleyecek, hata tespiti yapabilecek ve sorunları en aza indirgeyecek yapıyı oluşturmalıdır. Veri için bu andan itibaren hayat başlamış ve bilgi değerini yitirene kadar da devam etmektedir. Hassas veriler dahil olmak üzere tümü her zaman koruma altında tutulmalıdır. Bu kısımda tekrar, açıkça belirtilmiş politikalar ile veri sınıflandırma şeması kaza riskini azaltmaya yardımcı olacaktır. Tüm hayatı boyunca veriler bir düzen içinde tutulmalı, değerlendirilmesi için uygun bir sınıflandırmaya atanması şartı aranmalıdır. Veri sınıflandırma sonrasında

kurumsal değişiklikler ile veriler zamanla daha hassas veya daha az hassas hale gelebilmektedir [5]. Veri ömrünün sonunda kullanılmayan ya da bir ortamda arşivlenmiş olarak durmaktadır.

2.2. Sınıflandırmanın Tamamlanması

Kurumsal verilerin sınıflandırılması için çeşitli teknikler kullanılabilir. Veri sınıflandırması kurumların veriye ihtiyacı ölçülerek yapılacaktır. Veriler yasalara uymak, sorumluluk sınırlandırılmak ve gizliliğini korumak için sınıflandırılır. Verilerin sınıflandırılması bilgi kaybına yol açmasına karşın yapılacak hesaplamalarda ve sonuç çıkarmada çeşitli kolaylıklar sağlamaktadır. Verinin değerini ölçmek için sınıflandırma kurumlarda yaygın olarak kullanılmaktadır. Sınıflandırılması tamamlanan veri etkin olacağı süre boyunca aynı sınıflandırma kurallarını koruyacaktır. Önemini kaybeden ya da yeniden sınıflandırılma ihtiyacı olan veriler, işlemin ardından tekrar etkin sınıf kümesine geri dönmektedir. Yasal gereksinimlerinden dolayı tekrar sınıflandırılma isteği, veriyi aynı aşamalardan bir kez daha geçirecektir. Kurumsal veriler etikete sahip olarak, çok gizli, gizli (secret – confidential) ya da kullanıma açık gibi sınıflandırılır. Kurumlar standart sınıflandırma düzeylerini belirlemeli ve koruma etkenlerini yerine getirmelidir.

Verinin sınıflandırılması ve kontrolü el ile yapılamayacağından bu işlemi otomatik olarak yapan bir sistemin geliştirilmesi gerekmektedir. Otomatik sınıflandırma son kullanıcı müdahalesi olmaksızın yapılmakta ve arka planda çalışmaktadır. Burada sistem yöneticileri, sınıflandırma kurallarını oluşturmaktadır. Dosya sisteminde çeşitli teknikler kullanılarak veriler sınıflandırmaya çalışılmaktadır. Sınıflandırmanın bazı avantajları ve dezavantajları vardır. Dezavantajı sınıflandırma işlemine müdahale edilememesidir. Avantajı ise kullanıcının kasten sınıflandırılmamış ya da hatalı sınıflandırılmış veri bırakmasını engellemektedir. Elle sınıflandırma verileri tek tek, uzun zamanda ve en az dosya üzerinde işlem sonucuna ulaşarak sınıflandırabilmektedir. Bu yüzden kontrolü kurumsal ağ üzerinde bu şekilde yapmak zor olacaktır.

2.3. Veri Erişim Kontrol Sistemleri

Kontrol stratejileri üç bölüme ayrılmıştır (Kasparick, 2008):

- İsteğe Bağlı Erişim (DAC)
- Zorunlu Erişim Kontrolü (MAC)
- Rol Tabanlı Erişim Kontrolü (RBAC)

2.3.1. İsteğe Bağlı Erişim (Discretionary Access Control - DAC)

Kurum bünyesinde bulunabilecek dosya sunucuları üzerinde erişim izinlerinin kontrol edilmesi gerekmektedir. Kullanıcıların girişlerinin nesneler üzerinde tanımlanmasına erişim kontrol girişi (access control entries) denir. Kullanıcıların kaynaklara erişimini kullanıcı kimliği ve nesne sahipliği ile kontrol eden birim isteğe bağlı erişim (DAC) denetimidir.

2.3.2. Zorunlu Erişim Kontrolü (Mandatory Access Control – MAC)

Erişim izinleri merkezi olarak kontrol edilir, yönetilebilir ve bütün erişim şekilleri sistem yöneticisi tarafından önceden tanımlanmaktadır. Sistem merkezi ve zorunlu erişime sahip olduğu için kullanıcıların izinlerini düzenleme yetkileri yoktur. Sistem içerisinde nesneler sınıflama etiketlerine (classification label) sahiptir.

2.3.3. Rol Tabanlı Erişim Kontrolü (Role Based Access Control – RBAC)

Kurum bünyesinde oluşturulan iş bölümlerine göre fonksiyonel erişimlerin verilmesidir. Grupların yönetimi üzerine odaklanan modelde verilere erişimi iş rolleri belirlemektedir.

3. VSE SİSTEMLERİNE GENEL BAKIŞ

Kurumların verilerini koruma ihtiyacı her geçen gün artmaktadır. Özellikle araştırma ve geliştirme yapan şirketler için geçerli olup, güçlü rekabetin nedeni olmaktadır. Kurumlar tarafından rekabetin sonucu olarak üretilecek teknolojinin korunması ve rakiplerinin veriye ulaşamaması hayati önem taşımaktadır.

Modern kurumsal bilgi koruması günümüzde meydan okumalarla karşı karşıyadır, mainframe bilgisayarın hâkim olduğu günlerden bugünlere hayal bile edilemeyen müthiş bir büyüme yaşanmaktadır. Basit bir dizüstü bilgisayarın ortalama veri depolama alanı bir terabyte düzeyine ulaşmakta ve anahtarlık büyüklüğündeki usb depolama ortamları ile kurumsal yapınızın veritabanlarını dahi saklamanız mümkün olmaktadır. Kullanıcılarınız taşınabilir cihazları ile herhangi bir ağa bağlanmakta, eposta veya sosyal paylaşım gibi çeşitli ortamlarda birçok hassas olabilecek bilgiyi taşıma gücüne sahip olmaktadır. Basit bir işlem ile kullanıcıları tamamı ile engelleyerek çözüm adıyla sunulacak çoğu uygulama, çözümsüzlük üretecektir. Bu nedenle yeni gereksinimlere ihtiyacımız vardır. Sistem kullanıcılarını daha az kısıtlayan, hassas olan bilgilerin akışını kontrol eden, sızdırma yapabilecek kullanıcıları engelleyen ve farkında olmadan yapılabilecek hataların en aza indirgenmesini sağlayacak çözümlere ihtiyaç duyulmaktadır. Kuruma zarar verebilecek eylemlerin engellemesi tercih edilmektedir. Bu bilgilerin korunması mobil, uyarlanabilir ve en az müdahaleye ihtiyaç duyan yapıda olmalıdır. Riskleri azaltmaya yardımcı olabilecek en iyi çözüm veri sızıntısı engelleme (VSE) olarak adlandırılmaktadır.

3.1. VSE Tanımlaması

Veri sızıntısı engelleme hassas verilerin kaybını önlemeye yardımcı olur (Abbadi, 2008). VSE son zamanlarda veri güvenliği konusunda geliştirilmiş raporlama, içerik inceleme, filtreleme ve hassas verilerin korunmasını sağlayan popüler sistemdir. VSE bilgi yaşam döngüsünde yaratma, dağıtım ve depolamayı içine alan koruma yöntemidir (Ubois, 2007). Veri kaynağı ile ilgili olup özel kullanıcı tanımını ortadan kaldırarak kurumsal yapının tamamını inceleyen sistemdir. Her organizasyonda bilginin hassasiyeti arttıkça, koruma mekanizmalarının da artışı beklenen bir sonuçtur. Bilginin akışı kontrol edilmek zorundadır ki, farkında olmadan da olsa organizasyon dışına çıkışı

engellenebilsin. Bu ihtiyaç modern işgücü tarafından farklı bir bakış açısı ile katlanarak büyümektedir.

Teknolojik anlayışlı çalışanlar, kurum içerisinde ve kurum dışında her ortamda webmail, sosyal paylaşım ortamları ve web tabanlı uygulamaları kullanarak üretkenliğini arttırmaya çalışmaktadır. Geleneksel veri sızıntısı engelleme kavramı uygun korunma ile içerik odaklı yaklaşım sunacaktır (Livingston, 2007). Günümüzün veri güvenliği ihlalleri geniş ortamları kapsayarak büyümeye devam etmektedir. Geçmiş yıllarda dışarıdan saldırılar, önemsiz e-postalar ve virüsler organizasyonları yoran en büyük nedenler arasında bulunmaktaydı. Fakat bilginin hassasiyeti fark edildikçe içeriden dışarıya sızıntıların önlenmesinin önemi artmıştır. VSE sistemleri, kredi kartı ve sosyal güvenlik numaraları olarak bilinen hassas bilgiler için ağ üzerinde arama yapmaktadır (Borders, 2009). Gizli bilgilere yapılacak temasların birçok kuruluş için bir numaralı tehdit unsuru olduğu günümüzde, özellikle yeni geliştirilen ürünler, finansal ve kişisel bilgiler çeşitli yasa ve yönetmeliklerle korunuyor olsa da veri sızıntısı gerçekleşmesi riskini taşımaktadır. Bilginin korunması ve kontrol edilmesi; tanımlama, izleme, şifreleme, filtreleme ve hassas verinin geçişine izin vermeme olanakları ile sağlanmaktadır. VSE yapısında kullanımdaki veri (data in use), dosya sunucuları ve veri tabanlarında ki veri (data at rest) ve ağ üzerinde kullanılan veri (data in motion) korunabilir kurumsal yapının içerisindedir.

3.2. VSE Büyüyen İhtiyaç

VSE, veri sızıntıları ile ilgili kötü bir tanıtım dahil, kurumsal düzenlemeler, yükselen kurumsal ürün maliyetleri ve bunların sızdırılmasını engelleyecek çeşitli nedenlerle ön plana çıkmaktadır. Veri kaybı engelleme sistem modülü paket analizi ve istemci yönetim ajanı için önemli fonksiyonlar sunmaktadır (Kim, 2010). Sızdırılabilecek veriler düşünüldüğünde, kişisel bilgiler, kurumsal arge bilgileri, yazışmalar, finansal bilgiler, yatırım bilgileri vb. en ağır darbenin alınacağı kollar arasındadır. Bu bilgiler müşteri kayıplarına, iş ve itibar kayıplarına neden olabilecek düzeydedir.

Bu aşamada bazı örneklemler aşağıda verilmiştir;

- Bir perakende şirketinin müşteri kredi kart bilgilerini yedeklediğini ve bu bilgilerin çalındığını,
- Kurumunuzdaki bir çalışanın sizin çok gizli ve gizli diye etiketlendirdiğiniz dokümanlarınızı dışarıya sızdırıldığını,
- Kurumsal ağda paylaştığınız sınav evrakı, kişisel raporlarınız ve proje belgelerinizin başka birinin eline geçebileceğini.

Bunları düşünmek belki kötü ama sonuçları teknolojinin sunduğu bir iyiliği tersine çevirebilecek düzeydedir. Bugün yurtdışında bazı şirket güvenlik yöneticisi olarak VSE ürünlerini kullanabilecek, güvenlik politikaları yaratacak ve yönetebilecek çalışanlar aramaktadır. Hükümetler her alanda kimlik ve veri hırsızlığından kişileri korumak için son yıllarda bazı standartlar geliştirmişlerdir. Bu düzenlemeler belirli kontrolleri, kurumsal uyum programlarını, denetimleri ve uyumsuzluğu cezalar ile engellemeye çalışmaktadır. Bunlardan bazıları sağlık alanında (HIPAA), veri koruma yasaları ve kredi kartları veri güvenliği (PCI-DSS) olarak sayılabilir. Kurumlar güvenlik stratejilerini geliştirmeli, korunacak öğeleri ve bilgiye erişim kurallarını belirlemeli aynı zamanda ağ üzerinde misafir kullanıcıların davranışlarını kontrol etmelidir.

3.3. Risklerin Tanımlanması

Bilgi koruma ve kontrolü kurumuzdaki hassas olabilecek verilerin tanımlanması ile başlamaktadır. Genel olarak veri sızıntısı engelleme sisteminde kullanımdaki veriler, durağan veriler ve ağ üzerinde hareket halindeki verilerin kontrolünü içermektedir.

Kurum içerisinde bilginin korunmasını zorlaştıran iç ve dış kaynaklı çeşitli tehditler mevcuttur. Bunlardan bazıları aşağıdaki gibidir:

- Çalışanların farkında olmadan gizli bilgilerin açığa çıkarılması,
- Trojan, virüs, solucan ve diğer zararlı kodlar,
- Spam,
- Çalışanlar ya da iş ortağı tarafından çalınan veriler,

- Hacker,
- Hatalı seçilen ağ ve güvenlik ekipmanları,
- Kurum içerisinden sabote,
- Spyware,
- Uygulamalardaki güvenlik açıkları,
- Kablosuz ve mobil internet.

Yukarıda sayılan unsurlardan %78 oranı ile veri hırsızlığı lider durumdadır [2]. Güvenlik teknolojilerinin kullanımı sayesinde güvenlik duvarı ve antivirüs gibi teknik araçları kullanarak riskler en aza indirgenmeye çalışılmaktadır. Yeni hedefler insanlar değil her zaman teknolojidir. Saldırgan birim sürekli erişim izinleri ile ilgilenmektedir. Bu nedenle gerekli teknik erişim kısıtlamaları değil, dosya sistem izinleri ya da bu izinleri aşmak için şifreleme gibi yollar tercih edilmektedir. Kullanıcıların tüm hareketlerini izleyebilmek için kontrol ajanına ihtiyacımız vardır (Redwood, 2011). Kullanıcılara seviyeli bir politika dili sağlanması için hassas veri koruması güvenilmeyen çıktı kanallarına (ağ iletişimi ve dosya paylaşımı gibi) uygulanmaz (Vachharajani, 2004). Ancak gizli verilere erişimin sürmesi güvenlik faktörlerine rağmen sürekli devam etmektedir. Yetkisiz durumlara karşı verilerin korunması veri sızıntısı engelleme sisteminin bir görevidir. Veri sızıntısı yazılımlarının kullanılmasından ziyade şu anda birçok şirket içerisinde güvenlik yazılımları sorunlu çalışmaktadır. Bu durumun bazı nedenleri vardır.

Bunlar:

- Standartların eksikliği,
- Bütçe limitleri,
- Eğitimli uzman eksikliği,
- Yasal engeller,
- İş dağılımı süreçlerinin eksikliği,

3.3.1. Kullanımdaki Veriler

Bu kısım ileriki bölümlerde anlatılacağı gibi uç nokta yönetimi (end point) olarak adlandırılmaktadır (Al-Fedaghi, 2011). İstemci taraflı ortamda bilginin güvenliğini korumak için çeşitli yöntemler içermektedir. Uç nokta yönetiminin sağlanması için gerekli olan ajan her istemcide kuruludur. Ağ yönetim sunucusundan aldığı çeşitli kurallara sahiptir (Luft, 2009). Burada ki amaç kullanıcının ağ çıkışı olmadan istemci üzerinden bilgi sızıntısı yapmasını engellemektir. Kurumsal bilgileriniz gizliliğini ve kullanma amaçlarını belirten eğitimlerin belirli zamanlarda kullanıcılara verilmesi ve kurumsal bilgi politikalarının anlatılması gerekmektedir. Sosyal mühendislik alanı bilgilerin kurallarını belirlemede en iyi alandır. Kurumsal kullanıcılar gelişen ve mevcut tehditlere karşı her zaman bilinçli tutulmalı ve sıklıkla eğitime alınmalıdır. Verinin kurum içerisindeki hareketini sağlayan kullanıcılar, kurumsal gizliliği ön planda tutmaya çalışmalıdır (Kurta, 2010).

3.3.2. Durağan Veriler

Durağan veriler depolama ortamlarındaki verilerdir. Kullanıcıların ve kurumsal bilgilerin yedeklendiği, veritabanı ve dosyaların yedeklendiği birimler olarak sayılabilmektedir. Bazı kurumlarda bu bilgiler çeşitli harici depolama birimleri, CD/DVD, usb sürücüler, yedekleme kasetleri veya arşiv ortamlarında saklanmaktadır (Kasparick, 2008). Durağan veriler VSE sistemlerinin bir ayağını oluşturmakta ve kurumların buradaki bilgileri sınıflandırması gerekmektedir. Veriler için yer önemli olmadığı için sabit depolama ortamı olmayan kurumlarda bu iş zorlu bir süreç anlamına gelmektedir. Çeşitli nedenlerden dolayı bir kurum tarafından verilerinizin istendiğini düşünürseniz, kurumsal verilerinizi sabit bir ortamda tutmanız daha sağlıklı olabilecektir. Kurum içerisinde saklanan verilerinizin neden önemli olduğunu bilmeniz gerekmektedir. Metin sınıflandırma, aynı grup içerisindeki benzer özelliklere sahip dokümanların özetlenmesi için kullanılmaktadır (Raman, 2011). Metin sınıflandırılması kurumunuza tecrübe ve zaman kazandıracaktır.

3.3.3. Hareket Halindeki Veriler

Hareket halindeki veriler yerel ağ üzerinden iletilen verileri ifade etmektedir. VSE ağ yönetim sunucusu ile hareket halindeki verinin içerik analizi yapılarak sızıntı tespiti sağlanmaktadır (Al-Fedaghi, 2011). Hareket halindeki verilerin ele geçirilmesi ve durdurulması ya da anlaşılması durağan verilerin aksine daha zor olmaktadır. Kullanıcının ağ üzerinden kurum dışına bilgi aktarımı esnasında güvenlik kurallarına takılarak veri iletiminin kesilmesi sağlanmaktadır. Kullanıcının hassas veri üzerindeki tüm hareketleri ağ akışı sırasında denetlenmektir (Luft, 2009). Okunabilir içerikteki tüm dokümanlar için bu işlem yapılabilmektedir. Bu durumdan habersiz kullanıcılar için önlem almak ve güvenlik prensipleri geliştirmek ise sistem yöneticisinin kontrolüne bırakılmıştır. Aslında bir sorunda misafir olarak ağa bağlanan (dizüstü, akıllı telefon vb.) kullanıcıların kontrolünde yaşanmaktadır. Bu kullanıcılarda kurum ağına dahil olacakları için bütün kurum politikaları bunlar içinde geçerli olacaktır.

3.4. Tehditlerin Belirlenmesi

Veri sızıntısının oluşması için birçok neden vardır, kurumsal çalışanlar en temel nedenler arasında sayılmaktadır. Her zaman güvenilir insan tanımlamasını VSE sistemlerinde kabul ederek yapılandırılma oluşturulmamalıdır. Bir dizi faktör vardır ki yetkili kullanıcılar tarafından veri ihlalleri sorununa katkıda bulunulmasın:

Bunlardan birincisi kurumsal e-postalardır. Osterman araştırma raporuna göre ortalama bir kullanıcı günde 44 mail göndermekte ve 123 mail almaktadır [3]. Bu rakamları yakalayan kurumsal bünyedeki tüm kullanıcıların sayısını almak ve bunu yıllık olarak hesaplamak büyük rakamlar yaratmaktadır. Bununla birlikte kullanıcıların sosyal paylaşım siteleri ve mesajlaşma ortamları bu rakamlara dahil değildir.

İkincisi taşınabilir cihazlardır. Teknolojinin sunduğu kablosuz internet ve ucuz taşınabilir cihazlar çalışanların mekan ve zaman bağımsız çalışmalarına neden olmaktadır. Kullanıcıların ortamlardan harici bellek, DVD veya dizüstü bilgisayarınıza kolaylıkla bilgi aktarımı mümkün kılınmaktadır.

Üçüncü olarak bilgiye erişim hakları konusunda kısıtlamaların yetersiz olması ya da yeni düzenlemelerin yeterince yapılmamasıdır. Deloitte araştırmasına göre finans

şirketlerinin çoğunda üst düzey güvenlik yöneticileri; erişim hakları denetimlerinin sorunlu çalıştığını belirtmiştir [4]. Dördüncüsü ise farkında olmadan yapılan hatalardır. Bugün çalışanların çoğu birçok işle uğraşan, sıkıştırılmış ve dikkati dağınıktır, bu yüzden güvenlik ve hassasiyet en son akıllarına gelen konu olmaktadır. Ne yazık ki kullanıcılar farkında olmadan veri ihlallerine neden olabilmektedir.

Çalışanlarınız arasında çok güvendiğiniz bir kullanıcınız ya da herhangi biri kurumunuza ihanet etmek için bekleyebilir. Özel veri içeren belgelerinize erişim hakkı olan kullanıcınız araştırmanızı rakibinize vermek isterse ya da kazanç elde edecekse, veri hırsızlığı onun için çok cazip olmaktadır. Etkili bir güvenlik politikasının geliştirilmesi ve uygulanması ile veri sızıntıları yok edilmeye başlanacaktır.

3.5. Bilginin Kurumsal Yapısı

Bir saldırganın hedefi kurum içerisinden veri çalmak ya da kaçırmaktır. Bu hedefe ulaşmak için veriyi birçok şekilde kurum dışarısına nasıl çıkarabileceği incelenmektedir. Kurum hassas verilerini kontrol etmek istediği için içerik taramasını düzenli olarak yapmalıdır (Mundada, 2011). BT güvenlik yöneticileri güvenli kanallar ile gizli veri çıkışını engellemek ve kullanıcıların hareketlerinin farkında olmadan onları izlemektedir. Korunma yöntemlerini uygulayabilmek için veri sızdırılma yöntemlerini ve hangilerinin daha çok tercih edildiğini bilmemiz gerekmektedir.

Bunlar aşağıdaki gibidir [4]:

- E-Posta (%15.84),
- Uç Nokta (%9.90),
- Http (%42.57),
- Internet Mesajlaşma Araçları (%0.99),
- Internet Mail (%4.95),
- Ağ Yazıcısı (%10.89),
- Web Posta (%2.97),

- USB Arayüzleri (%6.83),
- Diğer (%5.03).

3.5.1. Harici Depolama Ortamları

Harici depolama alanları ile başa çıkmak bir VSE sistem yeteneği olmalıdır ve "Uç nokta güvenliği" olarak bilinmektedir. Harici veri depolama ortamları nedeniyle veri kaybı ihtimali yüksek kabul edilmek zorundadır. Çünkü çok yüksek depolama kapasitelerinde kullanımı kolaydır. E-posta ve benzeri ortamlardan daha fazla alan sağlar ve işletme dışına ağ trafiğine takılmadan çıkışı kolaylaştıran elemandır. Bilgisayar bileşenlerine müdahale gerektirmeden veri kaybının yaşanacağı yer taşınabilir ortamlardır.

3.5.2. İnternet Tabanlı Yollar

İç ve dış veri kaynaklarına ek olarak internet tabanlı kaynaklar veri kaybının en çok yaşandığı yerdir. Düşük iletim hızı nedeniyle harici veri kaynaklarına göre büyük miktarda veri taşıma sınırlı olmaktadır. Bu durum sadece ilgili verinin azlığı ve sınırlı zaman diliminde çok büyük rol oynamaktadır.

3.6. VSE Politikalarının Önemi

Kurum içerisinde uygulanacak politikaların amacı veri sızıntısını engellemek ya da en aza indirgeyebilmektir. Sistem yöneticisi tarafından kurum için geliştirilen politikalar VSE sistemi ile yürürlüğe konulacaktır. Kurum içerisinde korunacak bilgiler belirlendikten sonra çeşitli çalışma bölümlerinde, hangi sınıf kullanıcıların, hangi verilere erişebileceğine karar vermek gerekmektedir. Hassas içeriğe sahip olan doküman erişimini tüm sınıflara açmak hatalı olacaktır. Kabul edilebilir kullanım politikası uygulayarak kurum içerisinde veri sızıntılarını engellemek için girişimler desteklenmelidir. Tabi ki VSE sistemi için katı kurallar koymaktan ziyade akışları kontrol etmek ve istenilen yerde müdahale etmek yararlı çözümdür. Kurumsal bilgilere, çalışanlar rahatça ulaşabildiğini fark etmeli, paylaşılabilir ortamı sunmanın sıkıntısını kullanıcılara hissettirmemek gereklidir.

Aşağıdaki sorulara yanıt vermek politikaların oluşmasında yardımcı olabilir:

- Kurum verileri ile ne yapabilirisiniz,
- Şirket verilerini aktarmak için hangi izinlere ihtiyacınız vardır,
- Bu izinleri vermek için yetkili kimdir,
- Bu bilgiler kime ya da hangi amaçlar ile aktarılabilir,
- Hangi bilgiler e-posta ile gönderilebilir.

Teknoloji kullanıcılar tarafından her zaman kötüye kullanma anlamına gelmemektedir. Nasıl ki antivirüs yazılımı kullanan bir kullanıcı tehdide maruz kalacak diye bir kural yoksa VSE çözümünü kullanan kurum içinde her zaman veri sızıntısına maruz kalacak diye bir yaklaşım söz konusu değildir.

VSE çözümünü uygulamak kurumsal farkındalığı arttıracak bir etmendir. Yaratılan politikalar ile kuruluşun genel hedefleri ve felsefesi eşleşmelidir. Kurumların VSE çözümlerinden beklentileri yüksektir ancak bu beklentiyi sadece doğru politikalar sağlayabilir. Politikaların kuralları net ve açık olmalıdır. Ayrıca kurumsal yapının tamamına hükmedebilecek kurallar dizisi oluşturulmalıdır.

3.7. Şirket Politikaları ve Örgütsel Önlemler

Açıklanan teknik güvenlik modellerini uygulamak, bilgi güvenliğini sağlamak için gereklidir. Teknik tedbirlerin yanı sıra şirket politikaları da tanımlanmalıdır. Kurumsal politikaları ilke veya kurala dâhil, şirket çalışanları için geçerli yapılar oluşturulur. Veri hırsızlığını ortadan kaldırmak için gerekli bir tedbir olarak, süreçlerde kullanıcılara tam yetki vermek ya da yasaklamak uygun görülmemiştir. Bu tür hükümler, örneğin bazı nesnelerin yasaklanması veya bilinmeyen dosyalar ile ilgili kullanım yönergeleri sunmaktadır. Politikaların tanımlanmasının yanı sıra, yararları pratikte kanıtlanmış, gizlilik ve veri güvenliği konusunda farkındalık eğitimi yürütmüş, çalışanlar eylemlerinin sonuçlarını açıkça ortaya koymaktadır. Kurumsal güvenlik politikalarını oluşturmak ve uygulamak için BT güvenlik yönetimi gereklidir. Bu uygulama veriyi etkin olarak korumak için ilk adımdır. Daha sonra açıklanacak ifadeler hangi önlemlerin uygulanacağını kararlaştırmaktır. BT yapısı olmadan güvenlik

yönetimi etkili bir VSE strateji tanımı başarısız olmaktadır. Ve örgütsel bir VSE sisteminin başarısı mümkün değildir. Kurumlar dış dünyanın ihtiyaç ve uyum kuralları için belirtilen süreçleri organizasyon içerisinde canlandırarak şirketin gizlilik politikalarını oluşturmaya çalışmaktadır. VSE sisteminin politikaları verimli şekilde oluşturmalı ve kullanıcı adımlarını sakince izlemeye çalışmalıdır.

