BİLGİ GÜVENLİĞİ VE SİBER GÜVENLİK KAPSAMINDA BAKANLIK UYGULAMALARI İÇİN GÜVENLİ YAZILIM GELİŞTİRME METODOLOJİSİ ÖNERİSİ

UZMANLIK TEZİ

HAZIRLAYAN: Gülizar Duygu KURT KAYA

ANKARA-2017

BİLGİ GÜVENLİĞİ VE SİBER GÜVENLİK KAPSAMINDA BAKANLIK UYGULAMALARI İÇİN GÜVENLİ YAZILIM GELİŞTİRME METODOLOJİSİ ÖNERİSİ

Tez Hazırlayanın Adı Soyadı : Gülizar Duygu KURT KAYA
Tez Danışmanın Adı Soyadı : Özge ERDEM
Birim Amirinin Adı Soyadı : Ömer ALAN

Gülizar Duygu KURT KAYA tarafından hazırlanan "Bilgi Güvenliği Ve Siber Güvenlik Kapsamında Bakanlık Uygulamaları için Güvenli Yazılım Geliştirme Metodolojisi Önerisi" adlı bu tezin Çevre ve Şehircilik Uzmanlık tezi olarak uygun olduğunu onaylarım.

Çevre ve Şehircilik Uzmanı,
Özge ERDEM
Tez Danışmanı

Bu çalışma, tez savunma komisyonumuz tarafından Çevre ve Şehircilik Uzmanlık tezi olarak kabul edilmiştir.

Başkan : Genel Müdür V. Ömer ALAN

Üye : Genel Müdür Yrd. V. Recep YILDIRIM

Üye : Daire Başkanı V., ,İskender ERMİŞ

Üye : Daire Başkanı V., ,Sibel ASLAN

Üye : Çevre ve Şehircilik Uzmanı, Özge ERDEM

Bu tez, Çevre ve Şehircilik Uzmanlığı Tez Hazırlama Yönergesi' ne uygundur.

İÇİNDEKİLER

ÖZET x
ABSTRACT xi
TEŞEKKÜR xii
TABLO LİSTESİ xiii
ŞEKİL LİSTESİ xiv
KISALTMALAR xv
GİRİŞ 1
1. BİLGİ GÜVENLİĞİ 4
1.1. Bilgi Güvenliği 4
1.2. Bilgi Güvenliği Unsurları 5
1.2.1. Gizlilik 6
1.2.2. Bütünlük 6
1.2.3. Erişilebilirlik 6
1.3. Sık Karşılaşılan Güvenlik Saldırıları 8
1.3.1. Gizliliği Tehdit Eden Saldırılar: 8
1.3.2. Bütünlüğü Tehdit Eden Saldırılar: 8
1.3.3. Erişilebilirliği Tehdit Eden Saldırılar: 8
2. SİBER GÜVENLİK 9
2.1. Genel Bakış 9
2.2. Siber Güvenlikle İlgili Temel Kavramlar 10
2.2.1. Siber Uzay 10
2.2.2. Siber Saldırı 11
2.2.3. Siber Silah 12
2.2.4. Siber Savaş 13
2.2.5. Siber Suç 13
2.2.6. Siber Güvenlik 13
2.2.7. Siber Savunma 13
2.3. Ülkemizde ve Dünyada Siber Güvenlik 14
2.3.1. Siber Güvenliğin Tarihçesi 16
2.3.2. Dünyada Siber Güvenlik 19
2.3.3. Türkiye'de Siber Güvenlik 21
3. GÜVENLİ UYGULAMA YAZILIMI GELİŞTİRME 27