3.8. Yasal Uygunluklar

Hükümetler hala veri gizliliği problemlerini çözmeye çalışmaktadır (Velte, 2010). Dünya çapında veri koruma yasalarını onaylanmakta ve çeşitli standartları kabul edilmektedir. Örnek olarak Amerika'da Sarbanes-Oxley yasası, Gramm-Leach Bliley yasası, HIPAA gibi veri koruma yasaları mevcuttur. Gizlilik ve veri koruma yasaları kurumsal verileri yetkisiz ve kötüye kullanımdan korumak için yürürlüğe alınmıştır. Hükümetlerin yanı sıra özel sektör tarafından da veri güvenliğini sağlamak için çeşitli standartlar geliştirilmiştir. Sektörel düzenleyici örgütlerin yaptırımları şirketleri veri güvenliğini sağlama konusunda mecbur bırakmıştır. Sağlık sektöründe HIPAA, finansta GLBA, BASEL II ve PCI Standards (Payment Card Industry) gibi regülasyon yaptırımları bilgi güvenliği ve dolayısıyla VSE konusunda ciddi adımların atılmasında önemli rol oynamaktadır. Hatta bunların tümünü geride bırakabilecek kadar kritik olan, Wall Street borsasında kâğıdı işlem gören her kurumun çok sıkı denetimlere maruz kaldığı Sarbanes – Oxley yasaları VSE konusunda en önemli yaptırımlardır (Lawrence, 2009).

4. VSE SİSTEMLERİNDE SIZINTI ENGELLEME

Veri sızıntısı engelleme sistemleri kritik bilgileri izlemek ve korumak için merkezi bir yönetim sistemi sunmaktadır. VSE sistemleri diğer güvenlik araçlarından farklı olarak veri ihlallerini önlemek için hassas veriye odaklanmaktadır (Al-Fedaghi, 2011). Bir VSE sistemi iki operasyonel sorun ile karşı karşıyadır: Performans ve doğruluk. VSE sistemleri binlerce veriyi durağan ortamda taramak, gerçek zamanlı ağ trafiğini izlemek ve uç nokta cihazlarında kullanıcı eylemlerini denetlemek zorundadır (Hart, 2011). Kurumsal ortamda teknik tedbirlerin yanı sıra gerekli örgütsel çözümler ile verilerin etkin bir şekilde yetkisiz kullanıma karşı korunması sadece yazılımdan ibaret değildir. Bir VSE konfigürasyonu ile sistem arasında, önceden tanımlanmış kurallar vardır.

Bir VSE sistemi veriler ile ilgili koruma kurallarının bütünüyle sınıflandırma işini yürütmektedir. VSE sistemi için kurallar BT güvenlik yöneticilileri tarafından tanımlanmaktadır. VSE yöneticisi tarafından düzenlenen kurallar sisteme uygulanmaktadır. BT altyapısı nedeniyle BT güvenlik yöneticileri bütünsel bir görünüm sergilemekte ve güvenlik gereksinimlerini uygulamaktadır, VSE yöneticisi sadece kendi alanındaki teknik koşullar ile ilgilenmektedir. VSE kurumsal sistemler aracılığıyla başlangıçta ve zaman içerisinde kuralları üretmekte veya değiştirebilmektedir.

4.1. VSE Çözümleri

VSE çözümleri üreten firmalar symantec, mcafee, promisec, citrix, mydlp, titus, frost gibidir. Yukarıda listelenen firmaların tamamı tecrübelerini çözümlerine yansıtmaktadır. Ancak bazıları tamamen sunucu taraflı, bazıları yazılım taraflı, bazıları hem sunucu hemde yazılım tabanlıdır. Üretilen çözümler hem ücret açısından yüksek, hemde çoğu kurumun problemini tam olarak çözemeyecek kadar komplekstir. Satılan ürünlerin ardındaki teknik destek zararları bazı durumlarda çok yüksek olmaktadır. Şu anda bu çözümleri kullanan çoğu kurum işin otomatik olarak yapılacağını zannetmiştir. Ama sonuç daha da baş ağrıtıcı olmuş ve şu anda Avrupa ve Amerika da firmalar VSE mühendisi çalıştırma yoluna girmeye başlamıştır. İnternet üzerinde birçok VSE mühendisi ilanı mevcuttur. Ve bu elemanı çalıştıracak firmalar bu pahalı yazılım ve donanımların tamamını kullanmalarına rağmen işleri daha da karmaşık hale gelmektedir. Bir VSE sistemi kurumsal veri kaybını önlemek için üç basamağı

gerçekleştirmektedir. Birincisi üç aşamalı sistem keşfi, depolama aygıtlarının taranması, gerçek zamanlı ağ trafiğinin engellenmemesi ve uç nokta yönetiminde son kullanıcının izlenmesi ile mümkündür. İkincisi, birinci basamakta keşfedilen hassas kurumsal verilerin tanımlanmasıdır. Üçüncüsü ise düzenlenen kurumsal politikaların hassas verilerin korunması için yürürlüğe koyulmasıdır (Hart, 2011). Tüm bu politikalar VSE sistemleri için birden fazla yazılım ve donanım ile birimler arası çok karmaşık yapılar sunmaktadır.

4.2. İçerik Tabanlı Sınıflandırma

İçerik tabanlı sınıflandırma aynı zamanda içerik denetimi olarak da adlandırılabilir. VSE sistemleri hassas verileri sınıflandırır, politikaları yönetir, veri akışını denetler ve sızıntıyı engeller (Al-Fedaghi, 2011). Veri sınıflandırma herhangi bir kurumun BT altyapısı içerisinde varlık yönetimi, risk değerlendirmesi ve güvenlik kontrollerinin stratejik kullanımı yapısı için zorunludur (Etges, 2006). BT yöneticisi sınıflandırma yapacağı zaman hangi bilgilerin korunacağını VSE sistemi arayüzü ile kolaylıkla yerine getirebilir. Bu işlem VSE tarafından desteklenen tüm dosya türlerini işlemek için kullanılabilir. VSE sistemi okunabilir dosya içerikleri için kullanılabilmektedir. Başarılı bir VSE sınıflandırıcısının çeşitli değerlendirme kriterlerini karşılaması gerekmektedir (Hart, 2011). Olası bir duruma örnek olarak dosya içeriğinde kredi kartı numarasının geçip geçmediğinin kontrol edilmesidir. Bu verinin belge içerisinde yer alıp almadığının kararını vermek için çeşitli yorumlama teknikleri kullanılabilir. İçerik tabanlı sisteme verilen değerler aracılığı ile doküman içeriği tanımlanarak sınıflandırma işlemi yapılmaktadır.

4.3. İstemci Tabanlı VSE Sistemi

Uç nokta yönetimi için istemci VSE çözümü kullanılmaktadır. Her istemci VSE sistemi için bir yazılım aracı (VSE ajan) veri izleme ile ilgili özel kurallar içermektedir. VSE ajan kuralları bir yönetim sunucusundan alır ve izin verilen ya da verilmeyen tanımlamaları yapmaktadır. Kullanıcı istenmeyen bir eylem gerçekleştirmek istediğinde, sistem doğrudan kullanıcı ile iletişime girmekte, örnek iletişim kutularını sergilemekte ya da işlemi engellemektedir. Veri sızıntısını engelleyecek geleneksel problemleri aşabilmek için Windows dosya sistemi filtreleme sürücüsü önerilebilir (Xiaosong,

2009). Windows dosya sistemi sürücüsü kullanılarak tüm dosya sisteminin kontrolü ve izlenmesi sağlanabilir. Bu konuda bell-lapadula ve bila modeline dayalı olarak zorunlu erişim kontrolü (mac) sunulmaktadır (Wang, 2006). Birçok teknoloji VSE için çalışmaktadır, bunlar kimlik doğrulama ve erişim kontrolü sayılabilir. İstemci tabanlı temel veri analizini kapsayan çözümler VSE sistemlerine uygulanabilir. Veri sızıntı engelleme sistemi Windows kayıt defterinin incelenmesini, olay günlükleri analizini, cd yazıcı olay günlüklerini ve dosya erişim analizini içine alarak bir çözüm sunmaktadır (Lee, 2009). İşletim sistemindeki çekirdek (kernel) modüllere VSE sistemi müdahale etmekte ve sistemin bütün faaliyetleri takip etmektedir. Bu şekilde VSE ajanı kullanıcının adımlarını ve eylemlerini izlemektedir. Bu işlemler uç nokta yönetim güvenliğini içermektedir, bu nedenle tüm yerel diskler izlenmiş olmaktadır. İstemci tabanlı VSE sistemleri düzenli ifadeler kullanarak hassas veriler izlenmesini sağlanmaktadır.

4.3.1. Ağ Tabanlı VSE Sistemi

VSE sistemleri IDS (intrusion detection system) e benzemektedir. Ancak tam tersi bir görev üstlenmektedir. IDS dışarıdan gelen saldırıları engellemek için çalışırken, VSE sistemi saldırganları durdurmak için çalışmaktadır. Her iki sistemde şüpheli modeller için ağ trafiği taranmakta, bu şekilde ağ bilgi akışı için izlenebilmektedir. Diğer tüm kanallarda gizli bir ağ tabanlı sistem vardır. Burada sistemin en zayıf tarafı, ağda veri hırsızlığı girişiminin sadece bir tarafı görülebilmektedir. Amaç, istenmeyen veri sızıntısını engellemek için ağ tabanlı sistemleri kullanarak, verilerin ağ üzerinde taşınması sırasında saptanmasından geçmektedir. VSE sistemleri ağa dâhil olan tüm cihazlarda paylaşılan verileri şifreleyebilir. Ağ geçidi tabanlı sistemler, hassas verileri engellemek için yetkisiz veri aktarımında giden trafiği (http, e-posta, mesajlaşma vb.) analiz etmektedir (Borders, 2009). Ağ üzerindeki akış kontrolünde verilerin şifrelenmemiş olduğu varsayılmıştır. Şifreli içeriğin izlenmesi mümkün değildir.

4.4. Kalıp Tepkileri

VSE sistemlerinde kullanılan teknolojiler kural ihlallerini belirlemek amaçlıdır. Reaksiyonlar (Extrusion Response) uygulamada yaygın olarak kural üreticisine bağlı olarak ayrı ayrı yapılandırılabilir.

Bunlar aşağıdaki gibidir:

- Loglama / Depolama,
- Uyarı başlangıcı,
- Uyarı tetikleme,
- Kullanıcıya uyarıyı gönderme,
- Engelleme.

VSE sistemi kullanıcılarına tüm serbest dolaşım fırsatlarını sunmaktadır. Loglama ve uyarı araçları bilgi sisteminde kural ihlali ile uygulanmaktadır. Uyarılar biriktikçe veya güçlü bir kural ihlali sonrasında, VSE sistemini takip etmek bir BT güvenlik yöneticisinin acil eylem planı olan alarmı tetiklemektedir. Bu uyarı günlük dosyası ya da anlık pencere gösterimi olabilir. Kullanıcıya kural ihlali haber verilmelidir, bir uyarı mesajı ekranda belirtilmeli veya uyarı kodu ile kullanıcılar bilgilendirilmelidir.

5. TEZDE KULLANILAN DİĞER YÖNTEMLER VE PROTOKOLLER

VSE sistemleri bütünleşik bilgi koruma sistemidir. Bu bölümde tezimde tasarlanan VSE mimarisinde kullanılan makine öğrenmesi yöntemlerinin ön tanımlı ifadeleri, ağ iletişimi ve ICAP protokollerinin açıklamaları anlatılmaktadır.

5.1. VSE Metin Sınıflandırma İçin Kullanılan Yöntemler

VSE son zamanlarda bilgi güvenliği konusunda geliştirilmiş raporlama, içerik inceleme, filtreleme ve hassas verilerin korunmasını sağlayan popüler bir sistemdir (Xiaosong, 2009). Bu tez çalışmanın konusu kurumsal ağ üzerinde hassas veri içeren dokümanların kurum dışına sızdırılmasını engellemek için VSE mimarisi tasarlamak ve uygulamaktır. Ağ üzerinde akan verinin sınıflandırılması ve kontrolü el ile yapılamayacağından bu işlemi otomatik olarak yapan bir sistemin geliştirilmesi gerekmektedir. Otomatik sınıflandırma son kullanıcı müdahalesi olmaksızın yapılmakta ve arka planda çalışmaktadır. Burada sistem yöneticileri, sınıflandırma kurallarını oluşturmaktadır. Dosya sisteminde çeşitli teknikler kullanılarak veriler sınıflandırmaya çalışılmaktadır. Sınıflandırmanın bazı avantajları ve dezavantajları vardır. Dezavantajı sınıflandırma işlemine müdahale edilememesidir. Avantajı ise kullanıcının kasten sınıflandırılmamış ya da hatalı sınıflandırılmış veri bırakmasını engellemektedir. Elle sınıflandırma verileri tek tek, uzun zamanda ve en az dosya üzerinde işlem sonucuna ulaşarak sınıflandırabilmektedir. Bu yüzden kontrolü kurumsal ağ üzerinde bu şekilde yapmak zor olacaktır. VSE sisteminin tanıtımı ve başlangıcında veriler otomatik olarak sınıflandırılmalıdır. Ve bu sistemde gerçekleştirilen sınıflandırma içerik tabanlıdır. Dosya türlerini ve içeriklerini sınıflandırmak için karar verme işlemidir. Örneğin bir dosya sunucusu ya da istemci üzerinden usb belleğe ya da ağ üzerine dosya aktarımı yapılacak olduğunda alternatif olarak VSE sistemi bu dosyanın içeriğine göre sınıflandırma yapmak zorundadır. Sistem bu belgeyi ya normal olarak etiketleyecek ya da hassas veri sınıfına koyacaktır. Teknik açıdan bir VSE sistemi yetkisiz kişiler tarafından kaldırılamayacak sınıflandırma işlemini yerine getirmektedir. Sınıflandırma ataması yapıldığı için VSE verileri daha güvenli bir şekilde korunabilmektedir. VSE sisteminin metinlerinde sayısal formata dönüşümü sağlamak amacıyla kullanılan Text2arff yazılımının özellik çıkarım teknikleri aşağıda anlatılmıştır (Amasyalı, 2010).

5.1.1. Naïve Bayes

Sınıflandırma yöntemlerinden Naïve Bayes metin sınıflandırmada başarısı kanıtlanmış popüler bir makine öğrenmesi tekniğidir (Domingos, 1997). Naïve Bayes algoritması, bir dokümanın içindeki özellikleri birbirinden bağımsız olarak düşünerek kelimelerin ve sınıfların birleşik olasılıkları ile bir dokümanın sınıfının belirlenmesinde kullanılmaktadır (Adsız, 2006). Hedef fonksiyon için bir eğitim seti hazırlanır. <a1, a2,…an> şeklindeki özellik vektörüyle ifade edilen yeni bir örnek alınmakta ve eğiticiden bu örneğin hedef değerini yani ait olduğu sınıfı tahmin etmesi istenmektedir (Domingos, 1997). NaïveBayes olasılıklı bir sınıflandırıcıdır ve verilen bir dokümanı sınıflandırmak için kelimelerin ve sınıfların olasılıklarını kullanmaktadır (Náther, 2005). Özellik vektörü (x1,…,xn) olan bir X örneği verildiğinde, NaïveBayes sınıflandırıcısı denklemini kullanarak benzerliği en yüksek yapan bir C sınıf etiketi aramaktadır (5.1) (Jackson, 2002).

$$P(x_1,..., x_n|C) = \alpha P(C)\prod_i P(x_i|C) \tag{5.1}$$

5.1.2. Karar Ağaçları

Karar Ağaçları (J48) makine öğrenme tekniğidir, bağımlı ve bağımsız özelliklerin kümesini gösterir, ağaç şeklinde yapıyla karar kümelerini oluşturur (Witten,2005). Tahmin edici ve tanımlayıcı özelliklere sahip olan karar ağaçları akış şemalarına benzeyen yapılardır. Her bir nitelik bir düğüm tarafından temsil edilir. En üstteki düğüm ise kök düğüm (root node) olarak adlandırılır (Alpaydın, 2010). Bir karar ağacı kurarken, kazanç hesaplamasıyla eğitim kümesindeki değerler bilinmeden sadece nitelikler bilinerek işlem yapılmaktadır. Bir karar ağacı kullanımında bilinmeyen nitelik değerlerine sahip olan kayıtlar mümkün olan sonuçların olasılıklarını tahmin ederek sınıflanabilir. J48 bilgi kazancı ile beraber alt sette yer alan karakteristiklerin bilgi kazanç oranlarını da hesaplayarak bilgi kazanç oranı en yüksek olan karakteristiği düğüm noktası olarak seçmektedir. Karar ağacının her dalı sadece bir tek sınıfa karşılık gelinceye kadar işlemleri sürdürmektedir. Weka yazılımı J48 sınıfı ile C4.5 karar ağacı uygulamasını sunmaktadır.

J48 algoritmasının tercih edilme nedenleri;

- Eksik nitelik değerlerini ele alabilmektedir,
- Karar ağacının budanması ve kural çıkarımı gibi işlemleri yapabilmektedir,
- Yeni durumların sınıflamasını doğru yapabilmektedir.

5.1.3. Destek Vektör Makinesi

Destek Vektör Makinesi (SVM) yöntemi, Vapnik ve arkadaşları tarafından geliştirilmiş etkili bir öğrenme yöntemidir (Vapnik, 1995). Öğrenme, sınıflandırma, kümeleme, yoğunluk tahmini ve veriden regresyon kuralları üretmek için kullanılan bir eğitme algoritmasıdır. Bu teori, yapısal risk minimizasyonu prensibine dayanmaktadır. SVM, iki sınıfa ait elemanların oluşturduğu eğitim kümelerini en uygun düzlem ile birbirlerinden ayırmaya çalışan bir yöntemdir. SVM yaklaşımı, yüksek boyutlu verilerin olduğu durumlarda dahi güçlü bir genelleme özelliğine sahip olup tahmine dayalı bir model geliştirebilir. SVM'nin amacı bir test kümesinin içinde sadece nitelikleri verilmiş olan veri örneklerinin hedef değerlerini tahmin eden bir model oluşturmaktır (Hsu, 2009). SVM düşük boyutta ayrıştırılamayan bir veri dizisini hiperdüzlem kullanarak daha büyük bir boyuta ayrıştırmaya yaramaktadır.

5.1.4. K-En Yakın Komşu Algoritması

K-en yakın komşuluk algoritması (IBK) metin sınıflandırmada kullanılan en popüler yöntemlerden birisidir (Manning, 1999). K en yakın komşuluk yönteminde en yakın k adet örnek bulunur ve bu örnekler en çok hangi sınıftan iseler, test örneği de o sınıfa aittir denir. Bu metotta, test örneğinin, eğitim kümesindeki bütün örneklere olan yakınlığı hesaplanmaktadır. K en yakın komşuluk algoritması sorgu vektörünün en yakın k komşuluktaki vektör ile sınıflandırılmasının bir sonucu olan denetlemeli öğrenme algoritmasıdır. Bu algoritma ile yeni bir vektörü sınıflandırabilmek için doküman vektörü ve eğitim dokümanları vektörleri kullanılmaktadır. K-en yakın komşu algoritması için uygun k değerinin belirlenmesi çok önemlidir; çünkü k değerine göre sorgu noktasının hangi sınıfa ait olacağı değişebilir. (Kocamaz, 2011).

5.1.5. Kelime Kümeleme Algoritması

Özellik çıkarım tekniklerinden kelime kümeleme algoritması (K-Means) veri dağılımını en iyi temsil edebilecek küme merkezlerinin bulunması fikrine dayanmaktadır (Güven, 2007). En eski kümeleme metotlarından biri olan kelime kümeleme algoritmasının genel mantığı n adet veri nesnesinden oluşan bir veri setini giriş parametresi olarak verilen k adet kümeye bölümlemektir. Amaç, gerçekleştirilen bölümleme işlemi sonunda elde edilen kümelerin küme içi benzerliklerinin maksimum ve kümeler arası benzerliklerinin minimum olmasını sağlamaktır. Küme benzerliği, kümenin ağırlık merkezi olarak kabul edilen bir nesne ile kümedeki diğer nesneler arasındaki uzaklıkların ortalama değeri ile ölçülmektedir. Nesnenin atandığı kümenin saptanmasında nesnenin en yakın veya benzer olduğu küme merkezi dikkate alınmaktadır (Dinçer, 2006).

5.1.6. N-Gram

N-gram yöntemi bir metnin içerisindeki kelimeleri oluşturan harflerin yan yana gelme örüntülerine bakarak metnin hangi amaç için yazıldığını bilgisayar tarafından belirleyebilmek amacıyla kullanılmaktadır (Güven, 2007). N-gram yöntemi metin içerisindeki karakter tabanlı n-gramların kullanım sıklığına dayalı bir işlemdir (Doğan, 2010). N-gram, bir karakter katarının n adet karakter dilimidir. N-gram tabanlı sınıflandırma yöntemi, doküman içerisindeki n-gram karakterlerin kullanım sıklığına dayalı bir işlemdir. Tezde geliştirilen VSE sistem mimarisinde, n-gram'ın farklı birkaç uzunluğu olarak 2-gram'lar kullanılmıştır. N-gram yöntemi, dokümanları sınıflandırmak için kullanılan basit ve güvenilebilir bir yöntemdir. N-gram frekans yaklaşımı dilden bağımsız çalışmaktadır. Tüm harflerin veya hecelerin istatistiklerini kullanarak benzer sonuçlara ulaşmak mümkündür.

5.1.7. Kelime Kökleri

Kelime kökleri (Words) metinlerin ifade edilmesinde en yaygın kullanılan özellik grubudur. Metinlerin içerdikleri kelimelerle ifade edilirler. Metinleri gösteren vektörlerin boyutu, tüm metinlerde en az bir kere geçen farklı kelime sayısıdır. Kelimelerin kendilerini kullanmak Türkçe gibi eklemeli dillerde farklı kelime sayısını çok fazla arttırmaktadır.

Ağaç, ağaçlı, ağaçlandırma ya da visiting, visited kelimeleri farklı kelimeler ve dolayısıyla farklı özellikler olarak ele alınması buna örnek verilebilir. Probleme çözüm olarak literatürde kelimelerin kendileri yerine sadece kelime kullanılması önerilmektedir. Bu sayede hem aynı anlama işaret eden kelimelerin birleştirilmesi (böylelikle metinler arası benzerliğin daha iyi ifade edilmesi sağlanmaktadır), hem de özellik boyutunun azaltılmasıyla işlem karmaşıklığının ve gürültü veri olasılığının azaltılması sağlanmaktadır (Amasyalı, 2010).

5.1.8. Ağırlıklandırma Yöntemleri

Bir kelimenin bir doküman içerisinde ne kadar sıklıkta geçtiği o dokümanın hangi kategoriye atanması gerektiği konusunda önemli bir ipucudur. Ayrıca bir kelimenin çok fazla doküman içerisinde yer alması kelimenin ayrım gücünü düşürdüğünden dolayı metnin uygun kategoriye atanmasında bu kelimenin kullanılması uygun değildir (Adsız, 2006). Bu iki önemli durum göz önüne alındığında metnin sınıflandırılmasında ağırlıklandırmanın son derece önemli olduğu açıkça görülmektedir. Ağırlıklandırma, bir kelimenin bir dokümanda ne sıklıkla kullanıldığını gösteren sayısal değerdir. Literatürde yaygın olarak kullanılan ağırlıklandırma yöntemleri şunlardır:

- Frekansa göre ağırlıklandırma(tf)
- Terim frekansı x Ters doküman frekansı(tfxidf)

5.1.8.1. Frekansa Göre Ağırlıklandırma

Kelimelerin metinde kaç defa kullanıldığını gösteren oldukça basit bir yöntemdir (İlhan, 2008). Bu yöntemde ağırlık kelimenin doküman içerisindeki ham frekansına eşitlenmektedir (Adsız, 2006).

5.1.8.2. Terim Frekansı x Ters Doküman Frekansı

Tfxidf, dokümanları vektör uzay modelinde tanımlayabilmemiz için kullanılan en önemli ağırlıklandırma metotlarından biridir. Metin kategorizasyonunda tfxidf ağırlıklandırma metodu iki önemli öğrenme metodu ile ilişkilidir. Bunlar KNN ve DVM'dir. Tfxidf ağırlıklandırmasında her bir dokümandaki kelimelerin frekansı rol oynamaktadır. Böylece dokümanda daha fazla görülen kelimeler varsa (TF, terim

frekansı yüksek) o doküman için daha değerli olduğu anlaşılmaktadır. Ayrıca IDF tüm dokümanlarda seyrek görülen kelimeler ile ilgili bir ölçü vermektedir. Bu değer tüm eğitim dokümanlarından hesaplanmaktadır. Bu yüzden eğer bir kelime dokümanlarda sık geçiyorsa doküman için belirleyici olmadığı düşünülebilir (Adsız, 2006). Eğer kelime dokümanlarda çok sık geçmiyorsa o kelimenin o doküman için belirleyici özelliği vardır denilebilir. Tfxidf genel olarak sorgu vektörü ile eğitim dokümanı vektörü arasındaki benzerlik oranını bulmak için kullanılır. Tfxidf fonksiyonunun çeşitli versiyonları mevcuttur (Soucy, 2005). Ters belge frekansının temelindeki fikir, "bir terimin, derlem içinde gözlendiği belge sayısı azaldıkça, gözlendiği belgeler açısından ayırt ediciliğinin artacağı" şeklindedir.

5.2. Ağ Tabanlı VSE Sisteminde Kullanılan Protokoller

Bu tezde gerçekleştirilen VSE mimarisinin en önemli adımlarından biri ağa dahil olan tüm istemcilerin trafik akışını kontrol etmektir. Ağ tabanlı VSE sistemi ICAP / HTTP / HTTPS / TSL protokollerini içermektedir. Ayrıntıları aşağıda anlatılmaktadır.

5.3. HTTP Protokolü

HTTP (*Hypertext Transfer Protocol – Hipermetin Aktarma İletişim Kuralı*) bir kaynaktan dağıtılan ve ortak kullanıma açık olan hiperortam bilgi sistemleri için uygulama seviyesinde bir iletişim kuralıdır [5].

1990 yılından itibaren Web üzerinde kullanılmakta olan HTTP 'nin şuan ki sürümü 1.1 dir ve RFC 2616 [6] ile tanımlanmıştır. Aşağıda http ile ilgili kavramlar açıklanmaktadır.

5.3.1. URI (Uniform Resources Identifier)

URI'ler birçok isimle bilinirler: WWW adresleri, Evrensel Doküman Tanımlayıcıları, Evrensel Kaynak Tanımlayıcıları ve son olarak Uniform Kaynak Bulucuları (URL) ve İsimleri'nin (URN) kombinasyonu. HTTP açısından URI'ler bir kaynağı tanımlayan basitçe formatlanmış dizgelerden ibarettir. HTTP URL'leri şema adı (http), host, port, mutlak dosya yolu ve sorgu bilgilerini içerebilmektedir. Bu bilgilerden ilk ikisi zorunludur. Port belirtilmezse varsayılan port olan 80 kabul edilmektedir.



Şekil 5.1. Örnek bir URL

Yukarıda ki şekilde http:// şema adı olup, "Servis Adı", "Alan Adı", "Alan Adı Uzantısı" ve "Bölge Uzantısı" kısımları bilgi amaçlı yazılmaktadır. Adı geçen son dört bölüm birlikte URI'nin alan adı kısmını oluşturmaktadırlar.

5.3.2. Varlık

HTTP istek ve cevapları metotları tarafından sınırlanmadıkları sürece bir varlık taşıyabilirler. Bir varlık, varlık başlıkları ve varlık gövdesinden oluşmaktadır. Varlık gövdesinin olması, mesaj gövdesinin olmasına bağlıdır. Varlık gövdesi, mesaj gövdesine uygulanan transfer kodlamalarının çözülmesinden sonra ortaya çıkan veridir.

5.3.3. İçerik Kodlamaları

İçerik kodlama değerleri taşınan varlığa uygulanabilir ya da uygulanmış bir kodlamayı belirtilmektedir. Başlıca kullanım amaçları dokümanın sıkıştırılması ya da altında yatan medya türünün kimliğini kaybetmeden ve bilgi kaybı olmadan dönüşümüdür. Genellikle varlık kodlanmış şekilde saklanır, doğrudan iletilir ve yalnızca alıcı tarafından kodu çözülmektedir.

HTTP/1.1 içerik kodlama değerlerini Accept-Encoding ve Content-Encoding başlıkları içinde kullanılmaktadır. Değer içerik kodlamasını tanımlasa da asıl önemli olan gerekli kod çözme mekanizmasını belirtmesidir.

IANA (The Internet Assigned Numbers Authority), içerik kodlama değerleri için bir kayıt defteri gibi davranmaktadır.

5.3.4. Transfer Kodlamaları

Transfer kodlaması değerleri, ağ üzerinden güvenli taşımayı sağlamak için bir varlık gövdesine uygulanmış, uygulanabilir ya da uygulanması gereken kodlama

dönüşümleridir. Transfer kodlamaları, içerik kodlamalarının aksine, varlığın değil HTTP mesajının bir özelliğidir.

HTTP/1.1 transfer kodlaması değerlerini TE ve Transfer-Encoding başlıklarında kullanılmaktadır.

IANA, transfer kodlamaları için bir kayıt defteri olarak davranır ve başlangıçta şu değerleri içermektedir: “chunked”, “identity”, “gzip”, “compress”, “deflate”.

Parçacıklı Transfer Kodlaması kullanılan örnek bir HTTP mesajı aşağıda verilmiştir:

```
HTTP/1.1 200 OK

Content-Type: text/plain

Transfer-Encoding: chunked

1E

Bu mesaj birden fazla parçadan

D

oluşmaktadır.

0
```

Şekil 5.2. Parçacıklı Transfer Kodlaması Kullanan http Mesajı

Görüldüğü gibi her parçanın önünde 16 lık (hexadecimal) sayı sistemi kullanarak kodlanmış bir boyut belirtici bulunmaktadır. Boyutu 0 olan bir parça ile de transfer bitirilmektedir.

5.3.5. Medya Türleri

MIME Türleri olarak da bilinen bu türler, ilk başta SMTP prokolüyle kullanılmak için tanımlanmış olup, günümüzde HTTP, RTP, SIP gibi çeşitli

protokollere de geçmiştir. HTTP medya türlerini Content-Type ve Accept başlıklarında kullanılmaktadır.

Medya türleri tür ve alt tür olmak üzere iki bölümden oluşurlar. IANA (Internet Assigned Number Authority) medya türleri için de bir kayıt defteri olarak davranmaktadır. Kaydedilmemiş türlerin kullanılması tavsiye edilmemektedir. Örnek medya türleri: application/x-gzip, application/pdf, audio/mp4, image/png, multipart/form-data. Tam liste için RFC 1700 [10] incelenebilir.

MIME, bir dizi "multipart" (çok parçalı) tür sağlar, böylece bir ya da daha fazla varlık tek bir mesaj gövdesi içinde kapsüllenir.

5.3.6. HTTP Mesajları

HTTP mesajları, istemciden sunucuya gönderilen istekler ve sunucudan istemciye dönen cevaplardan oluşmaktadır.

İstek ve cevap mesajları RFC 822 [13] ile tanımlanan jenerik mesaj formatını kullanılmaktadır. Her iki tür mesaj da bir başlangıç satırı, sıfır ya da daha fazla başlık alanı ve opsiyonel olarak bir mesaj gövdesinden oluşmaktadır.

Mesaj başlıkları: İletilen mesaj ile ilgili fakat içeriğe dahil olmayan bilgileri sağlamaktadır. Tüm başlıklar bir isim ardından ":" karakteri ve alan değerinden oluşmaktadır.

5.3.7. HTTP İstekleri

İstemciden sunucuya giden; bir istek satırı, başlıklar ve opsiyonel olarak bir mesaj gövdesinden oluşmaktadır.

- İstek Satırı: Metot, İstek URI ve HTTP sürümünden oluşan bir satırlık dizgedir.

-- Metot: İstek sırasında kullanılan HTTP metodunu belirtir. OPTIONS, GET, HEAD, POST, PUT DELETE, TRACE, CONNECT metotları HTTP/1.1 de ön tanımlı metotlardır.

Bir kaynak tarafından izin verilen metotların listesi Allow başlığında belirtilir. GET ve HEAD metotlarının tüm sunucular tarafından uygulanması zorunludur, diğer metotlar opsiyoneldir.

-- İstek URI: İsteğin uygulanacağı kaynağı tanımlayan bir URI den ibarettir. İstek URI 'nin "*" olması isteğin spesifik bir kaynağa uygulanması gerekmediği anlamını taşımaktadır.

5.3.8. HTTP Cevapları

Bir sunucu bir HTTP isteğini alıp yorumladıktan sonra bir HTTP cevabıyla istemciye cevap vermektedir. Durum satırı, başlıklar ve opsiyonel olarak bir mesaj gövdesinden oluşmaktadır.

- Durum Satırı: Protokol sürümü, durum kodu ve durum kodunu açıklayan kısa bir cümleden (sebep metni) oluşan bir satırlık dizgedir.

-- Durum Kodu ve Sebep Metni: Durum kodu 3 basamaklı bir sayı olup, isteği yorumlama işleminin bir sonucudur. Sebep metni ise durum kodunu açıklamayı amaçlayan kısa bir yazıdır.

Durum kodunun ilk basamağı cevabın sınıfını belirlemektedir.

1xx: Bilgi

2xx: Başarı

3xx: Yönlendirme

4xx: İstemci Hatası

5xx: Sunucu Hatası

5.3.9. Kalıcı Bağlantılar

Kalıcı bağlantılar, aynı bağlantının birden fazla istek-cevap çifti için kullanılmasını ifade etmektedir.

Kalıcı bağlantılardan önce tüm URI lere ulaşmak için ayrı bir TCP bağlantısı kullanılmaktaydı. Resimlerin ve diğer bazı verilerin kullanımı da genellikle bir

istemcinin kısa bir süre içerisinde aynı sunucuya birden fazla istek göndermesine neden olmaktaydı. Bu da HTTP sunucularında ki yükü artırıp internette bir tıkanıklık oluşturmaktaydı. HTTP/1.1 de aksi belirtilmediği sürece tüm bağlantılar, kalıcı bağlantı olarak kabul edilmektedir.

Kalıcı bağlantıların avantajlarından bazıları:

- Daha az sayıda TCP bağlantıları açıp kapamak ile işlemci zamanından kazanç,
- TCP bağlantılarının açılması için gerekli paketlere gerek kalmadığından ağ yükü azalmaktadır,
- TCP bağlantısı açarken yapılan "el sıkışma" işlemine zaman harcanmadığından ardışık istekler arasındaki gecikme azalmaktadır.

5.4. ICAP Protokolü

ICAP (*Internet Content Adaptation Protocol – İnternet İçerik Uyarlama Protokolü*), RFC 3507 [14] ile tanımlanmış, ulaşım protokolü olarak TCP kullanan uygulama seviyesinde bir protokoldür. ICAP Protokolü, ön bellekli bir vekil sunucu yardımıyla HTTP trafiği üzerinde ek servislerin uygulanmasını sağlamaktadır. Yapısı açısından HTTP protokolü ile benzerlik göstermesine rağmen HTTP üzerine kurulu değildir.

ICAP protokolü günümüzde web tabanlı anti-virüs sistemleri, içerik filtreleme sistemleri ve veri sızıntısını engelleme çözümleri gibi genelde güvenlik sistemlerinde kullanılmaktadır. Mevcut sürümü 1.0 olan protokol, şu an için HTTP mesajları üzerinde uyarlama yapabilmektedir.

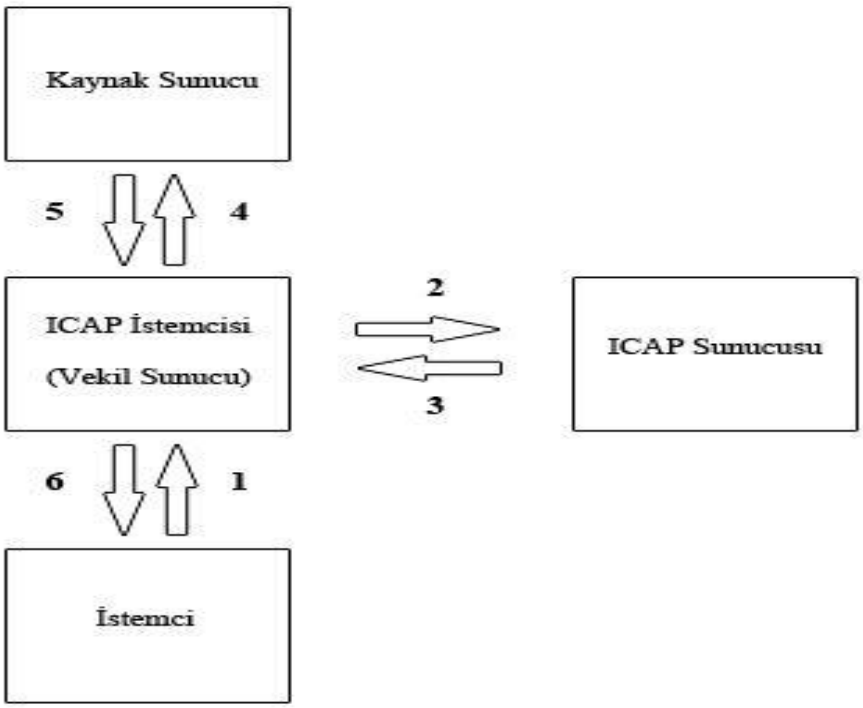
ICAP kullanılarak içerik uyarlama işlevinin gerçekleştirilmesi için bir istemci, bir ICAP istemcisi (örneğin: ön bellekli bir vekil sunucu) ve bir ICAP sunucusu gereklidir.

Temel ICAP fonksiyonları REQMOD (*Request Modification Mode – İstek Uyarlama Modu*) ve RESPMOD (*Response Modification Mode – Cevap Uyarlama Modu*) olmak üzere 2 adettir.

5.4.1. REQMOD

Bu modda HTTP isteği ICAP istemcisi tarafından ICAP sunucusuna iletilmektedir. ICAP sunucusu bu durumda 3 farklı şekilde davranabilir:

1. HTTP isteğinin uyarlanmış halini geri döndürebilir,
2. İsteğe karşılık bir HTTP cevabı döndürebilir,
3. Bir hata döndürebilir.



Şekil 5.3. REQMOD İçin Tipik Veri Akışı

1. Bir istemci ICAP destekleyen bir vekil sunucuya kaynak sunucudaki bir nesne için istek göndermektedir,
2. Vekil sunucu bu isteği ICAP sunucusuna göndermektedir,
3. ICAP sunucusu, ilgili servisini istek mesajı üzerinde çalıştırır ve duruma göre isteğin uyarlanmış halini ya da isteğe karşılık bir cevap göndermektedir,

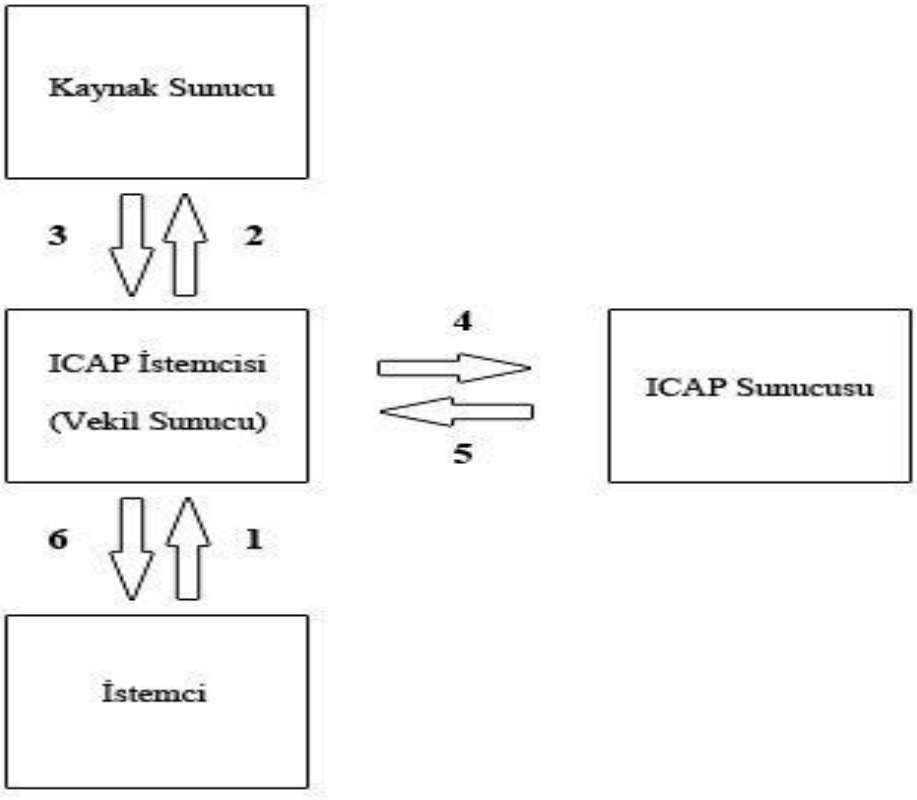
Eğer 3. adımdan bir istek geri dönerse:

4. Vekil sunucu, -uyarlanmış olma ihtimali bulunan- bu isteği kaynak sunucuya göndermektedir,
5. Kaynak sunucu isteği cevaplamaktadır,
6. Vekil sunucu ICAP sunucusundan ya da kaynak sunucudan gelmiş olan bu cevabı istemciye göndermektedir.

5.4.2. RESPMOD

Bu modda kaynak sunucudan dönen HTTP cevabı ICAP istemcisi tarafından ICAP sunucusuna iletilmektedir. ICAP sunucusu bu durumda 2 farklı şekilde davranabilir:

- HTTP cevabının uyarlanmış halini geri döndürebilir,
- Hata vererek dönebilir.



Şekil 5.4. RESPMOD İçin Tipik Veri Akışı

1. Bir istemci ICAP destekleyen bir vekil sunucuya kaynak sunucudaki bir nesne için istek göndermektedir,
2. Vekil sunucu bu isteği kaynak sunucuya göndermektedir,
3. Kaynak sunucu isteği cevaplamaktadır,
4. Vekil sunucu kaynak sunucudan gelen cevabı ICAP sunucusuna göndermektedir,
5. ICAP sunucusu ilgili servisini cevap mesajı üzerinde çalıştırır ve uyarlanmış olma ihtimali bulunan cevabı vekil sunucuya göndermektedir,
6. Uyarlanmış olma ihtimali bulunan bu cevap vekil sunucu tarafından istemciye göndermektedir.

5.4.3. HTTP Mesajlarının Kapsüllenmesi

ICAP kapsülleme modeli, HTTP mesajlarının ICAP sunucusu ve istemcisi arasında iletimi için geliştirilmiş hafif bir paketleme modelidir.

Bu işlem, HTTP mesajlarının belirli bölümlerinin tek bir ICAP mesajının gövdesi olacak şekilde birleştirilmesiyle gerçekleştirilir. Kapsüllenen bölümler HTTP mesajlarının başlıkları ya da gövdesi olabilmektedir.

Kapsüllenen gövdeler HTTP 1.1 ile belirtilen “Chunked Transfer Encoding” (Parçacıklı Transfer Kodlaması) ile iletilmelidir. Bunun aksine kapsüllenen HTTP başlıklarında kesinlikle bu kodlama kullanılmamalıdır. Bir HTTP mesajının kapsüllenen bölümlerinin başlangıç ofsetleri “Encapsulated” başlığında belirtilir.

Encapsulated: req-hdr=0, res-hdr=36, res-body=90

Yukarıdaki başlık alanı bir grup istek başlığını, bir grup cevap başlığını ve bir cevap gövdesini kapsülleyen bir mesajı belirtir. Tüm bunlar belirtilen byte ofsetlerinde başlamaktadır. Bu ofsetler 10 luk sayı sistemine göre belirtilir.

İstek ve cevap gövdesi içermeyen ICAP mesajlarında Encapsulated başlığında gövde “null-body=Ofset” şeklinde belirtilebilir.

5.4.4. Örnekler

Bu bölümde Reqmod ve Respmod fonksiyonları için örnek birer ICAP istek-cevap çifti verilecektir.

ICAP Reqmod Örneği:

Aşağıdaki örnekte bir vekil sunucu istemciden gelen basit bir GET isteğini alır. Bir ICAP istemcisi olarak hareket eden vekil sunucu, bu isteği düzenlenmesi için ICAP sunucusuna gönderir. ICAP sunucusu isteği düzenler ve düzenlenmiş halini vekil sunucuya geri gönderir. Bu örnekte kaynak sunucu adresi “www.origin-server.com”, ICAP sunucusunun adresi ise “icap-server.net” dir.

```
REQMOD icap://icap-server.net/server?arg=87 ICAP/1.0
Host: icap-server.net
Encapsulated: req-hdr=0, null-body=170


GET / HTTP/1.1
Host: www.origin-server.com
Accept: text/html, text/plain
Accept-Encoding: compress
Cookie: ff39fk3jur@4ii0e02i
If-None-Match: "xyzzy", "r2d2xxxx"
```

Şekil 5.5. ICAP Reqmod İstek Mesajı

```
ICAP/1.0 200 OK
```

```
Date: Mon, 10 Jan 2000  09:55:21 GMT

Server: ICAP-Server-Software/1.0

Connection: close

ISTag: "W3E4R7U9-L2E4-2"

Encapsulated: req-hdr=0, null-body=231



GET /modified-path HTTP/1.1

Host: www.origin-server.com

Via: 1.0 icap-server.net (ICAP Example ReqMod Service 1.1)

Accept: text/html, text/plain, image/gif

Accept-Encoding: gzip, compress

If-None-Match: "xyzzy", "r2d2xxxx"
```

Şekil 5.6. ICAP Reqmod Cevap Mesajı

RESPMOD Örneği:

Aşağıdaki örnekte, bir ICAP istemcisi, istemcinin -vekil sunucu aracılığıyla- kaynak sunucuya yaptığı bir GET isteği sonucu dönen cevabı, düzenlemesi için ICAP sunucusuna gönderir. Bu örnekte kaynak sunucu adresi "www.origin-server.com", ICAP sunucusunun adresi ise "icap.example.org" dir.

```
RESPMOD icap://icap.example.org/satisf ICAP/1.0

Host: icap.example.org

Encapsulated: req-hdr=0, res-hdr=137, res-body=296
```

```
GET /origin-resource HTTP/1.1

Host: www.origin-server.com

Accept: text/html, text/plain, image/gif

Accept-Encoding: gzip, compress



HTTP/1.1 200 OK

Date: Mon, 10 Jan 2000 09:52:22 GMT

Server: Apache/1.3.6 (Unix)

ETag: "63840-1ab7-378d415b"

Content-Type: text/html

Content-Length: 51

33

This is data that was returned by an origin server.

0
```

Şekil 5.7. ICAP Respmod İstek Mesajı

```
ICAP/1.0 200 OK

Date: Mon, 10 Jan 2000  09:55:21 GMT

Server: ICAP-Server-Software/1.0

Connection: close

ISTag: "W3E4R7U9-L2E4-2"

Encapsulated: res-hdr=0, res-body=222
```

```
HTTP/1.1 200 OK

Date: Mon, 10 Jan 2000  09:55:21 GMT

Via: 1.0 icap.example.org (ICAP Example RespMod Service 1.1)

Server: Apache/1.3.6 (Unix)

ETag: "63840-1ab7-378d415b"

Content-Type: text/html

Content-Length: 92


5c

This is data that was returned by an origin server, but with

value added by an ICAP server.

0
```

Şekil 5.8. ICAP Respmod Cevap Mesajı

5.4.5. ICAP Protokolünün VSE Amaçlı Kullanımı

Önerilen yöntem, ICAP protokolünün REQMOD fonksiyonu kullanılarak istemci tarafından gönderilen isteğin kaynak sunucuya ulaşmadan vekil sunucu aracılığıyla ICAP sunucusuna yönlendirilip – incelenmesidir. Daha sonra ICAP sunucusu ICAP isteği içerisinden HTTP mesajını ayıklamaktadır. HTTP isteğini de başlık ve gövde kısımlarına ayırmaktadır. Daha sonra bu kısımlar üzerinde hassas veri tanımlama yöntemlerini uygulamaktadır. Eğer uygulanan yöntemler sonucu risk faktörü bulunmuşsa istek yerine bir HTTP cevabı (403 Forbidden) gönderilerek erişimin kısıtlandığı vekil sunucu aracılığıyla istemciye bildirilmektedir. Eğer risk faktörü

bulunamamışsa mesaj geldiği şekliyle gönderilir; böylece ICAP sunucusunun etkisi olmaksızın gönderilen istek kaynak sunucuya ulaştırılmıştır.

1. Vekil sunucu istemciden gelen isteği ICAP sunucusuna göndermektedir,
2. ICAP sunucusu alınan ICAP isteğinden HTTP mesajını ayrıştırmaktadır. Daha sonra bu mesajı başlık ve gövde olmak üzere iki kısıma ayırmaktadır,
3. ICAP sunucusu başlık ve gövde kısımları üzerinde ayrı ayrı hassas veri tanımlama yöntemlerini uygulamaktadır,
4. Eğer risk faktörü bulunursa, gelen istek yerine "403 Forbidden" (yasak) hata kodunu içeren bir HTTP cevabı döndürmektedir,
5. Eğer risk faktörü bulunamazsa gelen mesaj geldiği şekliyle geri döndürülmekte ve akış sürmektedir.

Yapılan örnek program (M-ICAP) yazılımı yasaklamanın yanında kayıt altına alma seçeneği de sunmaktadır. Bu seçenekler birlikte kullanılabileceği gibi ayrı ayrı da kullanılabilmektedir. Yukarıda anlatılan sistemin temel işleyişi olup, M-ICAP programında kullanıcı tarafından özelleştirilebilir birçok seçenek bulunmaktadır.

5.5. HTTPS

HTTPS (*Hypertext Transfer Protocol Secure – Güvenli Hipermetin Aktarım İletişim Kuralı*) hipermetin aktarım iletişim kuralının (HTTP) güvenli ağ protokolü ile birleştirilmiş olanıdır. Klasik HTTP protokolüne SSL/TLS protokolünün eklenmesi ile elde edilir [15]. Kriptolanmış bir iletişim sağlamaktadır. Bu yüzden internet üzerinden ödeme sistemlerinde sıklıkla kullanılmaktadır. HTTP 80. portu kullanırken, HTTPS için varsayılan port 443 tür.

SSL, 1994 yılında Netscape tarafından Web üzerindeki işlemleri güvenli hale getirmek için geliştirilmiştir. Daha sonra, Internet Engineering Task Force (IETF) tarafından SSL 3.0 protokolünü baz alan standart bir protokol geliştirme çalışmalarına başlandı ve TLS ortaya çıkmıştır.

HTTP OSI Modelinin en üst katmanında (uygulama seviyesinde) çalışan bir protokol iken güvenlik protokolleri olan SSL ve TLS bir alt katmanda (sunum katmanı) çalışmaktadır.

5.5.1. HTTPS Trafiğinin Denetlenmesi

Bir vekil sunucu üzerinden HTTPS bağlantıları kurulurken HTTP protokolünün CONNECT metodundan yararlanılır. Bu metot, vekil sunucunun kaynak sunucu ve istemci arasında bir tünel açarak her ikisiyle birden bağlantı sağlayıp, birinden gelen veriyi içeriğini incelemeden diğerine aktarmasıyla gerçekleştirilir. Burada vekil sunucu man-in-the-middle (ortadaki adam) konumundadır.