3.1. En Sık Karşılaşılan Uygulama Güvenlik Riskleri: 30
3.1.1. A1- Enjeksiyon Açıkları: 32
3.1.2. A2- İhlal Edilmiş Kimlik Doğrulama ve Oturum Yönetimi 33
3.1.3. A3- Siteler Arası Betik Yazma (XSS) 34
3.1.4. A4- Emniyetsiz Doğrudan Nesne Referansı 35
3.1.5. A5- Yanlış Güvenlik Yapılandırması 36
3.1.6. A6- Hassas Bilgi Sızıntısı 36
3.1.7. A7- Eksik İşlev Seviyesi Erişim Kontrolü 37
3.1.8. A8- Siteler Ötesi İstek Sahteciliği (CSRF) 37
3.1.9. A9 - Bilinen Güvenlik Açığı Olan Bileşenleri Kullanma 38
3.1.10. A-10 Doğrulanmamış Yönlendirmeler 38
3.2. Uygulama Yazılımı Geliştirme Süreçlerinde Ele Alınması Gereken Temel Güvenlik Konuları: 39
3.2.1. Girdi Doğrulama 40
3.2.2. Kimlik Doğrulama 41
3.2.3. Kriptografi 43
3.2.4. Yetkilendirme 45
3.2.5. Oturum Yönetimi 45
3.2.6. Konfigürasyon Dosyaları Yönetimi 47
3.2.7. Hassas Bilgi 47
3.2.8. Parametre Manipülasyonu 47
3.2.9. Hata Yönetimi 48
3.2.10. Kayıt Tutma ve Denetim 48
3.3. Yazılım Geliştirme Süreç ve Modelleri 49
3.3.1. Yazılım Yaşam Döngüleri 50
3.3.2. Süreç İyileştirme ve Olgunluk Modelleri 54
3.4. Güvenli Yazılım Geliştirme Modelleri 60
3.4.1. Yazılım Güvencesi Olgunluk Modeli (SAMM) 61
3.4.2. Microsoft Güvenlik Geliştirme Yaşam Döngüsü (Microsoft SDL) 67
3.4.3. Yazılım Güvenliği Temas Noktaları (Touchpoints) 71
3.4.4. Sistem Güvenlik Mühendisliği Yetenek Olgunluk Modeli (SSE-CMM) 73
3.4.5. Ortak Kriterler (ISO 15408) 74
3.4.6. Güvenli Yazılım Geliştirme Modelleri Genel Değerlendirme 78

4. GÜVENLİ UYGULAMA YAZILIMI GELİŞTİRME ARAŞTIRMASI, BULGULAR VE YORUM 79
4.1. Yöntem 79
4.1.1. Araştırma Modeli 79
4.1.2. Araştırma- Çalışma Grubu 80
4.1.3. Veri Toplama Aracı 82
4.1.4. Verilerin Analizi 83
4.2. Bulgular ve Yorum 83
5. BAKANLIK İÇİN YOL HARİTASI VE ÖNERİLEN GÜVENLİ YAZILIM GELİŞTİRME MODELİ 92
5.1. Mevcut Durum Değerlendirilmesi 92
5.2. Bakanlık İçin Öneriler ve Yol Haritası 96
5.2.1. Çevre ve Şehircilik Bakanlığı için Önerilen Güvenli Yazılım Geliştirme Metodolojisi 98
SONUÇLAR VE TARTIŞMA 108
KAYNAKÇA 111
EK-1 : Önerilen Güvenli Yazılım Geliştirme Politikası 116
EK-2 : Önerilen Güvenli Yazılım Geliştirme Kontrol Listesi 117
ÖZGEÇMİŞ 118
ETİK KURALLARA UYGUNLUK BEYANI 119

ÖZET

ÇEVRE ve ŞEHİRCİLİK BAKANLIĞI	
Tezin Adı	Bilgi Güvenliği Ve Siber Güvenlik Kapsamında Bakanlık Uygulamaları için Güvenli Yazılım Geliştirme Metodolojisi Önerisi
Türü	Çevre ve Şehircilik Bakanlığı Uzmanlık Tezi
Yazar	Gülizar Duygu KURT KAYA
Teslim Tarihi	06.04.2017
Anahtar Kelimeler	Bilgi Güvenliği, Siber Güvenlik, Güvenli Yazılım Geliştirme, Metodoloji
Tez Danışmanı	Özge ERDEM
Sayfa Adedi	140