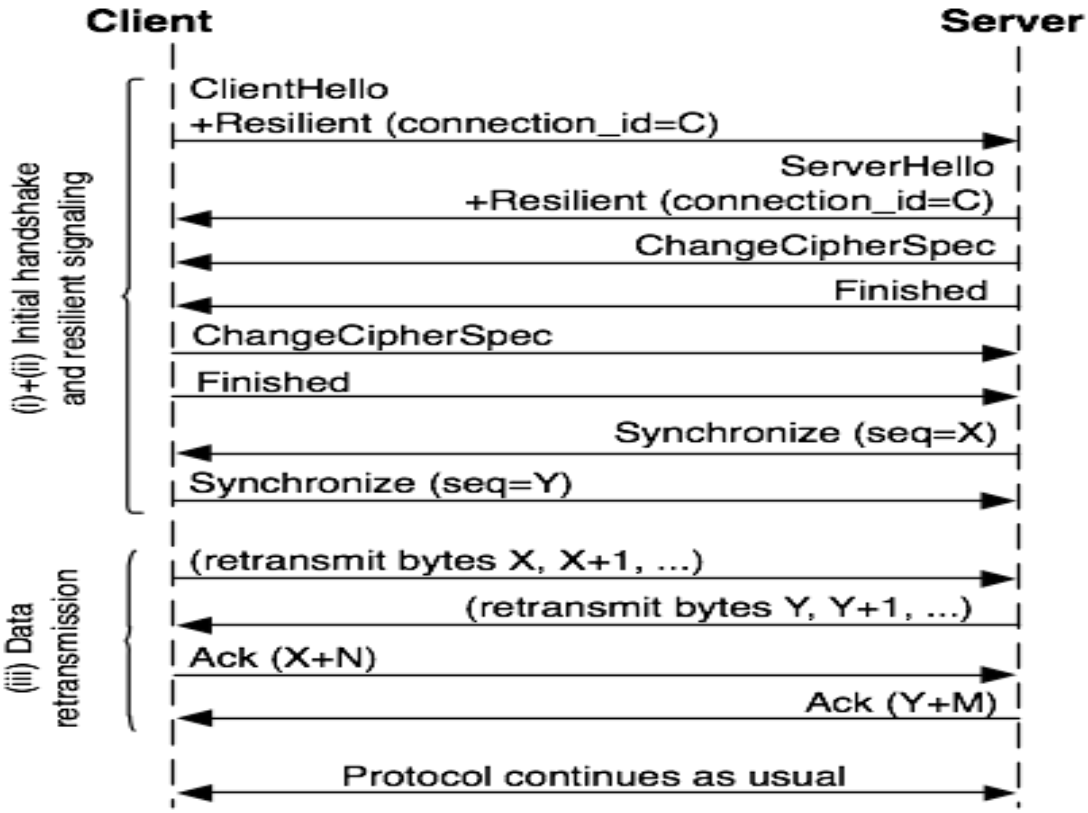
Tezin mimarisinde bulunan ağ tabanlı sunucu içerisinde açık kaynak kodlu Squid ön bellekli vekil sunucu yazılımı kullanılmıştır.

Squid programı sslBump adlı özelliği sayesinde istemcinin gönderdiği veriyi inceleyip, kullanılan şifrelemeyi çözebilmektedir. Yani en başta istemciye kendi sertifikasını gönderip istemciden gelen verinin şifresini gizli anahtarı ile çözmektedir. Daha sonra bunu kaynak sunucunun istemciye gönderilmesi için yolladığı fakat vekil sunucu tarafından saklanan açık anahtar ile şifreler ve kaynak sunucuya iletilmektedir. Benzer şekilde kaynak sunucudan gelen cevabı açık anahtarla açar daha sonra kendi sertifikasının gizli anahtarıyla şifreleyip istemciye ulaştırılmaktadır. Etik olmayan bu davranış, ICAP aracılığıyla HTTPS trafiğinin yakalanması ve denetlenmesi için tek yöntemdir.

Burada kullanılan “sahte sertifika” nın sebebi, TLS de kullanılan şifreleme algoritmalarının asimetrik olması ve vekil sunucunun gizli anahtara sahip olmamasıdır. Bu yüzden OpenSSL programı kullanılarak bir sahte SSL sertifikası üretilmiştir. Sertifikaya sahte denmesinin sebebi herhangi bir Certification Authority (sertifika sağlayıcı) tarafından imzalanmamış olmasıdır. Bu nedenle popüler web tarayıcıları, istemciyi olası bir man-in-the-middle saldırısı hakkında uyaran mesajlar göstermektedir. OpenSSL yardımıyla üretilen .pem dosyası bu tarayıcılarda güvenilir olarak gösterilip, bu mesajlar mümkün olduğunca engellenmeye çalışılmıştır.

5.6. TLS

TLS (*Transport Layer Security – Taşıma Katmanı Güvenliği*), günümüzde HTTPS kullanan web sitelerinin büyük bölümü tarafından şifreleme protokolü olarak kullanılmaktadır. Bir TLS bağlantısının genel işleyişi Şekil 5.9. 'de görüldüğü gibidir.



Şekil 5.9. Transport Layer Security İletişimi

Basitçe TLS El Sıkışması:

1. İstemci bir oturum isteği gönderir. Örneğin https://example.org/index.html adresine girmek istemektedir,
2. Sunucu açık anahtarını (public key) içeren sertifikasını istemciye göndermektedir,
3. İstemci, gelen sertifikayı bilinen sertifika sağlayıcıları ile doğrulamaktadır, (Eğer doğrulama başarısız olursa bir çok web tarayıcı seçimi kullanıcıya bırakacaktır.)

4. İstemci rasgele bir simetrik anahtar üretmekte ve bu anahtarı sunucunun açık anahtarını kullanarak şifrelemektedir,

5. Artık istemci ve sunucu üretilen simetrik anahtarı bilmektedir. Oturum boyunca, birbirleriyle iletişim kurarken veriyi bu anahtar ile şifrelemektedir.

Tezimde kullanılan protokollerin ve yöntemlerin tanımlanmasının ardından ileriki bölümlerde bu tezde geliştirilen VSE sistem mimarisinin bileşenleri anlatılacaktır.

6. İSTEMCİ TABANLI VSE MİMARİSİ TASARIMI

Tezin bu bölümü VSE sisteminde geliştirilen istemci tabanlı mimari yapıyı anlatmaktadır. Veri sızıntısının başlayacağı yerlerden biri kullanıcıların bilgisayarlarıdır. Ağ bağlantısı olmadan istemci üzerinden veri sızıntısının engellemesi bu bölümün konusudur. İstemci olarak Windows tabanlı işletim sistemi kullanılmıştır. İstemci tabanlı çözümü de içine alan VSE organizasyon şeması Şekil 6.1.'de gösterilmiştir. VSE yönetim sunucusu istemcilere kuralları dağıtır. VSE yönetim sunucusunun dağıttığı kuralları dizüstü ve masaüstü bilgisayarlar almaktadır. İstemcilerin her birinde bir VSE ajan yüklüdür. Hassas verilerin erişimini kısıtlamak için VSE ajan kullanılmaktadır. Bir VSE ajanın var olma zorunluluğu istemci tabanlı güvenlik sistemlerine dayanmaktadır. İstemci gerekli şekilde izlenmeli ve kontrol edilmelidir. VSE ajan var olmadığında, veri sızıntısı oluşacak ya da söz konusu olacaktır. Genelde veri sızıntısını yapan kullanıcı çeşitli işletim sistemi haklarına sahip olacaktır.



Şekil 6.1. VSE Organizasyon Şeması

VSE ajan veri sızıntısı engelleme çözümü olarak temel istemci tabanlı veri bileşenidir. Kullanıcı işlemlerini sağlayabilmek için kurallara ve denetim elemanlarına uymak zorundadır. Windows altında VSE ajan parçası bir çekirdek (kernel) sürücüsü

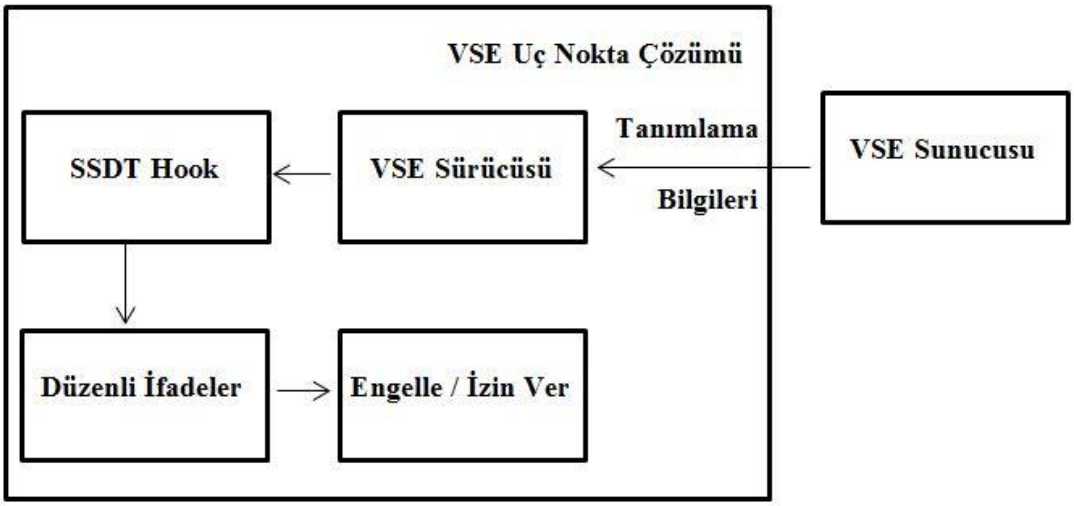
olarak uygulanmakta ve işletim sisteminin izlenmesi mümkün olmaktadır. Windows altında VSE ajanı rootkit kullanımına benzemektedir. Bu yüzden bazı üreticiler kendi güvenlik çözümlerinin içerisine Rootkit ajanlarını gizlemektedir. Ayrıca yönetici hakları ve standart araçları ile VSE ajanların genellikle tespiti mümkün değildir. Bunun mümkün olması için bir anti – rootkit aracı kullanmak ve Windows sistem yönetici haklarına sahip olması gerekmektedir.

Bu tezde geliştirilen VSE mimarisi uygulamasının istemci tarafındaki görevleri aşağıdaki gibidir:

- Yönetim sunucusundan kuralları almak,
- Kuralların uygulanmasını sağlamak,
- Verileri sınıflandırmak,
- İzleme / Loglama.

VSE ajan verilerinizi korumak için önce doğru tanımlamaları almak zorundadır. Hangi eylemlere izin verildiğini, hangi eylemlere izin verilmediğini bilmelidir. VSE bu kuralları yönetim sunucusundan almaktadır. Veri aktarımı bütünlüğü burada sağlanmalıdır. Aksi takdirde doğru kararlar alınamayacak ve doğru sonuçlar verilemeyecektir. Kuralların işletimi VSE yönetim sunucusundan istemci tarafına doğru yapılmaktadır. Kurallar her zaman gizli olmasa da (onlar şirket politikası olarak yayınlanmaktadır), temel bir koruma sağlamaktadır. Tüm içeriğin genel olarak bilinmesi gerekmektedir.

Geliştirilen mimaride VSE ajan kuralları toplar ve yetkisiz değişikliklere karşı korumaktadır. Eğer bilgilendirme dosyalarının alınmasında ya da sunucu ile veri aktarımının kopması halinde tanımlama dosyalarının istemcide saklanması zorunlu olmaktadır. Yerel depolama bu yüzden gereklidir. VSE ajanı ağ bağlantısı olmadan da çalışabilir durumda olmalıdır. Geliştirilen VSE istemci mimarisinin yapısı aşağıdaki şekilde gösterilmektedir.



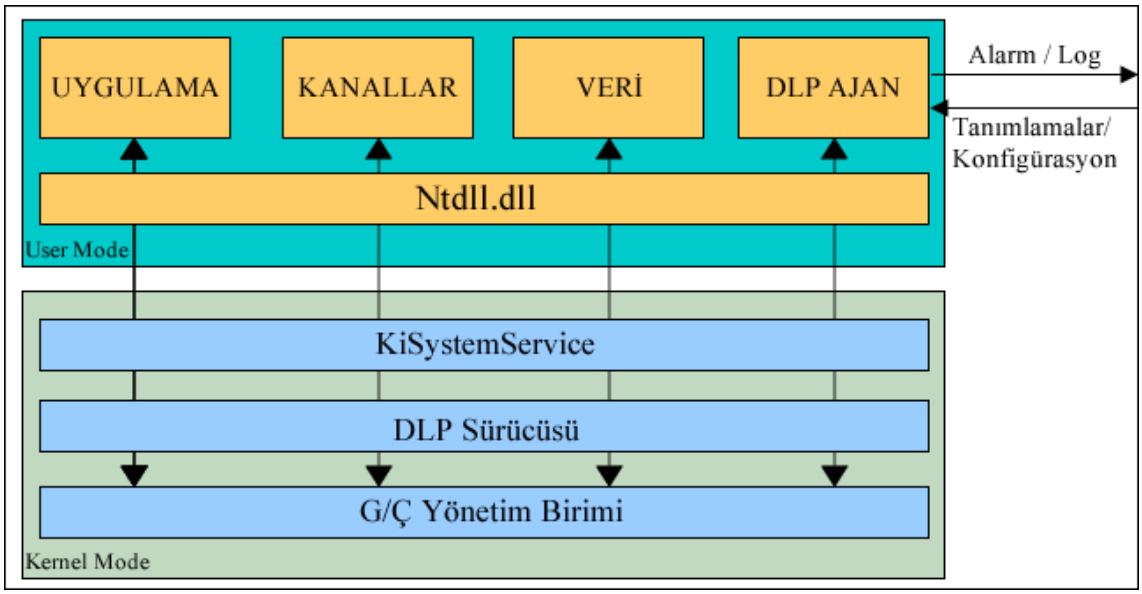
Şekil 6.2. VSE İstemci Mimari Yapısı

Şirket ağına kalıcı bir bağlantı yapmadan önce, tanımlama dizüstü bilgisayarlar içinde önemlidir. Sabit disk üzerinde yerel olarak depolanan verilerin korunması için kurallar oluşturulmalıdır. Çünkü risk, saldırganın kuralları değiştirmeye çalışarak VSE sistemini atlamasıdır. Sunucu tabanlı kurallar yönetimi VSE ajan kullanılarak kullanıcılara dağıtılır. Bir kural ihlali olması durumunda VSE ajan etkisini göstermektedir. Ayrıca kullanıcıya bir iletişim kutusu gösterilir, onaylanabilir ve kural ihlali olarak kabul edilmektedir.

Buna paralel olarak VSE ajan ihlalleri günlük (log) dosyası olarak tutulur ve mesajları VSE yönetim sunucusuna gönderilir. Bu nedenle olaylarda merkezi bir ilişki mevcuttur. VSE yönetim sunucusuna erişilemiyorsa, VSE ajan tarafından tanımlamalar önbelleğe alınmakta, sunucuya erişim sağlandığında veri gönderilmektedir. Verilerin korunması için kuralların uygulanmasının yanı sıra VSE ajan ile dosyaların sınıflandırılması gereklidir. Yönetim sunusunda yapıldığı gibi dokümanların parmak izi ve istatistiksel içerik analizi uç nokta yönetimine de uygulanabilir. İstemci tarafına istatistiksel tanımlama dosyalarının yönetim sunusundan gönderilmesi ve kuralların uygulanması gerekmektedir. Ağ iletişiminin olmadığı bir istemci düşünülürse, değişen istatistiksel içerik analiz bilgileri yanlış sınıflandırmaya neden olabileceği için bu sistem, istemci tarafına dahil edilmemiştir.

Dosyalar VSE ajan tarafından korunmaya alınmaktadır. Eğer dosya ajan tarafından değiştirilseydi bütünlük anlamı ile kayıp yaşanmasına neden olabilirdi.

VSE Ajanı çekirdek (kernel) modül olarak uygulanabilir. Çünkü kopyalama ve taşıma gibi işlemleri kapsayarak kullanıcının veri kontrolünü sağlamayabilmelidir. Şekil 6.3. geliştirdiğimiz mimarideki VSE sürücüsünün istemci tabanlı kernel yapısını göstermektedir. Sistem iki bölümden oluşmaktadır, birincisi VSE kernel mode sürücüsü ikincisi VSE user mode programıdır. VSE sürücü çekirdek (kernel) modunda çalışmakta ve ilgili tüm işlemleri denetleyebilmektedir.



Şekil 6.3. VSE Sürücüsü Kernel Şematiği

Bu işlemler dosya sistemi istekleri, ağ iletişimi ya da diğer girdi / çıktı işlemleridir. VSE sürücüsü ile iletişim kurularak öngörülen kurallara uyumu VSE ajan izlemektedir. VSE ajan aktif değilken, kullanıcı VSE sistemi tarafından yerleştirilen tüm kısıtlamaları atlayabilmektedir. Böylece bir dosya sistemi izinleri ve güvenlik duvarları gibi sadece koruyucu önlemler burada kalmaktadır. Yönetici haklarına sahip bir saldırgan VSE ajanı devre dışı bırakabilir. Ama veri sızdırmak için VSE ajan devre dışı bırakıldığında işletim sistemi koruma olarak mavi ekrana düşürülmektedir. Bu şekilde VSE ajanın devre dışı bırakılması engellenmektedir.

6.1. Çekirdek (Kernel) Sürücüsü Uygulaması

Bu tezde geliştirilen VSE ajan iki bölümden oluşmaktadır. Bir parçası kullanıcı parçası (user land), diğeri ise çekirdektir. Çekirdek parçası işletim sisteminin ilgili tüm birimlerine erişimi sağlamaktadır. Çekirdek sürücü (kernel driver) uygulaması VSE sistemi için çekirdek rootkite (kernel rootkit) benzemektedir. Yüklenebilir çekirdek modülleri çalışma zamanında işletim sistemine entegre olmaktadır. Bir kez model veya sürücü yüklenmekte, VSE ajan çekirdek veri yapıları için sistem çağrı tablosu (system-call-table) veya sürücü modları listesi kullanılmaktadır.

Aşağıdaki rootkit teknolojileri VSE sistemleri tarafından kullanılabilir:

- Sistem Çağrı Tablosu Manipülasyonu (Windows altında SSDT Hooking),
- Inline Hooking (Programlama ile doğrudan bellek erişimi)
- İşlem listesine (processlist) manipülasyon (User Mode üzerinde çalışan program için kabuk örme işlemi)
- Interrupt Hooking (Interrupt Descriptor Table üzerinde yapılan kesme işlemleri)

Yukarıdaki dört olasılıktan tezde kullanılan SSDT hooking yöntemidir.

6.2. Sistem Çağrı Tablosu Manipülasyonu (SSDT Hooking)

System Service Descriptor Table, işletim sisteminin açılışı sırasında hazırlanan ve daha sonra tüm sürücüler ve diğer çekirdek bileşenleri tarafından kullanılan kritik bir sistem tablosudur. Temel olarak bakıldığında aslında basit bir listedir, hangi sistem işlevi hangi adreste sunuluyor anlamında bir bakım listesi olarak kullanılabilmektedir. Sürücümüz bu tabloyu düzenleyerek sisteme ait bir takım işlevlerin öncelikle kendi kod lokasyonlarına uğramasını ve daha sonra sistemin yapacağı işleme devam etmesini sağlamaktadır.

Bu tekniği kullanmaktaki amacımız hassas veri içeren ilgili dokümanı ilgili işlevden geçmekte olan isteğin kullanıcı tarafından istenmiş bir işlem olup olmadığını

anlamak ve eğer hassas içerik içeriyorsa kullanıcı isteğini gerçekleşmeden durdurmaktır. Günümüz işletim sistemlerinde bu işin yürütülebilmesi için iki yol vardır. Kullanıcı modu (user mode) ve çekirdek modu (kernel mode). Intel mimarisi işletim sistemi yazarları için dört farklı yetki modu sunmaktadır (ring 0-3). Bu yetki modlarının sadece iki tanesi işletim sistemi tarafından kullanılır (ring 0 ve ring 3). Ring 0 kernel mode, ring 3 user mode olarak bilinmektedir. User mode da çalışan kod bazı korumalı işlemci komutlarını çalıştıramaz, korumalı bellek alanlarını okuyup yazamaz ve donanıma doğrudan erişemez. Kernel mod'da çalışan programlar ise hiçbir güvenlik engeline takılmadan tam yetki ile çalışmaktadır.

Apilerin WinNT32 üzerindeki yolculuğu C dili ile yazılmış programda, fprintf gibi bir fonksiyon çağırıldığında, öncelikle C runtime library içindeki fprintf fonksiyonu çalışmaktadır (msvcrt.dll). Bu fonksiyon gerekli parametrik kontrol ve ayarlamalardan sonra WriteFile apisini çağırmaktadır (kernel32.dll). WriteFile apisi gerekli parametrik ayarlamaların ardından NtWriteFile fonksiyonunu çağırmaktadır (Ntdll.dll).

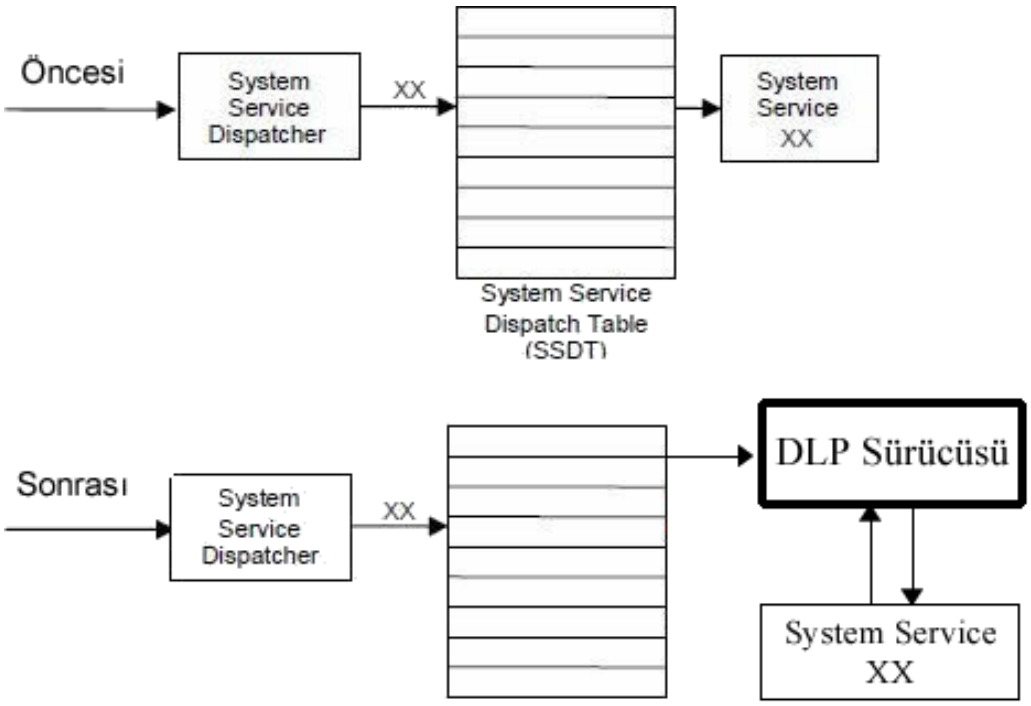
Bu aşamada Ntdll.dll içerisindeki bir stub, gerekli ayarlamaları yapmakta, ilgili sistem çağrısının indeks numarasını eax register değerine atamakta, ardından sysenter – syscall işlemci komutunu çalıştırmaktadır. Bu komutun görevi işlemciyi kontrollü bir şekilde user mod'dan kernel mod'a geçirmektir. User mode programlarının donanıma doğrudan erişemediğini belirtilmiştir, buna rağmen kullanıcıların doğrudan kullandığı programların neredeyse tamamı user mode programlarıdır. Bu durumda kullanıcının bir diske ya da bir aygıta veri yazmasını sağlayan sysenter komutudur. Kullanıcı bir aygıta yazma işlemi uygulamak istediğinde ntdll.dll son noktayı oluşturmaktadır. Bu noktada işlemciyi kernel mod'a geçirip gerekli sistem çağrısı tamamlandığında tekrar sysexit ile user mod'a geri dönmek işletim sisteminin görevindedir. Kullanıcı talep ettiği sistem çağrısı tamamlandıktan sonra ntdll.dll içerisine geri dönmektedir.

Eax register kısmında değeri işletim sistemi index gibi kullanır ve SSDT tablosundaki adres karşılığını bulmaktadır. Bu adresteki fonksiyon çağırılmak istenen sistem fonksiyonudur.

SSDT tablosunun manipülasyonu nasıl bir sonuç doğuracaktır? SSDT tablosunu değiştirmenin tek yolu kernel mode üzerinde çalışan bir program yazmaktır. SSDT

tablosundaki herhangi bir girdiyi değiştirir, onun yerine kendi fonksiyonumuzun adresi yazılırsa, tüm user mode programlar, yazdığımız programı çalıştırmış olacaktır. Bu programın parametreleri incelenerek, fonksiyonun çağırılması kontrol edebilmektedir. Eğer SSDT tablosundaki eski fonksiyon kendi içimizde çağırılırsa, kullanıcı hiçbir şeyin farkında olmayacaktır. Eğer izin verilmezse, basitçe fonksiyonu sonlandırır ve user mod'un program isteği çağrıyı yapamamış olacaktır.

VSE istemci tabanlı mimari çözümünü sağladığımız SSDT tarafında NtWriteFile sistem çağrısı hook edilmektedir. Bu çağrı, pipe, disk dosyası vb. aygıtlara veri yazmayı sağlayan fonksiyondur. Fonksiyonun bir parametresi, yazılmak istenen bellek (buffer), bir başka parametresi de bu belleğin boyutudur. Bu bellek kontrol edilerek, yazma işlemi onaylanabilir veya reddedilebilir. VSE istemci tabanlı çözümü ile bellek, user mode programına aktarılmış ve user mode üzerindeki belleğe hassas veri arama yöntemleri uygulanmıştır. Bu teknikler uygulanarak karar işlemi gerçekleştikten sonra user mode programı kernel mode sürücüsü ile tekrar iletişime geçerek karar bildirilir ve bu karar mevcut yazma işleminin sonucunu belirlemektedir.



Şekil 6.4. SSDT Hook Tablosu

Tezde geliştirilen mimaride SSDT hook uygulama adımları aşağıdaki gibidir:

- Yazma korumasını pasifleştirmek,
- Gerekli yapıların yerlerini belirlemek,
- Tabloyu yamalamak,
- Yazma korumasını tekrar aktifleştirmek.

Geliştirilen VSE mimarisinde yönetim sunucusundan istemci tarafına yüklenen tanımlama dosyası ile istemci tarafında çalışan VSE ajan tanımlamaları almış bulunmaktadır. Alınan tanımlanamalar ile yerel disklerde ve dışarıdan takılan medyalarda aktarım esnasında ve dosya yaratım esnasında hassas veri aranmaktadır. Arama işlemlerinde tanımlama dosyasında bulunan kelimelere ve düzenli ifadelere göre sonuçlar belirlenmektedir. Düzenli ifadeler, veriler içinde istenilene göre belirleyebileceğimiz kompleks sözdizimine sahip ifadeleri bulmak için kullanılan bir tekniktir. Bu teknik bir string içinde başka bir string bulmamıza ve istersek değiştirmemize yaramaktadır. Sözdizimi oluşturmak için kullanabileceğimiz birkaç yöntem bulunmaktadır:

- Karakter uyuşması,
- Tekrarlanma uyuşması,
- Pozisyon uyuşması.