Günümüzde teknolojinin gelişmesiyle birlikte siber uzayda bulunan bilgi miktarı artmış ve saldırılara açık bir hale gelmiştir. Bu sebeple, bir varlık türü olarak bilginin izinsiz veya yetkisiz bir biçimde erişim, kullanım, değiştirilme, ifşa edilme, ortadan kaldırılma, el değiştirme ve hasar verilmesini önlemek için bilgi güvenliği ve siber güvenlik konuları büyük önem kazanmıştır. Siber Güvenliğin sağlanması için atılması gereken önemli adımlardan biri güvenli uygulama yazılımları geliştirebilmektir. Bu konu 2016-2019 Ulusal Siber Güvenlik Eylem Planında "Stratejik Siber Güvenlik Amaçları ve Eylemleri" başlığı altında 15.maddede "Güvenli yazılım geliştirme ve tedarik yönetimi kültürünün oluşturulması" hedef olarak yer almıştır.

Hazırlanan bu tez çalışmasında Bilgi Güvenliği ve Siber Güvenlik konuları açıklanmış, Güvenli Uygulama Yazılımı Geliştirme metodolojileri araştırılmıştır. Ayrıca "Güvenli Uygulama Yazılımı Geliştirme" konusunda yapılan anket çalışması ve saha ziyaretleriyle yazılım geliştiricilerin konu ile ilgili düşünceleri araştırılmıştır. 2016-2019 Ulusal Siber Güvenlik Eylem Planında belirtilen hedef kapsamında, Çevre ve Şehircilik Bakanlığının kurumsal altyapısı ve mevcut durumuna uygun Güvenli Uygulama Yazılımı Geliştirme Metodolojisi önerilmiş, Güvenli Yazılım Geliştirme Politikası ve Güvenli Yazılım Geliştirme Kontrol Listesi hazırlanmıştır.

ABSTRACT

MINISTRY OF ENVIRONMENT AND URBANIZATION	
Thesis	Methodology for Secure Software Development for Ministry Applications in the Scope of Information Security and Cyber Security
Type	Ministry of Environment and Urbanization Expertise Thesis
Author	Gülizar Duygu KURT KAYA
Submission Date	06.04.2017
Key Words	Information Security, Cyber Security, Secure Software Development, Metodology
Advisor	Özge ERDEM
Total Page	140

Today, with the development of technology, the amount of information found in cyberspace has increased and becomes open to attacks. For this reason, information security and cyber security issues have gained a great importance in order to prevent access, use, modification, disclosure, removal, alteration and damage of information in an unauthorized or unauthorized way. One of the most important steps of assuring cyber security is devopling secure application software. This topic has been taken as the target of "Secure Software Development and Supply Management Culture Creation" under the heading "Strategic Cyber Security Objectives and Actions" in the 2016-2019 National Cyber Security Action Plan.

In this thesis, information security and cyber security issues are explained and methods of Secure Application Software Development are researched. In addition, a "Secure Application Development" questionnaire was conducted and field visits were carried out in order to learn about software developers' thinking about the subject, Within the scope of the target specified in the 2016-2019 National Cyber Security Action Plan, the Secure Software Development Methodology recommended by taking into consideration the institutional infrastructure and the current situation of the Ministry of Environment and Urbanization. In addition, a Secure Software Development Policy and Secure Software Development Checklist are recommended for the Ministry.

TEŞEKKÜR

Tez çalışmam boyunca desteğini esirgemeyen başta eşime, Daire Başkanımız İskender ERMİŞ'e, tez danışmanım Özge ERDEM'e ve mesai arkadaşlarıma teşekkür ederim.