Şartlarımıza uyan verilerin kopyalanmasına ve başka ortamlara taşınmasına izin verilmemektedir.

6.3. VSE İstemci Mimarisinin Özellikleri

VSE Ajan aşağıdaki özellikleri taşımaktadır.

	Uç Nokta Yönetimi
Filtreleme Türleri	Mime (dosya) türleri Regular expressions Kredi kart numarası Sosyal güvenlik numarası Uluslar arası banka hesap numarası T.C kimlik numarası
Filtreleme	Uç nokta verileri Taşınabilir ortam güvenliği CD / DVD hassas veri içerik engelleme
Dosya Türleri	MS Office 2003/2007/2010 OpenOffice PDF XML / HTML RTF / Text

Tablo 6.1. VSE istemci mimari özellikleri

7. VSE SİSTEM MİMARİSİ İÇİN METİN SINIFLANDIRMA

Tezin bu bölümü VSE sistem mimarisinde geliştirilen ağ tabanlı mimaride kullanılacak metin sınıflandırma yöntemlerinin karşılaştırmalı deneysel sonuçlarını içermektedir. Metin sınıflandırma, yazılı belgelerin içeriklerine bağlı olarak belirli sınıflara atanması işlemine verilen isimdir (Jackson, 2002). Metin sınıflandırmadaki amaç bir dokümanın özelliklerine bakılarak önceden belirlenmiş belli sayıdaki kategorilerden hangisine dâhil olacağını belirlemektir. Makine öğrenmesi ile bilgisayarda işlenebilir hale getirilen verilerden örnek bir dizi eğitilerek sınıflandırıcı oluşturulmaktadır. Makine öğrenmesi kişisel yorumlardan kaynaklanan sınıflandırma hatalarını ortadan kaldırabilmektedir. Makine öğrenmesi yaklaşımı ile yapılan otomatik sınıflandırmada iş yükü azaltılmıştır. Ayrıca elle indeksleme yolu ile uzun zaman alan çalışmalarda makine öğrenmesi yaklaşımı zaman tasarrufu sağlamaktadır. Sınıflandırma hatasının azaltılması, zaman tasarrufu sağlaması, iş yükünü azaltması ve metinlerin sınıflandırılmasında iyi performans sergilemesi ile makine öğrenmesi yaklaşımı bilgi teknolojisinde gittikçe önem kazanmaktadır. VSE metin sınıflandırma konusu, kurumsal ağ üzerinde paylaşılan dokümanların sınıflandırılmasını sağlayarak, kurum için önem arz edenleri otomatik engellenmesidir. Bu nedenle hangi sınıflandırma algoritmalarının veri seti üzerinde performanslı çalıştığı geliştirdiğimiz VSE mimarimizde metin sınıflandırmanın amacıdır. Metinlerin veya dokümanların sınıflandırılmasında, verilerin bilgisayar ortamında incelenmesini sağlayan özellik çıkarımı teknikleri ve ağırlıklandırma yöntemleri ile sınıflandırma algoritmalarının incelenmesi ve performans değerlerinin karşılaştırılması VSE mimarimizin önemli konusudur. VSE mimarimiz için metin sınıflandırmada, özellik çıkarımı tekniklerinden words ve n-gram teknikleri incelenmektedir. Ağırlıklandırma yöntemlerinden frekansa göre ağırlıklandırma, terim frekansı x ters doküman frekansı, kelime kümeleme (k-means) yöntemlerine yer verilmektedir. Makine öğrenmesi yöntemlerinden Naïve Bayes, K-En Yakın Komşuluk Algoritması (IBK), Karar Ağaçları (J48) ve Destek Vektör Makinesi (SVM) sınıflandırma algoritmaları incelenmektedir. İki sınıflı veri seti üzerinden toplam 16 adet özellik vektörü oluşturulmaktadır.

7.1. Deneysel Çevre

Verilerin kurumların izin verdikleri şartlar altında kullanılması ve kurumsal veri düzenine verilecek zararların en aza indirgenmesi gerekmektedir. Doküman sınıflandırmadaki amacımız kurum için hassas ve gizli olarak etiketlenmiş ve önceden belirlenmiş kategorilerden hangisine dâhil olacağını belirlemektir. Tezde geliştirilen VSE mimari metin sınıflandırması kategorilere göre metin akışına izin verilip verilmeyeceğine karar vermektedir. Kullanılan dokümanların boyutları 1 kb ile 36 kb arasında değişen 800 adet İngilizce doküman kullanılmıştır. Hassas veri içeren dokümanlar Amerika Birleşik Devletlerine ait savaş konulu secret, confidential ve sensitive olarak etiketlenmiş 400 adet dokümandır [18]. Diğer normal veri içeren dokümanlarda 400 adet ekonomi, spor, sosyal ve diğer bilgilerden oluşan makalelerden alınmıştır. Bu dokümanların yüzde onu ile sınıflandırma başlamıştır. İki adet sınıf içerisine 40 adet hassas veri içeren (class1), 40 adet normal içerikli (class2) doküman konulmuş ve sayı her seferinde 5 arttırılmıştır. Her bir veri seti için iki (2-gram ve words) özellik vektörü kullanılarak sınıflandırma işlemi gerçekleştirilmiştir. Özellik vektörleri hazırlanırken 2-gram ve Words için max. frequency değeri 10 ve 50 olarak değiştirilmiş ve k değeri içinse denemeler sonucunda sabit 50 değeri alınması kararlaştırılmıştır.

7.2. Çalışmanın Amacı ve İzlenen Metodoloji

Tezde geliştirilen VSE mimarisi, doküman sınıflandırılmasında kullanılan özellik çıkarımı teknikleri ve ağırlıklandırma yöntemleri ile sınıflandırma algoritmalarının performans değerlerinin karşılaştırılmasını içermektedir. Gövdeleme yöntemi kullanılarak elde edilen 2-gram ve words, kümeleme yöntemlerinden K-Means algoritması gibi dil bilgisel ve istatistiksel özellikler kullanılarak 16 farklı özellik vektörü çıkarılmış ve bu özellik vektörleri ile sınıflandırma işlemi gerçekleştirilmiştir. Makine öğrenmesi yöntemlerinden olan Naïve Bayes, destek vektör makinesi, k en yakın komşuluk algoritması ve karar ağaçları kullanılarak dokümanın hassas veri içerip içermediği belirlenmiştir. Geliştirilen mimaride, tüm analizlerde 10-kat çapraz geçerlilik veri eğitim seçeneği kullanılmıştır. K katlı çapraz doğrulama, yaklaşık olarak eşit boyutta k alt parçaya bölünen bir indükleyici doğruluğunu tahmin etmeye çalışan bir metottur. İndükleyici k defa eğitilir ve test edilir. Doğruluk tahmini k katlar için

ortalama doğruluktur (Kohavi, 1998). Çapraz doğrulama kullanılarak veri miktarında sınırlama yapılabilmektedir (Witten, 2005).

Kullanılan sınıflandırıcıların performanslarını belirlemek için f-measure değeri kullanılmıştır. F-Measure değeri precision ve recall değerlerinin harmonik ortalaması olarak ifade edilmektedir. Pozitif ve negatif olarak sınıflandırılan örnekler için precision, pozitif olarak tahmin edilen örnekler içerisinden doğru olarak sınıflandırılan pozitiflerin oranını vermektedir. Recall ise gerçek sınıfı pozitif olan örnekler içerisinden doğru olarak sınıflandırılan pozitif örneklerin oranını vermektedir (Kocamaz, 2011).

$$\Pr ecision = \frac{TP}{TP + FP} \tag{7.1}$$

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \tag{7.2}$$

$$F - Measure = \frac{2 \times \Pr ecision \times \mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \tag{7.3}$$

F-measure değeri özellikle eğitim verilerinin hazırlanış sürecinde sınıflandırıcının performansını arttırmak için önemli bir ölçüt olmaktadır (Kocamaz, 2011). Bu açıdan Kılıçaslan'ın sınıflandırıcıların performans analizinde F-measure ve veri büyüklüğü arasındaki ilişkiyi ortaya koyduğu çalışma önemlidir (Kılıçaslan, 2009). VSE mimarimizde belirtilen kriterler kullanılarak çizilen öğrenme eğrisiyle happy graphy (Russell, 2002) elde edilmeye çalışılmıştır.

Ayrıca Amasyalı ve arkadaşları tarafından metinlerin özellik çıkarımını çeşitli yöntemlerle yapan, ağırlıklandırma metotları yardımıyla metinleri sayısallaştıran, metinleri WEKA programının giriş dosyası olan ARFF formatına çeviren "text2arfff" adlı yazılım geliştirilen sistemle bütünleştirilmiştir (Amasyalı, 2010).

7.3. Deneysel Sonuçlar

Algoritmalar arasındaki farklılıklar aşağıdaki nedenlere bağlanabilir;

- Veri önişleme aşamasında yapılan nitelik seçimi, verilerin tamamlanması gibi analiz sonuçlarını etkileyici işlemlerde model çıkarımını etkileyici işlemler yapılması,
- Farklı önişlemlerle oluşturulan verilerin analiz sonuçlarının farklı olması.

Sınıflandırma performansını etkileyen faktörler aşağıdaki gibi olabilir;

- Algoritmalardaki farklılıklar,
- Veri setine özgü özellikler,
- Yöntem ve problemler arasındaki uyumsuzluklar.

Veri setine özgü özellikler aşağıdaki gibi olabilir;

- Sınıf belirsizliği,
- Örnek azlığı.

Sınıf belirsizliği, sınıflandırma problemi içinde verilen özelliklerle herhangi bir sınıflandırma algoritması kullanılarak ayrım yapılamadığı durumları belirtmektedir. Sınıflandırma zorluğunu arttıran bir diğer etken de verinin seyrek olmasıdır. Sınıflandırıcının genelleme mekanizmasını sınırlayacak kadar örnek ile örneklenmemiş durumların sınıflandırması büyük oranda rastgele olacaktır.

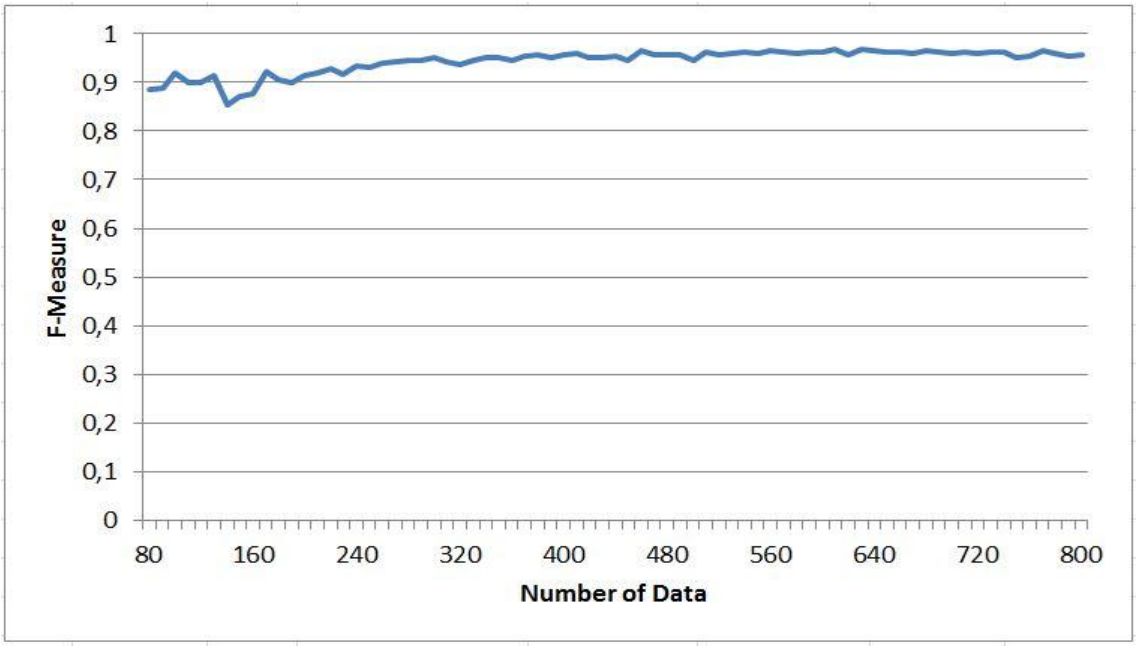
Naïve Bayes, normal dağılım varsayımına (Manning, 2008) sahip doğrusal bir sınıflandırıcı olup, çekirdek yoğunluk kestiricisi kullanıldığında doğrusal olmayan bir sınıflandırıcıya dönüşebilmektedir (John, 1995). Ağırlıklandırma yapılırken text2arrf yazılımı kullanılarak arff dosyaları Tablo 7.1'deki seçimlere göre yapılmıştır.

Tablo 7.1. Yapılan deneylerin parametre seçimleri

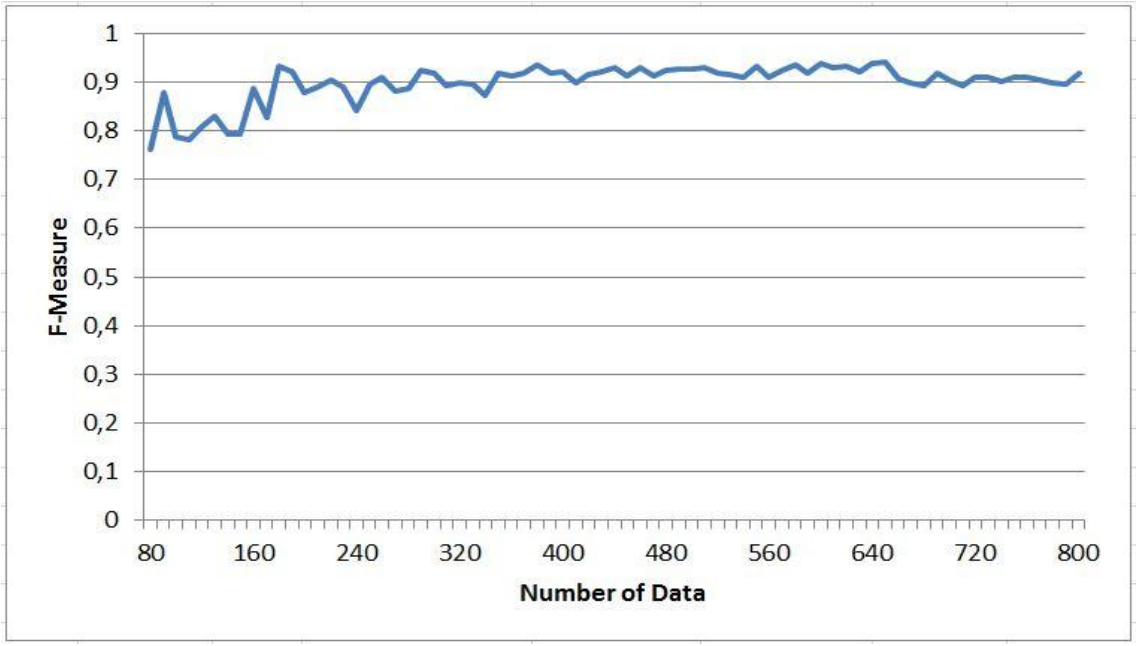
Deneyler	**Yöntem**	**Tf / Tfxidf**	**Frequency**	**k**
1. Deney	2-Gram	Tfxidf	10	50
2. Deney	2-Gram	Tfxidf	50	50
3. Deney	Words	Tf	10	50
4. Deney	Words	Tf	50	50

Yukarıda yazılan seçimler dört adet makine öğrenmesi yönteminde uygulanmıştır. Dokümanlar ise iki sınıf halinde ayrılmış ve kırkar adet doküman içererek bu sayı her klasörde beşer artarak uygulanmıştır. Örnek olarak kırk doküman ikinci denemede kırk beş, üçüncü denemede elli diye artarak son denemede ayrı ayrı iki sınıf içerisinde dört yüzer doküman sayısına ulaşılmış, toplamda sekiz yüz dokümana ulaşılmıştır. K en yakın komşu algoritmalarındaki k değerleri bir ile otuz arasında değiştirilerek hesaplanmıştır. Destek Vektör Makinelerinde yapılan bir değişiklik ise SVM type olarak nu-SVC classification kullanılmıştır.

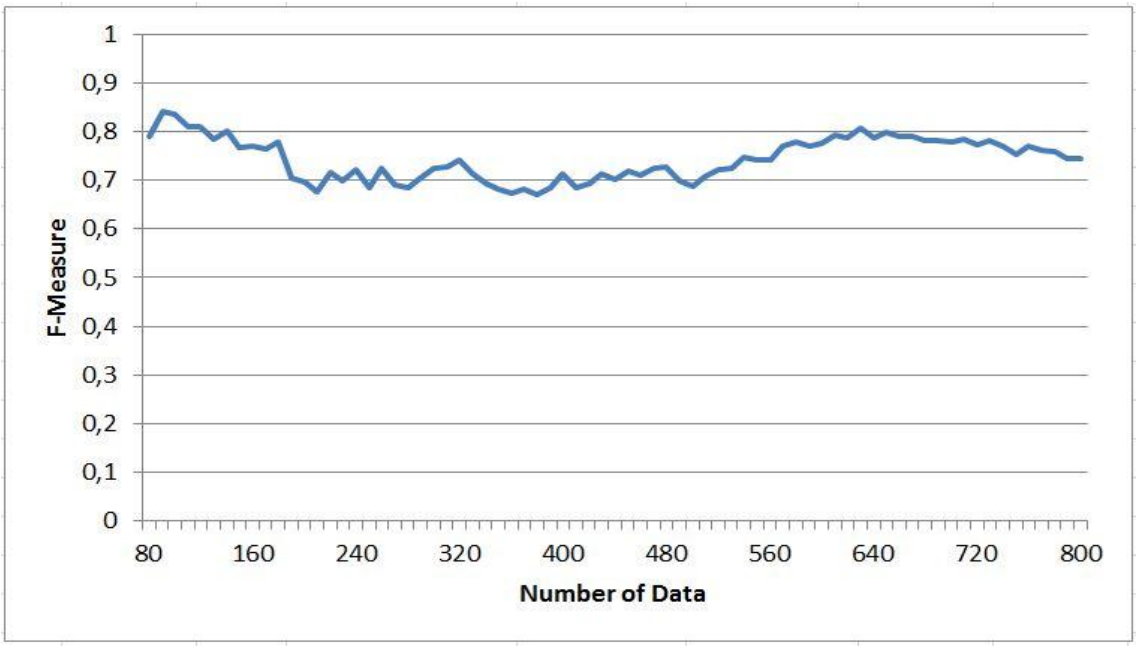
1. Deneyde, Text2arff yazılımı ile elde edilen eğitim kümeleri WEKA kullanılarak deneylerde kullanılan sınıflandırma algoritmaları tarafından öğrenilmektedir. Bu öğrenme işlemi sonunda elde edilen ortalama f-measure sonuçları Şekil 7.1, Şekil 7.2, Şekil 7.3 ve Şekil 7.4'de yer almaktadır. Bu sonuçlara göre sınıflandırıcıların karşılaştırılması ve performanslarının değerlendirilmesi yapılmaktadır.



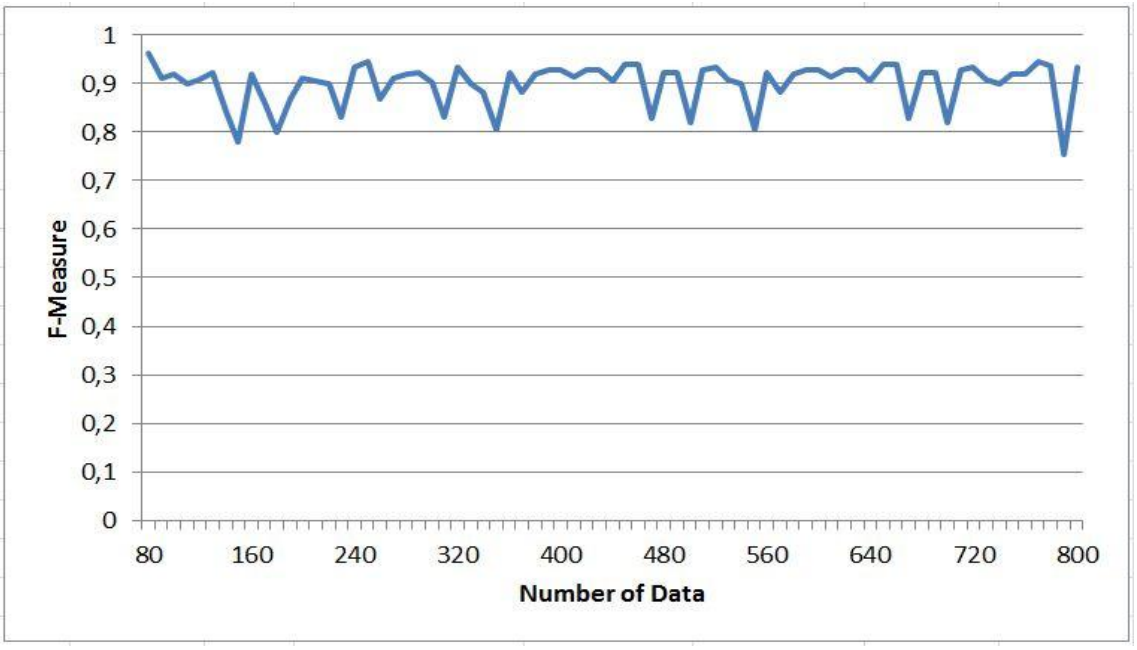
Şekil 7.1. Naïve Bayes sınıflandırıcının 1. deney f-measure sonuçları



Şekil 7.2. J48 sınıflandırıcının 1. deney f-measure sonuçları



Şekil 7.3. IBK sınıflandırıcının 1. deney f-measure sonuçları



Şekil 7.4. SVM sınıflandırıcının 1. deney f-measure sonuçları

Tablo 7.2'de deneylerde kullanılan sınıflandırıcıların öğrenme işlemi sonunda ki en yüksek ve en düşük doğruluk oranları gösterilmektedir. Buna göre NaïveBayes sınıflandırıcının örnekleri doğru bir biçimde sınıflandırma oranı diğer sınıflandırıcılara göre daha yüksektir.

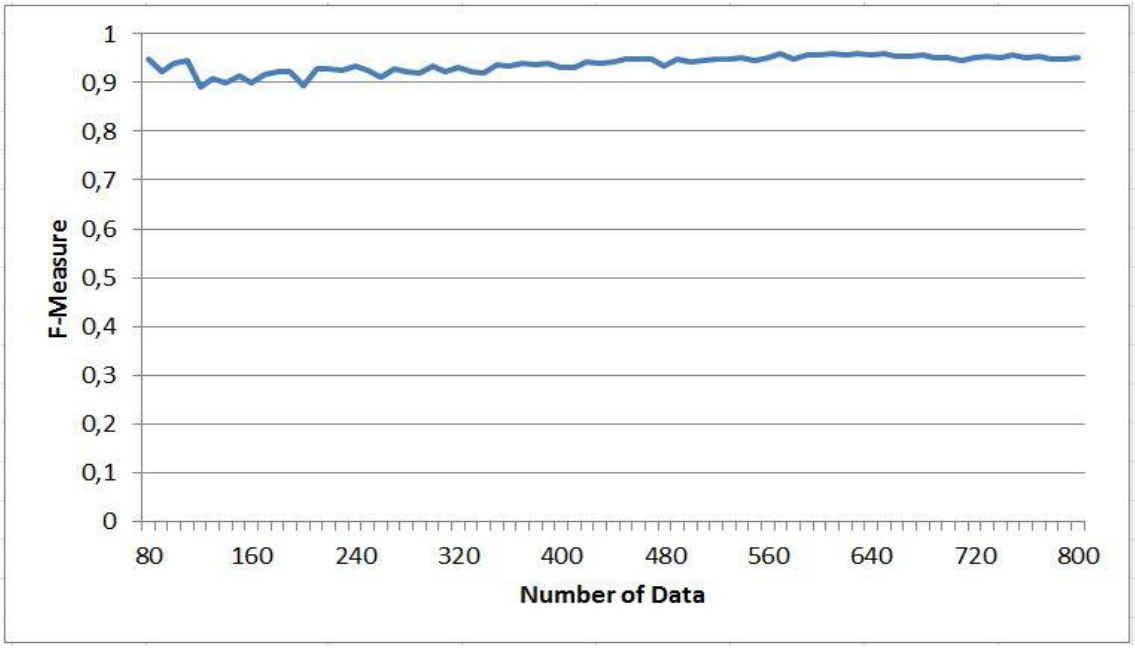
Tablo 7.2. Birinci deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları

Doğru Biçimde Sınıflandırılmış Örneklerin Yüzdesi	NaïveBayes	J48	SVM	IBK
Maksimum (Max)	96,8852	94,1538	96,25	84,4444
Minimum (Min)	85,7143	76,25	75,5932	69,2105

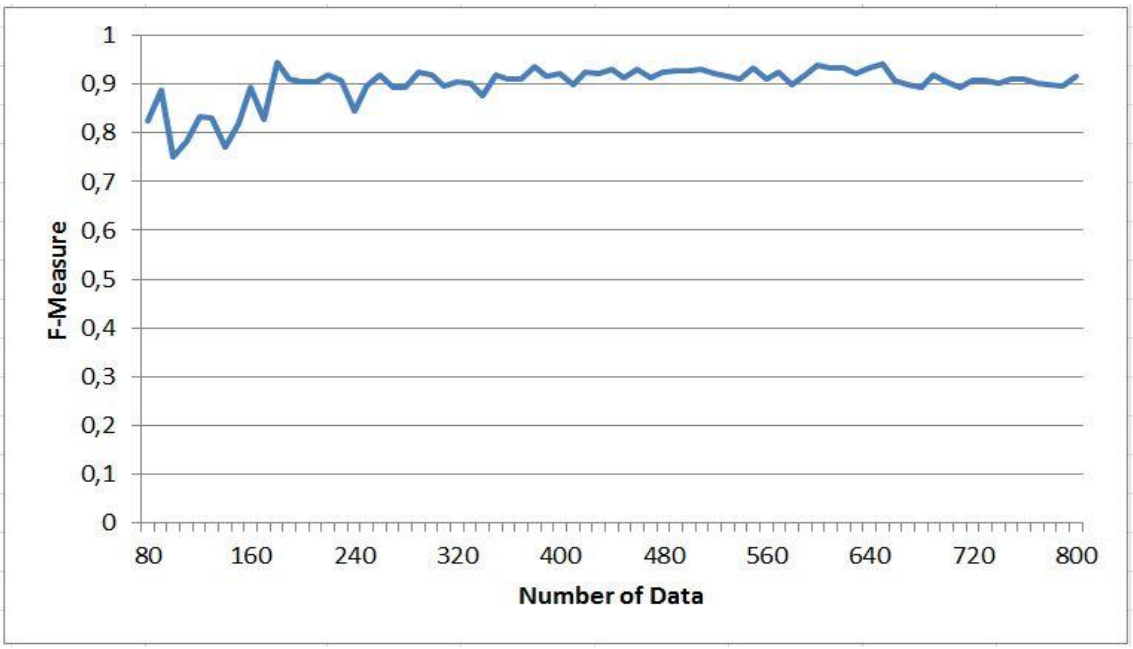
Yukarıdaki algoritmalarda tüm parametreler varsayılan değerleri ile kullanılmıştır. Sadece k en yakın komşu algoritmasında k değeri dört olarak seçilmiştir.

Algoritmalar arasındaki en yüksek başarı ve stabil çalışan Naïve Bayes'dir. Destek vektör makinelerinde stabil, fakat dalgalanma yapısının varlığı gözlenmiştir. Öğrenme oranı ile ikinci sırada yer alan algoritmadır. Karar ağaçları, destek vektör makinelerinden daha stabildir. Öğrenme oranı ile üçüncü sırada yer almaktadır. En kötü öğrenme yapısı K en yakın komşu algoritmasında yer almaktadır. K en yakın komşu algoritmasının en iyi değeri k = 4 seçilmiştir.

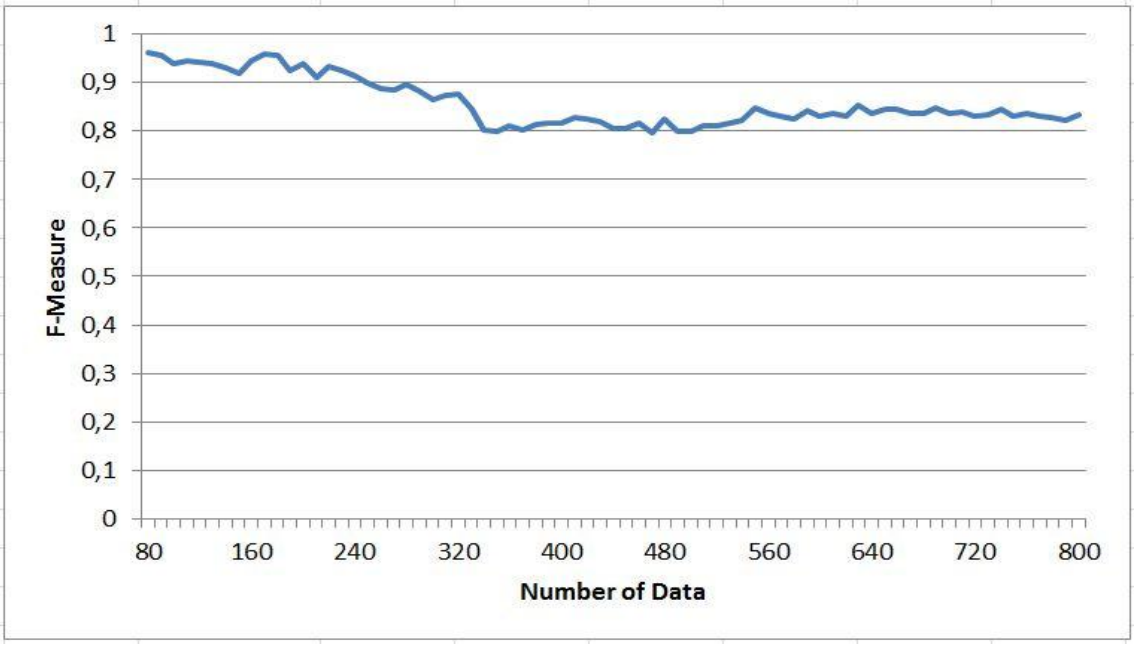
2. Deneyde, öğrenme işlemi sonunda elde edilen ortalama f-measure sonuçları Şekil 7.5, Şekil 7.6, Şekil 7.7 ve Şekil 7.9'de yer almaktadır. Bu sonuçlara göre sınıflandırıcıların karşılaştırılması ve performanslarının değerlendirilmesi yapılmaktadır.



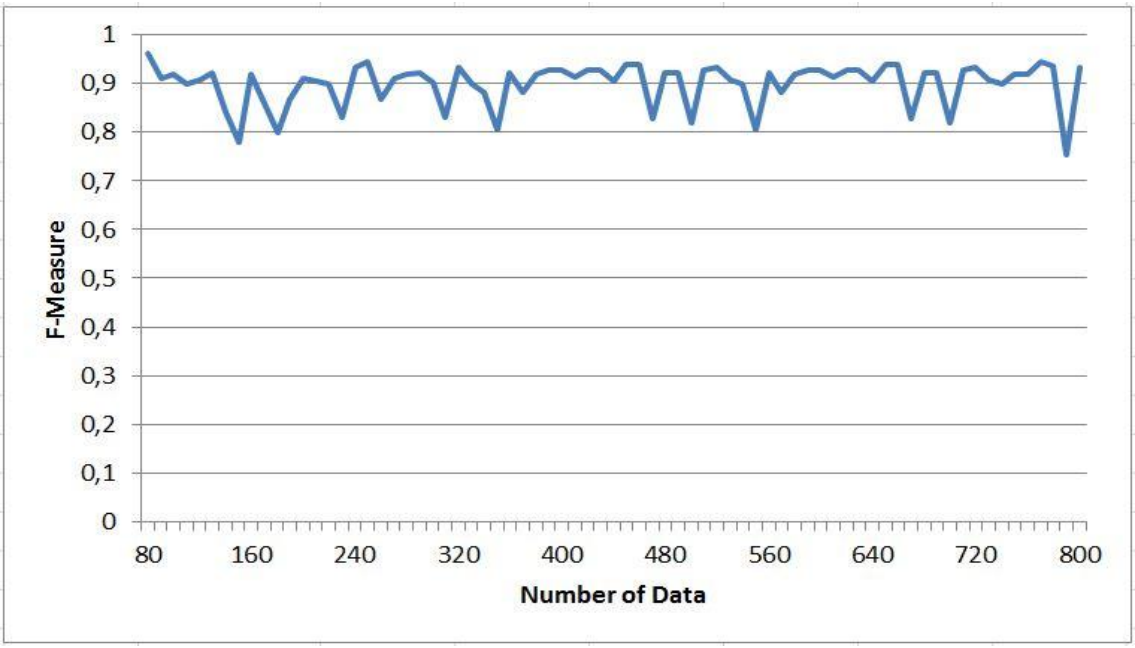
Şekil 7.5. Naïve Bayes sınıflandırıcının 2. deney f-measure sonuçları



Şekil 7.6. J48 sınıflandırıcının 2. deney f-measure sonuçları



Şekil 7.7. IBK sınıflandırıcının 2. deney f-measure sonuçları



Şekil 7.8. SVM sınıflandırıcının 2. deney f-measure sonuçları

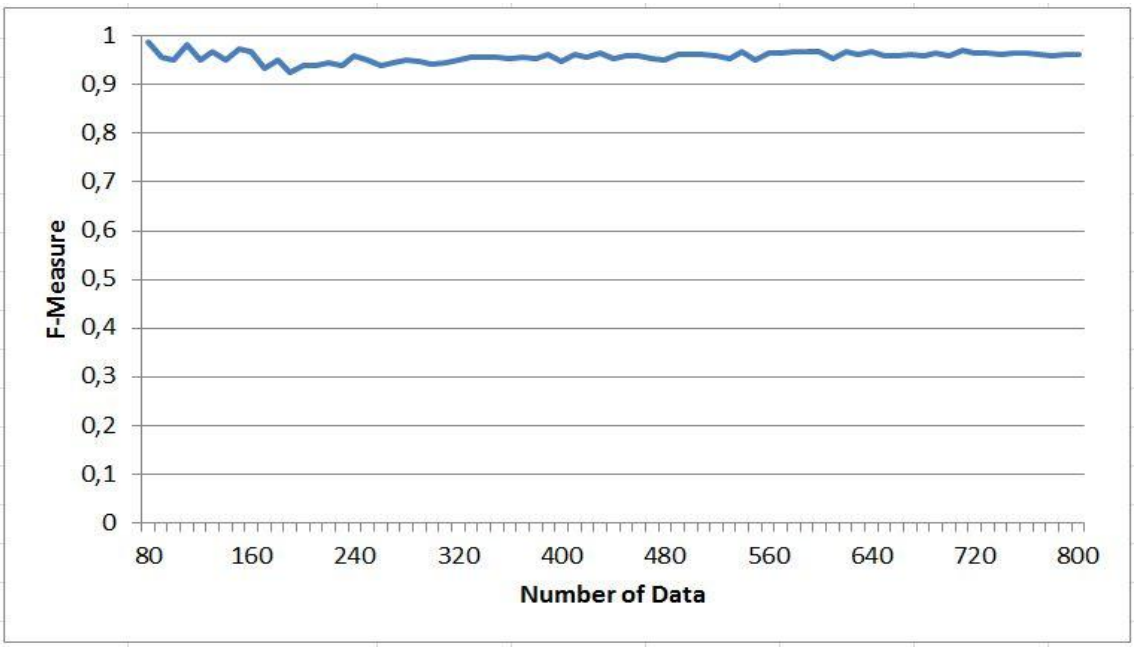
Tablo 7.3'te deneylerde kullanılan sınıflandırıcıların öğrenme işlemi sonunda ki en yüksek ve en düşük doğruluk oranları gösterilmektedir. Buna göre SVM sınıflandırıcının örnekleri doğru bir biçimde sınıflandırma oranı diğer sınıflandırıcılara göre daha yüksektir.

Tablo 7.3. İkinci deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları

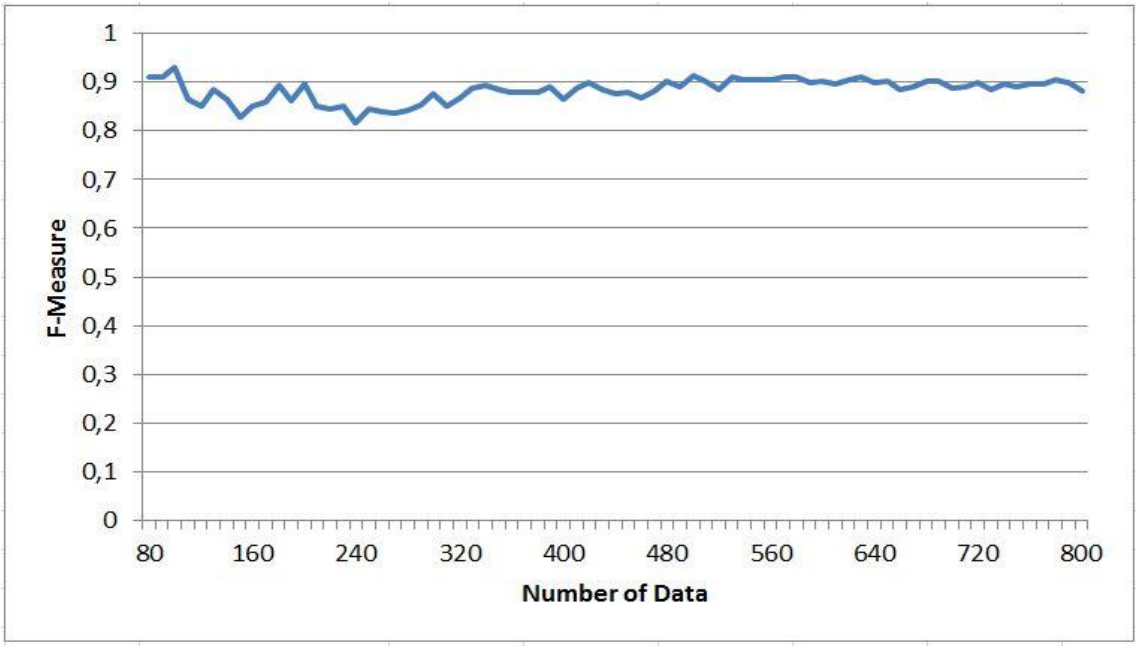
Doğru Biçimde Sınıflandırılmış Örneklerin Yüzdesi	NaïveBayes	J48	SVM	IBK
Maksimum (Max)	95,9649	94,4444	96,25	96,25
Minimum (Min)	89,1667	75	75,5932	80,4255

Algoritmalar arasındaki en yüksek başarı ve stabil çalışan NaiveBayes'dir. Destek vektör makinelerinde yine dalgalanma çok fazla gözlenmiştir. Karar ağaçları, frekans yükseltilmesinden dolayı daha başarılı bir sonuç vermiş ve daha stabil çalışmıştır. K en yakın komşu algoritması daha iyi bir sonuç vermiş ilk sınıflarda yüksek sonuçlar biraz düşmüş fakat devam eden sürede Naïve Bayes gibi başarılı bir sonuç vermiştir. K en yakın komşu algoritması değeri k = 7 seçilmiştir.

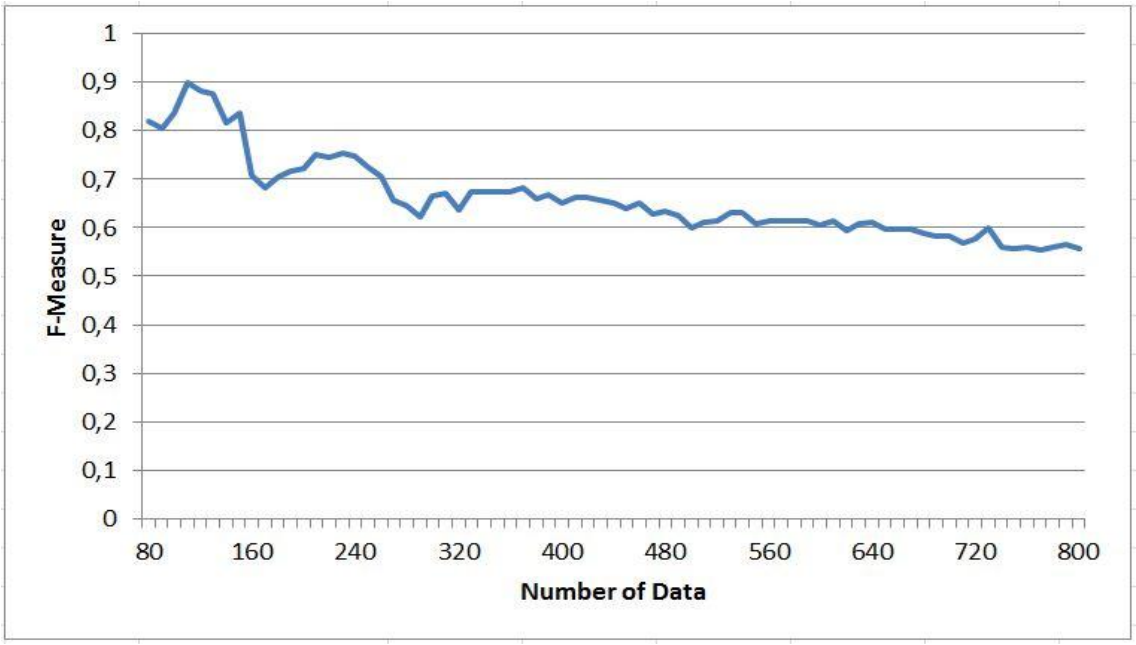
3. Deneyde, öğrenme işlemi sonunda elde edilen ortalama f-measure sonuçları Şekil 7.9, Şekil 7.10, Şekil 7.11 ve Şekil 7.12'de yer almaktadır. Bu sonuçlara göre sınıflandırıcıların karşılaştırılması ve performanslarının değerlendirilmesi yapılmaktadır.



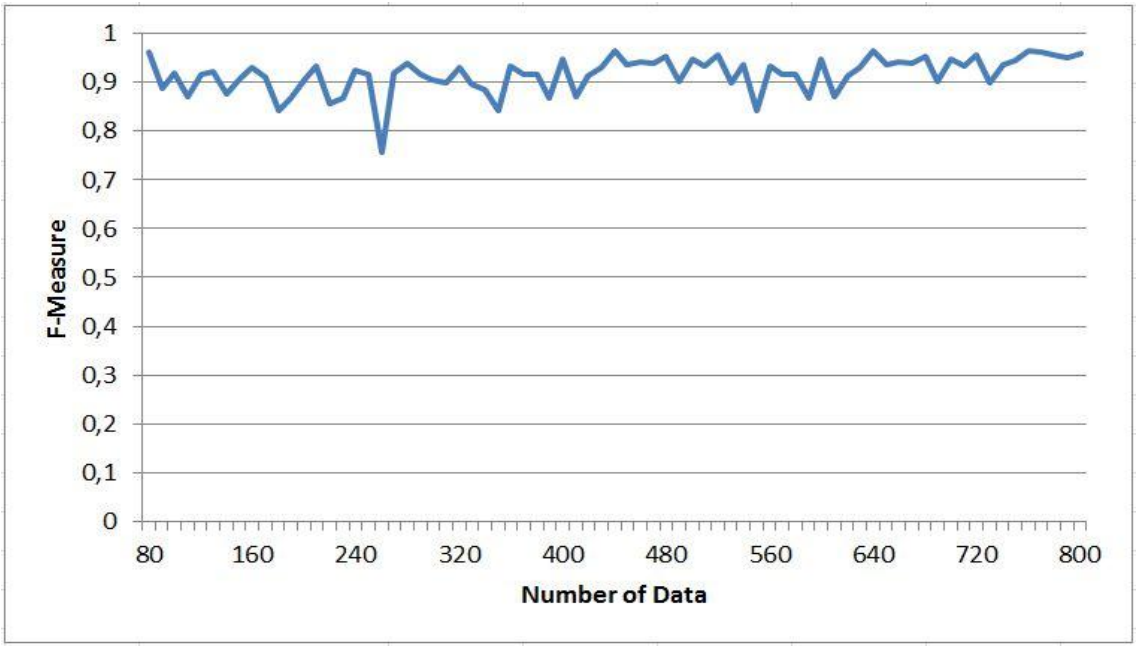
Şekil 7.9. Naïve Bayes sınıflandırıcının 3. deney f-measure sonuçları



Şekil 7.10. J48 sınıflandırıcının 3. deney f-measure sonuçları



Şekil 7.11. IBK sınıflandırıcının 3. deney f-measure sonuçları



Şekil 7.12. SVM sınıflandırıcının 3. deney f-measure sonuçları

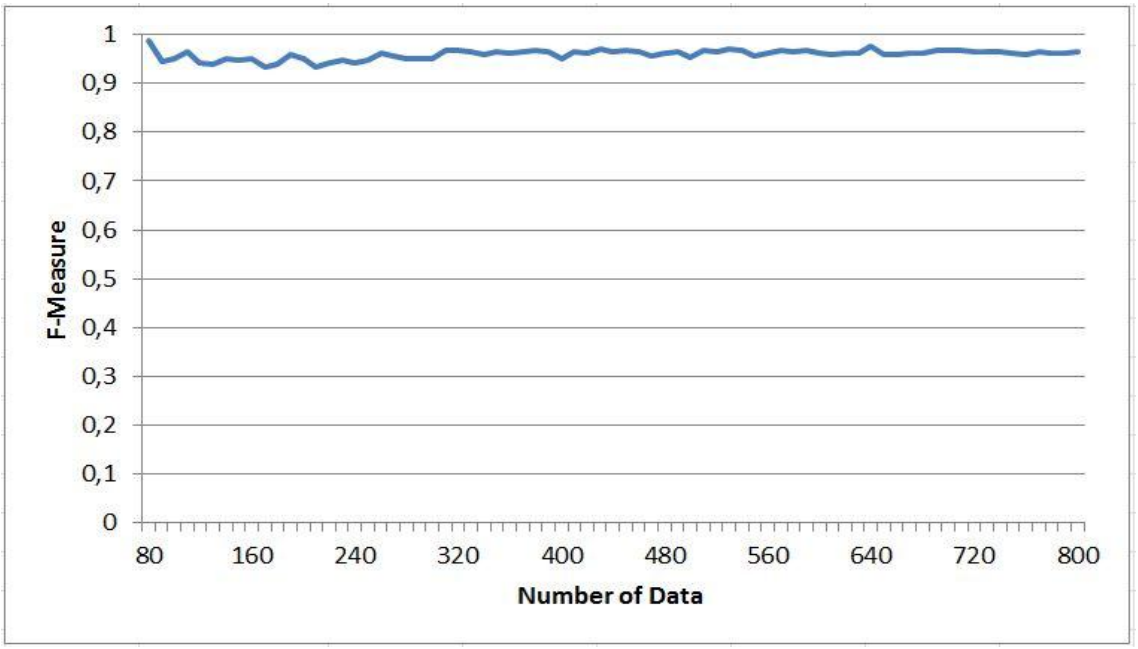
Tablo 7.4'te deneylerde kullanılan sınıflandırıcıların öğrenme işlemi sonunda ki en yüksek ve en düşük doğruluk oranları gösterilmektedir. Buna göre NaiveBayes sınıflandırıcının örnekleri doğru bir biçimde sınıflandırma oranı diğer sınıflandırıcılara göre daha yüksektir.

Tablo 7.4. Üçüncü deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları

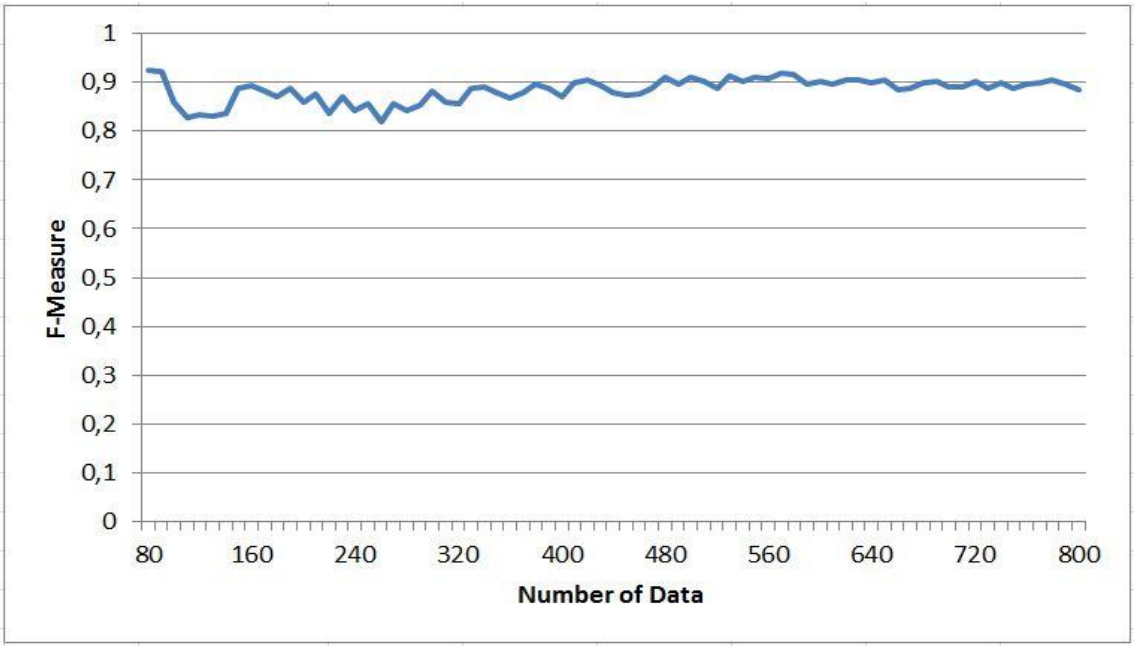
Doğru Biçimde Sınıflandırılmış Örneklerin Yüzdesi	NaïveBayes	J48	SVM	IBK
Maksimum (Max)	98,75	93	96,5909	90
Minimum (Min)	92,6316	81,6667	76,1538	61,8182

N-gram yerine words'ler kullanılmıştır. Algoritmalar arasındaki en yüksek başarı ve stabil çalışan NaïveBayes'dir. Destek vektör makinelerinde yine dalgalanma çok fazla gözlenmiştir. Karar ağaçları, dolayı daha başarılı bir sonuç vermiş ve daha stabil çalışmaktadır. K en yakın komşu algoritması bir düşüşe geçmiş ve şimdiye bu çalışmada alınan en kötü bir sonucu vermektedir. K en yakın komşu algoritması değeri k =1 seçilmiştir.

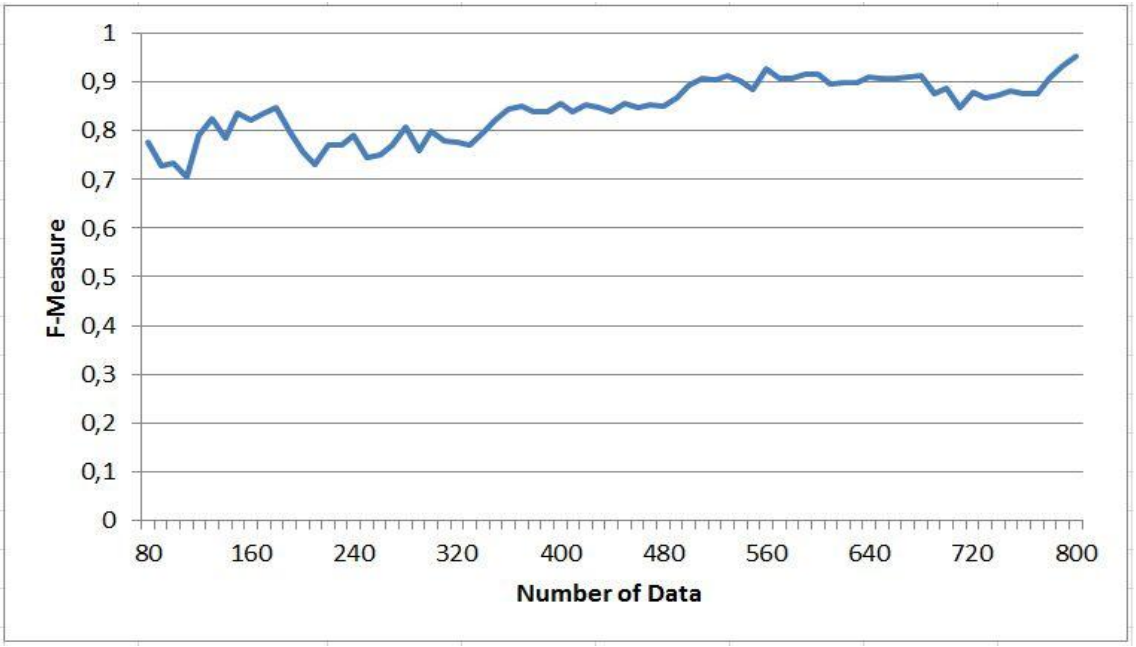
4. Deneyde, öğrenme işlemi sonunda elde edilen ortalama f-measure sonuçları Şekil 7.13, Şekil 7.14, Şekil 7.15 ve Şekil 7.16'da yer almaktadır. Bu sonuçlara göre sınıflandırıcıların karşılaştırılması ve performanslarının değerlendirilmesi yapılmaktadır.



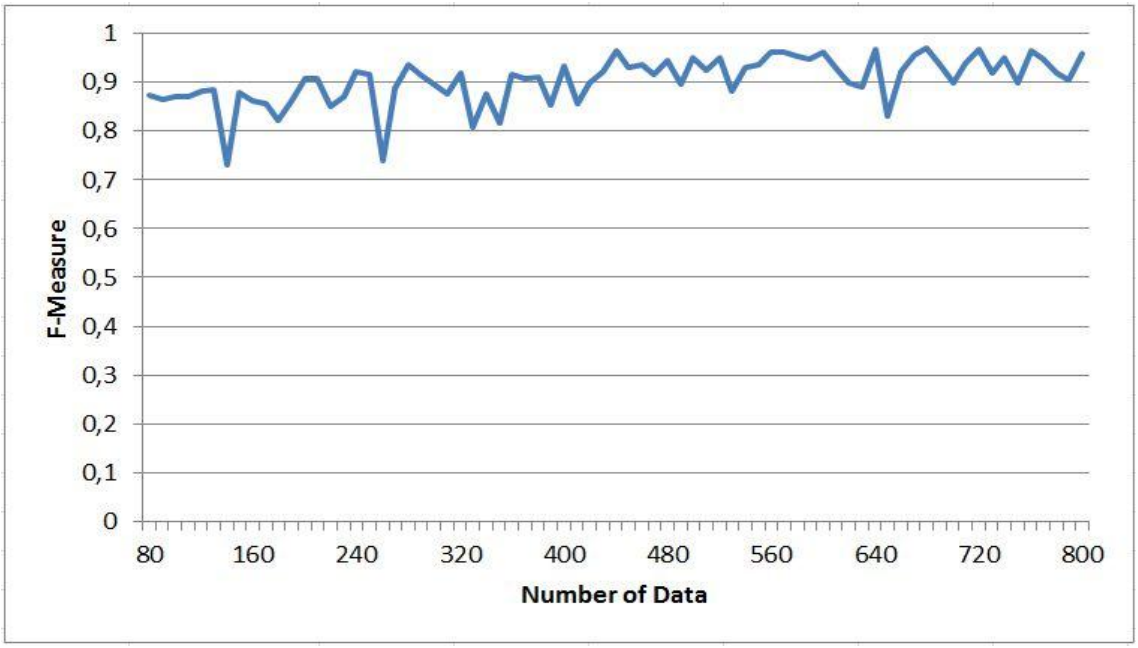
Şekil 7.13. Naïve Bayes sınıflandırıcının 4. deney f-measure sonuçları



Şekil 7.14. J48 sınıflandırıcının 4. deney f-measure sonuçları



Şekil 7.15. IBK sınıflandırıcının 4.deney f-measure sonuçları



Şekil 7.16. SVM sınıflandırıcının 4.deney f-measure sonuçları

Tablo 7.5'te deneylerde kullanılan sınıflandırıcıların öğrenme işlemi sonunda ki en yüksek ve en düşük doğruluk oranları gösterilmektedir. Buna göre NaïveBayes sınıflandırıcının örnekleri doğru bir biçimde sınıflandırma oranı diğer sınıflandırıcılara göre daha yüksektir.

Tablo 7.5. Dördüncü deneye göre sınıflandırıcıların maksimum ve minimum doğruluk oranları

Doğru Biçimde Sınıflandırılmış Örneklerin Yüzdesi	NaïveBayes	J48	SVM	IBK
Maksimum (Max)	98,75	92,5	96,9118	95,375
Minimum (Min)	93,3333	81,9231	73,5714	72,7273

N-gram yerine words'ler kullanılmaya devam edilmiş arff dosyaları oluşturulurken frequency değeri elli olarak değiştirilmiştir. Algoritmalar arasındaki en yüksek başarı ve sabit öğrenme değeri NaïveBayes'dir. Destek vektör makinelerinde dalgalanma azalmakta ve başarı oranı en yüksek seviyeye çıkmaktadır. Karar ağaçlarındaki başarının en az olduğu kısımdır. Fakat öğrenme bir noktadan sonra artarak devam etmektedir. K en yakın komşu algoritma başarısı 2.deneyde olduğu gibi yüksektir. Bu algoritma en düşük öğrenme düzeyinden başlayarak başarılı bir artış göstermiştir. K en yakın komşu algoritması değeri k =12 seçilmiştir.

VSE metin sınıflandırma mimarisinin en yüksek öğrenmesini NaïveBayes algoritması words ağırlıklandırması kullanılarak vermiştir. Fakat k değeri değişerek hazırlanmış words ağırlıklandırması için yüksek öğrenme oranı değişmemiştir.

VSE metin sınıflandırma mimarisinde 4.deney ile hazırlanan eğitim verileri üzerinde K en yakın komşuluk sınıflandırıcısının mutluluk eğrisi istenilen düzeyde gerçekleşmiştir. Ancak başarı oranı NaïveBayes kadar yüksek değildir.

Tüm bu sonuçlar analiz edildiğinde words özellik çıkarım yöntemi ile hazırlanan eğitim verileri üzerinde NaïveBayes sınıflandırıcının 98,75% oranında doğruluk sağlaması ve f-measure/veri sayısı grafiğinin yani mutluluk grafiğinin çok az dalgalı yapısından dolayı tezde geliştirilen VSE mimarisinde 4. deneysel çevre ve NaïveBayes sınıflandırıcısı temel alınmıştır. Böylelikle eğitim verilerinin hazırlanması sürecinde özellik çıkarım yöntemi olarak words, ağırlıklandırma yöntemi olarak idf, frekans değeri olarak 50 ve son olarak tekrar sayısı olan k için 50 değeri seçilmiştir. Daha sonra

eğitim verilerinin öğrenilmesi sürecinde sınıflandırıcı olarak NaïveBayes kullanılmış ve WEKA ortamında bu sınıflandırıcının varsayılan parametreleri kullanılmıştır. Ayrıca mutluluk grafiği bize sınıflandırıcının performansı hakkında önemli bilgiler sunmaktadır. Buna göre 800 adet veri yerine yaklaşık olarak 550 adet eğitim verisi ile sınıflandırıcı daha önceki doğruluk oranını yaklaşık olarak yakalamakla beraber performansında da önemli oranda artış yaşanmaktadır.

8. AĞ TABANLI VSE SİSTEM MİMARİSİ

Tezin bu bölümünde "hareket halindeki veriler" yani ağ üzerinden iletilen veriler üzerinde geliştirilen VSE sistem mimarisi anlatılmaktadır. VSE yöntemlerinin hareket halindeki verilere uygulanmasında ICAP protokolünden yararlanılmıştır. ICAP, HTTP protokolü ile birlikte içerik uyarlama ve filtrelemede kullanılmaktadır. Tezin bu bölümünde yapılan çalışmalar ICAP protokolünü ve VSE yöntemlerini kullanarak hareket halindeki veriyi analiz etmek ve Web üzerinden veri sızıntısını engellemeye yöneliktir.

8.1. M-ICAP Mimarisi

M-ICAP, veri sızıntısı engelleme amacıyla yapılmış yalnızca tek bir REQMOD servisine sahip RFC 3507 uyumlu bir ICAP sunucusudur. M-ICAP multi-threaded bir uygulama olup 4 adet thread'den oluşmaktadır. HTTP ve -sslBump yardımıyla- HTTPS protokolleri üzerinde çalışabilmektedir. ICAP protokolünü kullanarak "Data In Motion (Hareketli Veri)" üzerinde bir dizi kontrol gerçekleştirip hassas veriyi tespit edebilmekte; web arayüzünden düzenlenebilen ayarlara göre veri transferini engelleyebilmekte ve bu yetkisiz işlemleri kayıt altında tutabilmektedir. M-ICAP, C dili ile yazılmış olup, GCC (Gnu Compiler Collection) 4.6.1 derleyicisi ile derlenmiştir ve Linux ortamında çalışmaktadır. Testler Ubuntu 11.10 (32 bit) işletim sisteminde yapılmıştır. Veritabanı olarak MySQL 5.x kullanılmıştır.

8.1.1. Kullanılan Ek Kütüphaneler

- POSIX pthread ve semaphore kütüphaneleri,
- MySQL C API (MySQL kütüphanesi): http://dev.mysql.com/,
- PCRE düzenli ifadeler kütüphanesi: http://pcre.org,
- zlib sıkıştırma kütüphanesi: http://zlib.net,
- Zzip sıkıştırma yardımcı kütüphanesi: http://zziplib.sourceforge.net,
- Aladdin Enterprises Md5 kütüphanesi: http://sourceforge.net/projects/libmd5-rfc/files/,

- C Algorithms Trie kütüphanesi: http://sourceforge.net/projects/c-algorithms/.

8.1.2. Kullanılan Yardımcı Program

- Squid 3.1.14 Web Proxy: http://www.squid-cache.org/

8.1.3. Thread Mimarisi

Main Thread: main() fonksiyon bloğu ile diğer sunucu fonksiyonlarının bulunduğu, select giriş-çıkış çoklama fonksiyonunun çalıştığı thread dır. Aynı zamanda program açılışında (main fonksiyonu select döngüsüne girmeden önce) diğer threadları oluşturup başlatmakta görevlidir. Sonsuz döngü içerisinde select ile her saniye soketleri kontrol etmekte ve ICAP isteği geldiğinde bunları bir kuyruk veri yapısına eklemek suretiyle Request Handler Thread a aktarmaktadır.

Accept Thread: Dinleyici soketten gelen bağlantı isteklerini kabul edip ana dosya tanımlayıcı setine eklemekte ve istemciye özel bir tampon bellek alanı hazırlamaktadır.

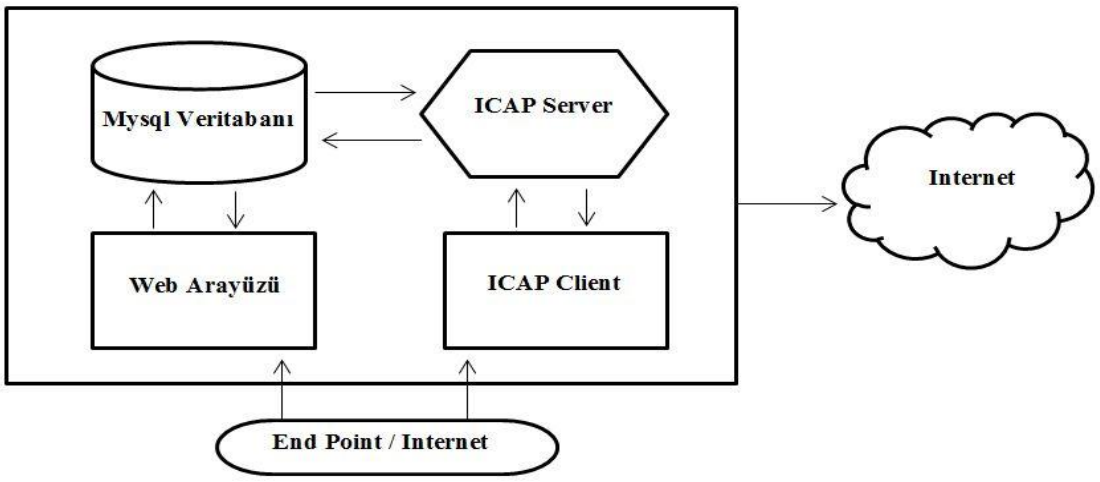
Controller Thread: Bu thread ın görevi zaman aşımına uğrayan soketleri kapatıp dosya tanımlayıcı setinden silmektedir. Zaman aşımı süresi varsayılan olarak 180 saniye olup web arayüzünden ayarlanabilmektedir.

Request Handler Thread: Ana sunucu tarafından kuyruğa eklenen ICAP isteklerini alıp işleyen ve buna göre bir cevap oluşturup yollayan thread'dır. Hassas verinin bulunup engellenmesi bu thread da gerçekleştirilmektedir.

Threadları senkronlamak için POSIX semaphore kütüphanesinden ve POSIX pthread kütüphanesindeki reader-writer lock'lardan yararlanılmıştır.

8.2. M-ICAP Programının Çalışma Şekli

select() giriş çıkış çoklama fonksiyonu kullanılarak main bloğu içerisinde, hangi soketler de hareketlilik olduğu tespit edilmektedir.



Şekil 8.1. M-ICAP Tasarım Şeması

Tezde geliştirilen VSE mimarisi uygulamasında, ICAP istemcisi tarafından ilgili istemciye ait soket aracılığıyla gönderilen ve ICAP isteklerinin bir parçası olan veriler, main thread da bulunan main() fonksiyonundaki select döngüsü aracılığıyla ICAP İstek Kuyruğu'na eklenmektedir. Semaphore nesnesinin sayacı artırılarak Request Handler Thread harekete geçirilmektedir. Request Handler Thread kuyruğun başındaki veriyi alarak ilgili istemciye ayrılmış tampon belleğe eklemektedir. Daha sonra ICAP isteğinin tamamlanıp tamamlanmadığını kontrol etmektedir. İsteğin tamamlanıp tamamlanmadığını kontrol etmek için istemciye ait tampon değişkendeki mesajın geçerli bir RFC 3507 (ICAP) mesajı olup olmadığına bakmaktadır. Bunu yaparken de RFC 3507 de verilen BNF gramerini kullanılmaktadır.

Kontrolden sonra, eğer istek tamamlanmışsa, aynı thread tarafından işlenmeye başlamaktadır. Bunun için ilk önce ICAP isteğindeki, istek satırı ayrıştırılmaktadır. Buradan METHOD değeri alınarak desteklenip desteklenmediğine bakılmaktadır. M-ICAP, verinin dışarıya sızmasını engelleme amaçlı bir yazılım olduğu için yalnızca HTTP isteklerinin kaynak sunucuya ulaşması sırasında çalışmaktadır. Dolayısıyla yalnızca REQMOD servisine sahiptir. Diğer servislere başvurulduğunda "501 - Method

Not Implemented" hatası oluşturarak istemciye döndürülmektedir. Daha sonra istek satırı içerisinde ki URI ve ICAP sürümü öğeleri ayrıştırılıp, programın servisiyle uyuşup uyuşmadığı kontrol edilmektedir. Bu işlemler sırasında bir sorun oluşmaz ise bir sonraki adıma geçilmektedir.

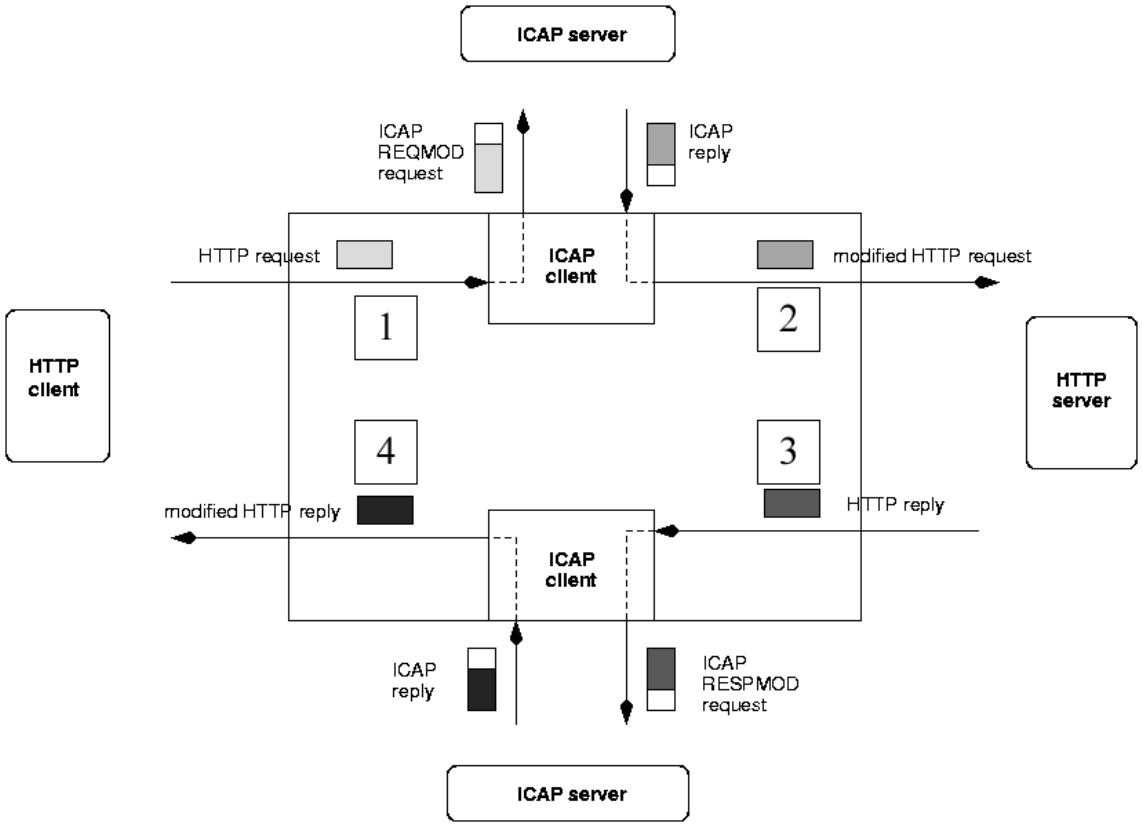
Bu adımda ICAP isteğinin başlıkları (headers) ayrıştırılmakta ve bir bağlı listede "alan:değer" şeklinde tutulmaktadır.

Son adımda ise eğer varsa mesajın gövdesi (body) ayrıştırılmaktadır. Mesajın gövdesinin olup olmadığı sınanırken:

Öncelikle Encapsulated başlığı, ayrıştırılan başlıklar içerisinde aranmaktadır. Eğer bulunamazsa mesajın gövdesi yoktur.

Eğer Encapsulated başlığı bulunduysa fakat req-body yerine null-body direktifi içeriyorsa ICAP mesajının gövdesi mevcut fakat kapsüllenen HTTP mesajının gövdesi yoktur. Bu durumda parçacıklı transfer kodlaması çözme mekanizmasının kullanılmasına gerek yoktur.

Eğer Encapsulated başlığı req-body direktifi içeriyorsa, parçacıklı transfer kodlaması çözme mekanizması çalıştırılmaktadır. Parçacık boyutları ve parçacık sınırlayıcıları (CRLF) silinerek, mesaj normal hale getirilmektedir.



Şekil 8.2. M-ICAP Reqmod izinsiz içerik engelleme [17].

Mesajın ayrıştırılması tamamlandıktan sonra ICAP mesajı içerisinde kapsüllenmiş HTTP isteği çıkarılır ve yukarıdaki adımlar benzer şekilde ona da uygulanmaktadır.

Daha sonra hassas veri tanımlama işlemine geçilmektedir.

Hassas veriyi tanımlamada aşağıdaki yöntemler kullanılmaktadır:

1. MD5 parmak izi kontrolü,
2. Düzenli ifadeler aracılığıyla desen eşleştirme,
3. N-gram tabanlı otomatik doküman kategorilendirme.

8.2.1. Md-5 Parmak İzi Kontrolü

MD5 (Message-Digest algorithm 5), veri bütünlüğünü test etmek için kullanılan, Ron Rivest tarafından 1991 yılında geliştirilmiş bir kriptografik özet (tek yönlü

şifreleme) algoritmasıdır. Girdi verinin boyutundan bağımsız olarak 128 bitlik özetler üretmektedir. [16]

MD5 algoritmasının uygulanabileceği veri boyutunun bir sınırı olmadığından ve MD5 özetlerinin 128 bit olmasından dolayı, üretilen özetler eşsiz değildir.

MD5 parmak izi kontrolü'ndeki amaç, web üzerinden gönderilmekte olan medyanın yasaklı olup olmadığını tespit etmektir. Web arayüzü kullanılarak yasaklanmak istenilen dosyaların MD5 özeti önceden çıkarılmaktadır. Bir istemci web üzerinden veri yollamak istediğinde, yolladığı verinin MD5 özeti ile veritabanındakiler karşılaştırılır ve böylece verinin bu bağlamda "hassas" olup olmadığına karar verilmektedir.

8.2.2. Düzenli İfadeler Aracılığıyla Desen Eşleştirme

PCRE kütüphanesi kullanılarak Perl Uyumlu Düzeni İfadeler ile desen eşleştirme yapılarak verinin "hassas" olup olmadığına karar verilmesidir. M-ICAP programı bazı ön tanımlı desenlere sahiptir.

Bunlar:

- T.C Kimlik Numarası,
- Social Security Number, (SSN)
- Kredi Kartı Numarası,
- IBAN Numarasıdır.

Ayrıca kullanıcılar web ara yüzü aracılığıyla PCRE düzenli ifadelerini kullanarak kendi istedikleri desenleri yasaklayabilmektedir.

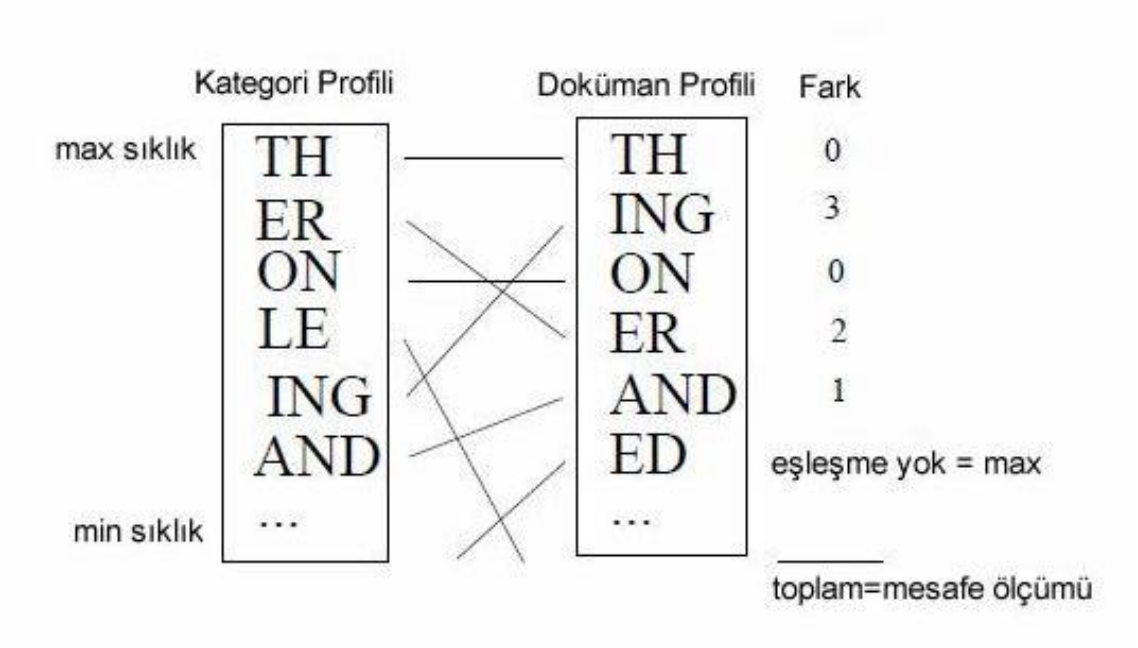
8.2.3. N-Gram Tabanlı Doküman Sınıflandırma

Geliştirilen VSE mimarisinde M-ICAP, hassas veri tanımanın son aşaması olarak n-gram tabanlı doküman sınıflandırma yöntemini kullanmaktadır. Kullanılan yöntem, Cavnar ve Trenkle'nin "Measure profile distance" (profiller arası benzerlik ölçümü) yöntemi kullanılarak geliştirilen algoritmadır (Cavnar, 1994).

M-ICAP programı başladığında tüm kategori profillerini veritabanından hafızaya yüklemektedir. Veri yapısı olarak erişimin az zaman olması ve uygulama kolaylığı açısından TRIE (metin ağacı) veri yapısı kullanılmıştır.

Zipf kanunları, metinlerin veya belli oranlardaki konu başlıklarından seçilen metinlerin bir araya getirilmesiyle oluşturulan ve dilbilimsel çalışmalarda ana malzeme olarak kullanılan derlemlerin içerdiği kelimelere ait gözlenme sıklıkları üzerinden, dile ait yapısal özellikleri genelleyen ilkeler olarak tanımlanmaktadır. Zipf tarafından ortaya konan her kanun, doğal dilin farklı açılardan tanımlanmasına olanak tanımaktadır. Zipf'in kanununa göre, derlemde gözlenme sıklığı en yüksek olan kelimeler aynı zamanda en fazla sayıda anlamı karşılamaya aday; çok anlamlılık derecesi en yüksek olan kelimelerdir.

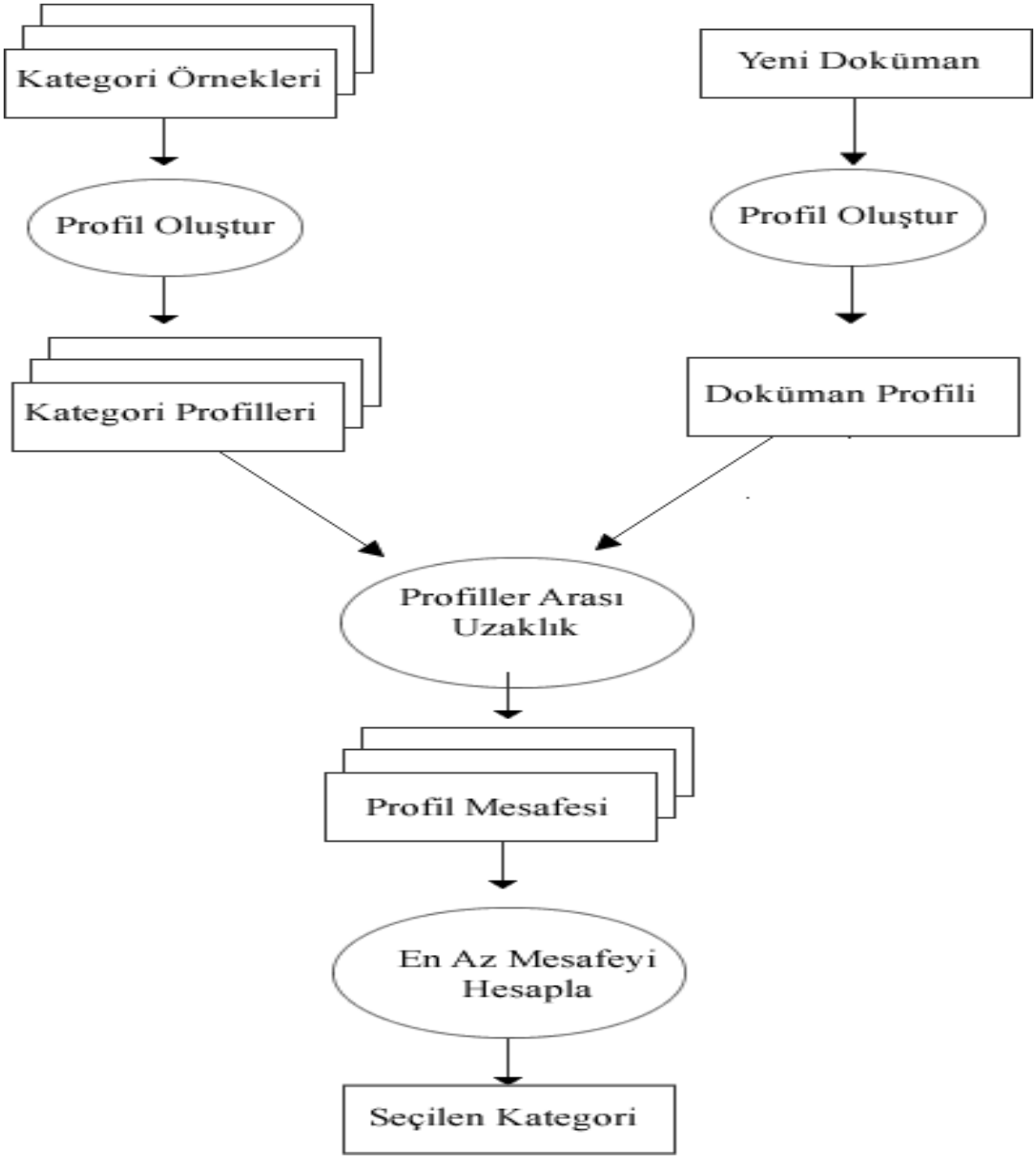
Profiller arasındaki benzerlik ölçümü işleminde, her bir kategorinin ve sınıflanacak olan her bir dokümanın n-gram profili hesaplanmaktadır. Profiller, bir dokümanda bulunan n-gram'ların en yüksek frekanstan en düşük frekansa sıralanmış halinden oluşmakta ve ayırt edici olduğu düşünüldüğünden yüksek frekanslı n-gram'lar (>10) kullanılmaktadır. Ölçüm aşamasında kategori ve sınıflandırılacak dokümanın profili olmak üzere iki n-gram profili alınmakta ve aralarındaki uzaklık için basit bir dizi-sıra istatistik hesabı yapılmaktadır. Bu ölçü, doküman profil içerisindeki bir n-gram'ın yerinin, kategori profil içerisindeki yerinden ne kadar uzakta olduğu bilgisini vermektedir. Doküman profilindeki her bir n-gram için, kategori profilinde onun benzeri bulunmakta ve aralarındaki uzaklığın mutlak değerce farkı hesaplanmaktadır.



Şekil 8.3. Benzerlik değerinin hesaplanması

Örnek benzerlik değerinin hesaplanması şekilde belirtilmiştir. Eğer doküman profilindeki herhangi bir n-gram, kategori profilinde bulunmuyorsa maksimum uzaklık verilmektedir. Basit algoritma aşağıdaki gibidir:

- Kategori için metin içerisinde harf dışında bütün karakterler yok edilmektedir,
- Kelimelerin başı ve sonunun bulunması için aralarına tire işareti koyulmaktadır,
- N-gramları hesaplanmaktadır,
- Frekansları oluşturulmaktadır,
- Sınıflandırılacak dokümanın n-gram ve frekans listesi hazırlanmaktadır,
- Kategorilerdeki n-gramlar ile sınıflandırılacak doküman karşılaştırılmakta ve eğer eşleşme bulunamazsa en yüksek değer ataması yapılmaktadır,
- Uzaklık listeleri oluşturulmaktadır,
- En az farka sahip olan seçilmektedir,
- Atama yapılmaktadır.



Şekil 8.4. N-gram tabanlı doküman sınıflandırma için veri akışı

N-gram tabanlı doküman sınıflandırma yönteminin son adımında doküman profilindeki her bir n-gram için, kategori profilindeki benzeri bulunmakta ve aralarındaki uzaklık değeri, doküman ve kategori profilinde hesaplanmaktadır.

Bir n-gram'ın profil frekans değerlerinin mutlak farkı ne kadar küçülürse profillerdeki benzerlik o kadar artmaktadır. Dokümanın profilinde bulunan tüm n-gram'lar için bu hesaplanan değerlerin toplamı alınarak, her bir kategori profili için bir

benzerlik ölçümü elde edilmektedir. Bu benzerlik ölçüm değerlerinden en küçük değere sahip olan kategori, test edilen dokümanın atanacağı sınıf olarak kabul edilmektedir.

8.3. Hassas Veriye Karşı Uygulanacak Davranış

Eğer MD5 parmakizi taraması sırasında, gönderilmek istenen içeriğin özeti veritabanındakilerle eşleşirse, desen eşleştirme sırasında gönderilen içerikte yasaklı desenlerden herhangi biri bulunuyorsa ya da n-gram algoritması kullanılarak kategorilendirme sırasında verinin tehlikeli olarak işaretlenen bir kategoriye ait olduğuna karar verilirse hassas veri tespit edilmiştir. Eğer verinin hassas olduğuna karar verilmişse, M-ICAP web arayüzünden seçilen davranış ayarına göre:

1) Hiçbir eylem gerçekleştirmemekte,

2) Veri sızıntısı kayıt altına almakta,

3) Veri sızıntısı engellenmekte,

4) 2. ve 3. seçenekler birlikte uygulanmaktadır.

Kayıt tutma işlemi sırasında istemcinin IP bilgisi ve kullanılan yönteme göre yasaklanan md5 özeti, desen ya da kategori bilgilerinden biri tarih ile birlikte ekrana ve veritabanına yazdırılmaktadır. Yasaklama işleminde ise vekil sunucu aracılığıyla istemciye “403 Forbidden” HTTP cevabı gönderilerek erişimin kısıtlandığı belirtilmektedir. Bu hata sayfası da web arayüzünden özelleştirilebilmektedir.

9. DEĞERLENDİRME

Literatür taramasında incelenen VSE akademik çalışmalarının tez ile karşılaştırılabilecek tek kısmı metin sınıflandırma sistemidir. VSE için uç nokta yönetimi ve ağ tabanlı sunucu çözümü hiçbir akademik çalışmaya konu olmamıştır. Ayrıca konu olan kısımları ise genel bilgi amaçlı olan taraflarıdır. Bu konuda genellikle özel sektör çözümleri mevcuttur. Özel sektör çözümleri ile tez içerinde oluşturulan VSE sisteminin genel farklılıkları aşağıda sıralanmıştır:

- Tez metin sınıflandırmada % 98,75 başarı elde edilmiştir,
- Diğer firmalara ait hiçbir çözüm kullanılmamış, tamamı özgün kod kullanılmıştır,
- ICAP sunucusunda metin sınıflandırma için Cavnar ve Trenkle tarafından geliştirilen profiller arası benzerlik ölçümü ile ilk kez VSE sistemlerine entegrasyonu sağlanmıştır,
- Kurumsal ağlarda oluşacak aşırı yükleme durumları için en hızlı cevabı verebilecek sunucunun tasarımı ve kodlanması yapılmıştır,
- Tüm ağ trafiğinin incelenmesini sağlayan tümleşik çözüm sunulmuştur.

Kurumsal yapının temel sorunlarından biri olan veri sızıntısını engelleme sistemi için geliştirilen mimari tezimizde üç kısım şeklinde incelenmiştir.

Sistem aşağıdaki özellikleri taşımaktadır:

- İstemci çözümü,
- Ağ tabanlı VSE çözümü,
- VSE sistemleri için metin sınıflandırma.

Bunlardan birincisi istemci tabanlı sistemlerde veri sızıntısını engellemedir. Sistem için Windows tabanlı istemciler kullanılmış ve C programlama dili ile yazılmış sürücü ile ssdt hook işlemi yapılmıştır. Bu işlem kullanıcı tarafından yaratılan, kopyalanan ve taşınan okunabilir içerikli dosyalarda düzenli ifadeler ile arama yapmakta, istenmeyen veri akışını engellemektedir. İstemci için konfigürasyon

dosyaları ağ tabanlı yönetim sunucusundan alınmaktadır. Program arayüzü işletim sisteminin kernel mode düzeyine inerek işlemleri yerine getirmektedir.

İkinci kısımda ağ tabanlı VSE çözümü sağlanması için birçok tasarım ve çözümler sağlanmıştır. Aşağıdaki bileşenlerden oluşmaktadır:

- Web tabanlı uygulama arayüzü,
- Uygulama veritabanı,
- ICAP İstemcisi,
- ICAP Sunucusu.

Web tabanlı uygulama arayüzü, php programlama dili ile yazılmıştır. ICAP sunucusunun ayarlarının yapılması, istemci konfigürasyon dosya ayarlarının yapılması ve metin sınıflandırma işleminin birinci basamağı için eğitimlerin yapılmasını sağlayan arayüzdür.

Uygulama veritabanı mysql ile oluşturulmuş ve tüm yapılandırma bilgilerinin üzerinde taşınmasını sağlamaktadır.

ICAP istemcisi kurumsal ağ trafiğini kontrol etmek için çalışmakta ve gerekli dosyaların incelenmesini sağlanması için ICAP sunucusunu kullanmaktadır. ICAP sunucusu trafik analizini sağlayacak kısımdır.

Tez süresince yapılan araştırmalar kompleks bir çözüm üretmek için odaklanmıştır. İstemci tabanlı çözüm, ağ tabanlı VSE sunucusu ve metin sınıflandırma işleminin tez içerisinde bir araya getirilmesi başarılı bir sonuç üretmiştir.

İstemci tabanlı çözümde hassas verinin engellenmesi amacıyla yapılmış, düzenli ifadelerin kullanıldığı bir metin analiz süreci yaşanmıştır. Düzenli ifadeler burada başarılı sonuç vermiştir. Ağ tabanlı sunucuda kullanılan n-gram tabanlı doküman analizi, istemci tabanlı sistemde de kullanılabilmektedir. Ancak ağ trafiğinin kontrolü yapıldığı ve istemciye tüm n-gram tanımlamalarının yüklenmesi sırasında ağ kesintisinin yaşanabileceği tehlikesi nedeni ile bu uygulamadan vazgeçilmiştir.

Metin sınıflandırma işleminde hassas veri analizleri weka yazılımında yapılmaktadır. Sınıflandırma işlemi Hart ve diğerleri tarafından yapılan “VSE için

metin sınıflandırma” ve %97 oranında başarı sağladıkları çalışma ile karşılaştırıldığında %98,75 başarı sağladığımız gözlemlenmektedir. ICAP sunucusunda uygulanacak sınıflandırma işlemi için NaïveBayes ve Cavnar ve Trenkle tarafından geliştirilen profiller arası benzerlik ölçümü algoritmaları iki aday olmuştur. ICAP sunucusunun kurumsal ağ trafiği göz önüne alındığında en hızlı cevap süresini yakalaması için Cavnar ve Trenkle tarafından geliştirilen profiller arası benzerlik ölçümü algoritması tercih edilmiştir. Sınıflandırma yöntemleri olarak J48, NaïveBayes, K – en yakın komşuluk, Destek vektör makineleri kullanılmıştır.

Ağ tabanlı VSE sunucusunda trafik analizi yapılırken, doküman sınıflandırma işlemi çok hassas bir şekilde yapılmaktadır. Parmak izi, n-gram tabanlı özellik çıkarım teknikleri ve düzenli ifadeler kullanılarak analizin yüksek başarı göstermesi sağlanmıştır. ICAP protokolü ile ağ üzerinde reklam gösterimi, içerik virüs taraması ve doküman analizi gibi çeşitli işlemler yapılmaktadır.

Veri sızıntı engelleme komple bir sistem olarak tasarlanmış ve istatistiksel analiz algoritmalarını kullanması nedeniyle yüksek başarılara ulaşmıştır.

10. KAYNAKLAR

Abbadi I. & Alawneh M. (2008). "*Preventing insider information leakage for enterprises*" in Emerging Security Information. Systems and Technologies. SECURWARE'08. Second International Conference on. pp. 99–106.

Adsız A. (2006). "*Metin Madenciliği*". Ahmet Yesevi Üniversitesi. Basılmamış Yüksek Lisans Bitirme Projesi. Fen Bilimleri Enstitüsü.

Al-Fedaghi S. (2011). "*A Conceptual Foundation for Data Loss Prevention*". International Journal of Digital Content Technology and its Applications. Volume 5. Number 3.

Alpaydın, E. (2010). "*Introduction to Machine Learning, 2nd edition*". The MIT Press, USA.

Amasyalı M. F., Davletov F., Torayew A., Çiftçi Ü. (2010). "*text2arff: Türkçe Metinler İçin Özellik Çıkarım Yazılımı*". SİU 2010.

Backes M. & Köpf B., Rybalchenko A. (2009). "*Automatic Discovery and Quantification of Information Leaks*". 30th IEEE Symposium on Security and Privacy.

Bhaskar R., Laxman S., Smith A., Thakurta A. (2010). "*Discovering Frequent Patterns in Sensitive Data*". KDD'10 Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Borders K. & Prakash A. (2009). "*Quantifying Information Leaks in Outbound Web Traffic*". 30th IEEE Symposium on Security and Privacy.

Cavnar W. B. & Trenkle J. M. 1994. "*N-gram-based text categorization*". Proceedings of SDAIR-94. 3rd Annual Symposium on Document Analysis and Information Retrieval. Information Systems Project Management. Jolyon E. Hallows. AMACOM Press.

Dinçer E. (2006). "*Veri Madenciliğinde K-Means Algoritması ve Tıp Alanında Uygulanması*". Yüksek Lisans Tezi. Kocaeli Üniversitesi Fen Bilimleri Enstitüsü. Kocaeli.

Doğan S. & Diri B. (2010). "*Türkçe Dokümanlar İçin N-Gram Tabanlı Yeni Bir Sınıflandırma (Ng-ind): Yazar, Tür ve Cinsiyet*". *TBV Bilgisayar Bilimleri ve Mühendisliği Dergisi*. Sayı:3. s. 11-21.

Domingos P. & Pazzani M. (1997). "*On the optimality of the simple bayesian classifier Under zero-one loss*". Machine Learning. 29:103-130.

Etges R. & McNeil K. (2006). "*Understanding Data Classification Based on Business and Security Requirements*". ISACA Journal. Vol. *5* , 1-8.

Gökçen H. (2007). "*Yönetim Bilgi Sistemleri*". Palme Yayıncılık. İstanbul. Türkiye.

Güven A., Bozkurt Ö.Ö., Kalıpsız O. (2007). "*Veri Madenciliğinin Geleceği*". Akademik Bilişim. Dumlupınar Üniversitesi. Kütahya.

Hart M., Manadhata P. K., Johnson R. (2011). "*Text Classification for Data Loss Prevention*". PETS'11 Proceedings of the 1th International Conference on Privacy Enhancing Technologies.

Hsu C., Chang C., Lin C. (2009). "*A Practical Guide to Support Vector Classification*", Department of Computer Science National Taiwan University. Taipei 106. Taiwan.

İlhan S., Duru N., Karagöz Ş., Sağır M. (2008). "*Metin Madenciliği İle Soru Cevaplama Sistemi*", ELECO 2008, s. 356-359.

Jackson P. & Moulinier I. (2002). "*Natural language processing for online applications: text retrieval, extraction, and categorization*". Amsterdam.

John G.H. & Langley P. (1995). "*Estimating Continuous Distributions in Bayesian Classifiers*.". Proceedings of the eleventh Conference on Uncertainty in Artificial Intelligence.

Kasparick H. (2008). "*Data Leakage Prevention, Fachhochschul-Masterstudiengang*". Sichere Informationssysteme in Hagenberg.

Kılıçaslan Y, Güner ES, Yıldırım S. (2009). "*Learning-based pronoun resolution for Turkish with a comparative evaluation*". Computer Speech & Language. 23(3): 311-331.

Kim J. & Kim H. J. (2010). "*Design of Internal Information Leakage Detection System Considering the Privacy Violation*". Information and Communication Technology Convergence.

Kocamaz A.F., Aydın F., Uçar E. (2011). "*Developing of a learning-based system to assist treatment process of arrhythmia patients*". Scientific Research and Essays. 6(32) pp. 6654-6665.

Kurta M. (2010). "*On The Road Towards Cloud Computing Services*". Master Thesis Economics & Informatics. Erasmus Universiteit Rotterdam.

Lawrence C.M. (2009). "*Data Leakage for Dummies*", Wiley Publishing.

Lee S., Lee K., Savoldi A., Lee S. (2009). "*Data Leak Analysis In A Corporate Environment*". Fourth International Conference on Innovative Computing. Information and Control.

Lee Y.C., Bishop S., Okhravi H., Rahimi S. (2009). "*Information Leakage Detection in Distributed Systems using Software Agents*". IEEE Symposium on Intelligent Agents.

Livingston, J. (2007). "*Tips and Strategies to Protect Laptops and the Sensitive Data They Contain*". Information Systems Control Journal. vol. 5.

Luft M. (2009). "*Can Data Leakage Prevention Prevent Data Leakage?*". University of Mannheim. Laboratory for Dependable Distributed Systems.

Manning C.D. & Schutze H. (1999). "*Foundations of Statistical Natural Language Processing*". [M]. Cambridge: MIT Press.

Manning C.D., Raghavan P., Schütze H. (2008). "*Introduction to Information Retrieval*". Cambridge UniversityPress. New York. 544p.

Marecki J., Srivatsa M., Varakantham P. (2010). "*A Decision Theoretic Approach to Data Leakage Prevention*". Research Collection School of Information Systems. Paper 617.

Mundada Y., Ramachandran A., Bin Tariq M., Feamster N. (2011). "*Practical DataLeak Prevention for Legacy Applications in Enterprise Networks*". Georgia Institute of Technology College of Computing.

Náther, P. (2005). "*N-gram based Text Categorization*". Diploma Thesis. Comenius University.

Papadimitriou P. & Garcia-Molina H. (2009). "*A Model for Data Leakage Detection*". IEEE International Conference on Data Engineering. 25th International Conference on Data Engineering.

Papadimitriou P. & Garcia-Molina H. (2011). "*Data Leakage Detection*". IEEE Transactions on Knowledge and Data Engineering. vol.23. no.1.

Raman P., Kayacık H. G., Somayaji A. (2011) "*Understanding Data Leak Prevention*". Annual Symposium on Information Assurance (Asia).

Redwood O. W. (2011). "*Apecs: A Dynamic Framework for Preventing and Mitigating Theft, Loss, and Leakage of Mission Critical Information in Trust Management Networks*". Master Thesis Department of Computer Science. Florida State University College of Arts and Sciences.

Ries C. (2006). "*Inside Windows Rootkits*". VigilantMinds.

Rost, J. & Glass, R. L. (2011). "*The Dark Side of Software Engineering*". Evil on Computing Projects. IEEE Computer Society. John Wiley & Sons, Inc. Hoboken. New Jersey, USA.

Russell S.J. & Norvig P. (2002). "*Artificial Intelligence: A Modern Approach 2nd ed*". Prentice Hall. Printed in the United States of America. ISBN 0 13 790395 2.

Soucy, P. & Mineau, W. (2005). "*Beyond TFIDF Weighting for Text Categorization in theVector Space Model*". IEEE International Conference. 1130-1135.

Ubois, J. (2007). "*Reinventing data loss prevention*". The Security Report.

Vachharajani, N., Bridges, M. J., Chang J., Rangan R., Ottoni G., Blome J. A., Reis G. A., Vachharajani M., August D. I. (2004). "*Rifle: An architectural framework for user-centric information-flow security*". in MICRO 37: Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture. Washington, DC, USA: IEEE Computer Society, pp. 243–254.

Vapnik, N. V. (1999). "*An overview of statistical learning theory*", IEEE Transactions on Neural Networks.

Velte A. T., Velte T. J., Elsenpeter R. (2010). "*Cloud Computing: A Practical Approach"*. McGraw-Hill.

Vercellis C. (2009). "*Business Intelligence; Data Mining and Optimization for Decision Making"*. John Wiley & Sons, Ltd. UK.

Wang L., ZHUANG Y., Pan L. (2006). "*Design and Implementation of File Watching System Based on Mandatory Access Control*". In: Computer Application. Vol.26 No.12.

Witten I.H & Frank E. (2005). "*Data Mining: Practical Machine Learning Tools and Techniques– 2nd ed. p. Cm*". Morgan Kaufmann Series in Data Management Systems. ISBN: 0-12-088407-0.

Xiaosong Z., Fei L., Ting C., Hua L. (2009). "*Research and Application of the Transparent Data Encpryption In Intranet Data Leakage Prevention*". International Conference on Computational Intelligence and Security.

İNTERNET KAYNAKLARI

[1] http://sourceforge.net/projects/weka/

[2] http://www.infowatch.com Nisan 2007 güvenlik bülteni

[3] Osterman Research, Inc. *The Case for Outbound Content Management.* April 2010.

[4] Deloitte. *2010 Financial Services Global Security Study: The faceless threat.* 2010.

[5] http://tr.wikipedia.org/wiki/HTTP

[6] http://tools.ietf.org/html/rfc2616

[7] http://tools.ietf.org/html/rfc1952

[8] http://tools.ietf.org/html/rfc1950

[9] http://tools.ietf.org/html/rfc1951

[10] http://tools.ietf.org/html/rfc1700

[11] http://tools.ietf.org/html/rfc2046

[12] http://tools.ietf.org/html/rfc1867

[13] http://tools.ietf.org/html/rfc822

[14] http://tools.ietf.org/html/rfc3507

[15] http://tr.wikipedia.org/wiki/HTTPS

[16] http://tr.wikipedia.org/wiki/MD5

[17] http://shweby.sourceforge.net/

[18] Torture Archive - http://www.aladin0.wrlc.org/gsdl/cgi-bin/library?c=torture&a=q

ÖZGEÇMİŞ

Adı Soyadı: Murat TOPALOĞLU

Adres: T.Ü. Keşan Meslek Yüksekokulu Keşan / Edirne

Öğrenim Durumu:

Derece	**Bölüm**	**Üniversite**	**Yıl**
Lisans	Computer Information System	Yakın Doğu Üniversitesi	2003
Y.Lisans	Bilgisayar Mühendisliği	Trakya Üniversitesi	2008
Doktora	Bilgisayar Mühendisliği	Trakya Üniversitesi	

Yüksek Lisans Tez Başlığı: Zihinsel engelli çocukların uzaktan eğitimi ve web tabanlı bir yazılım sisteminin geliştirilmesi

Danışman: Yrd.Doç.Dr. Özlem UÇAR

Görevler:

Görev Unvanı	**Görev Yeri**	**Yıl**
Yazılım Mühendisi	Beko A.Ş.	2003-2004
Bilgi İşlem Uzmanı	Harmanlı A.Ş.	2004-2006
Sistem Mühendisi	Mynet A.Ş.	2006-2007
Öğretim Görevlisi	Trakya Üniversitesi	2007-

Unvanları: Microsoft Sistem Engineer (MCSE) – Cisco Network Associates (CCNA)

Eserler Listesi:

Ulusal Bilimsel Toplantılarda Sunulan ve Bildiri Kitabında Basılan Bildiriler

Topaloğlu Özdemir A., Uçar E., **Topaloğlu M.,** Demiröz E., *Eğitilebilir Zihin Engelli Çocukların Uzaktan Eğitimi ve Uzaktan Eğitim Uygulamalarının Karşılaştırılması*. 18. Ulusal Özel Eğitim Kongresi. 13-15 Kasım 2008. Selçuk Üniversitesi Eğitim Fakültesi. Konya.

Uluslararası Hakemli Dergilerde Yayımlanan Makaleler

Topaloğlu Özdemir A., **Topaloğlu M.** *Distance Education Applications in Concept Acquisition for Disabled Indıviduals/Special Education for Handicapped*.Procedia-Social Science and Behavioral Sciences, Volume 1,Issue 1, Pages 1-2890. World Conference on Educational Sciences.4-7 Şubat 2009.Yakın Doğu Üniversitesi.Lefkoşa-Kıbrıs.

Uluslararası Bilimsel Toplantılarda Sunulan ve Bildiri Kitabında Basılan Bildiriler

Topaloğlu Özdemir A., **Topaloğlu M.,** *The Opinions of University Students About Computer-Based Education*., Science, Education and Time As Our Concern To Celebrate: The 45th Anniversary of Pedagogial Education in Smolyan and the 10th Anniversary of the Smolyan Branch of Plovdiv University, November 30th –December 1st 2007 Smolyan/Bulgaria.

Topaloğlu M., Topaloğlu Özdemir A*., Bilgisayar Destekli Eğitimin Öğretimde Kaliteyi Arttırmasına İlişkin Üniversite Öğrencilerinin Görüşleri*.,III.Uluslararası Eğitim Kongresi, 16-18 Nisan 2008,Girne-Kıbrıs.

Topaloğlu Özdemir A., **Topaloğlu M.** *The Attitudes Towards Computer Using To Candicates Of Preschool Teachers At The Preschool.* IV.International Conference-Contemporary Approaches in Educational Management. 13-15 Haziran 2008 Ohrid-Makedonia**.**

Topaloğlu M.,Umut,İ.(2011). *In Contemporary Education Understanding Resrtucturing The Programs Of The Vocational High Schools With Internet Based Processes.* September 2011 Ohrid/Makedonya.

Diğer Yayınlar

Topaloğlu M., *Zihinsel Engelli Çocukların Uzaktan Eğitimi ve Web Tabanlı Bir Yazılım Sisteminin Geliştirilmesi.* Yüksek Lisans Tezi. Trakya Üniversitesi: Fen Bilimleri Enstitüsü. Temmuz 2008.Edirne.