(İmza)

Gülizar Duygu KURT KAYA

TABLO LİSTESİ

Tablo 1: Siber Savunma Basamakları 14
Tablo 2: Genel siber güç sıralaması ilk 5 derecedeki ülkeler 21
Tablo 3: OWASP Uygulamalarda En Sık Karşılaşılan 10 Güvenlik Açığı Listesi ... 32
Tablo 4: SPICE Seviye Adları ve Nitelikleri 55
Tablo 5: CMMI Olgunluk Seviyeleri 56
Tablo 6: CMMI Yeterlilik Düzeyleri 57
Tablo 7: TMMI Olgunluk Seviyeleri 58
Tablo 8: Yazılım Yaşam Döngüsü Süreçleri 59
Tablo 9: Yazılım Yaşam Döngüsü Destek Süreçleri 59
Tablo 10: SAMM Olgunluk Seviyeleri 66
Tablo 11: Microsoft SDL Olgunluk Seviyeleri 68
Tablo 12: SSE-CMM Modelinin olgunluk seviyeleri 74
Tablo 13: Güvenli Yazılım Geliştirme Modelleri Genel Değerlendirme 78
Tablo 14: "Soru 11: Kurumunuzda dışarıdan edinilen yazılımlar için aranılan güvenlik isterleri nelerdir?" İçerik analiz tablosu 89
Tablo 15: "Soru 13: Görüş ve Önerileriniz" İçerik analiz tablosu 90
Tablo 16: ISO 27001 Bilgi Güvenliği Kontrolleri 99
Tablo 17: ISO 27002 Kontrolü 101

ŞEKİL LİSTESİ

Şekil 1: Bilgi Güvenliği Üçlüsü (CIA Triad) 5
Şekil 2: Parker Altılısı 7
Şekil 3: Günlük internet kullanımı örneği 9
Şekil 4: Siber Uzay Bileşenleri 11
Şekil 5: Siber Saldırı Haritası 15
Şekil 6: Türkiye Siber Saldırı Haritası 15
Şekil 7: Türkiye'nin Siber Güç Sıralamasındaki Yeri 25
Şekil 8: Türkiye'nin Dünya ülkeleri ile karşılaştırılması 26
Şekil 9: Bilgi Güvenliği, Siber Güvenlik ve Uygulama Güvenliği İlişkisi 28
Şekil 10: Güvenlik Tehditleri: En Çok Endişe Edilen Konular 28
Şekil 11: Yazılım geliştirme aşamalarına göre yazılım açıklarını giderme maliyeti (Zaman ve masraf) 30
Şekil 12: OWASP -Tehdit unsurları ile Kurumsal etkiler arasındaki ilişki 31
Şekil 13: Çerez işleyiş örneği 46
Şekil 14: Kayıt Altına alınması gereken bilgiler 49
Şekil 15:Şelale (Waterfall) Yazılım Geliştirme Modeli 50
Şekil 16: Çevik Yazılım Geliştirme Modeli 51
Şekil 17: Spiral Yazılım Geliştirme Modeli 53
Şekil 18: V Model 54
Şekil 19: Yazılım geliştirme aşamalarında ve güvenlik işlemleri 61
Şekil 20: SAMM İş Fonksiyonları ve Güvenlik Eylemleri 62
Şekil 21: Güvenlik için Genel Yaklaşım 68
Şekil 22: Temas Noktaları Ayakları 71
Şekil 24: SWOT Analizi 93
Şekil 25: Oluşturulması hedeflenen güvenli yazılım geliştirme metodolojisi 98
Şekil 26: SAMM Genel Yapısı 102
Şekil 27 : Değerlendirme çizelgeleri için akış diyagramı 106
Şekil 28: SAMM işletimi akış diyagramı 107

KISALTMALAR

BKK	:	Bakanlar Kurulu Kararı
BOME	:	Bilgisayar Olayları Müdahale Ekibi
CNNS	:	Ulusal Güvenlik Sistemleri Komitesi (Committee on National Security Systems)
DLP	:	Veri Kaybı/Sızıntısı Önleme (Data Loss Prevention)
DOS	:	Hizmet Dışı Bırakma (Denial Of Service) Saldırısı
DDOS	:	Dağıtık Hizmet Dışı Bırakma (Distributed Denial Of Service) Saldırısı
ICS-CERT	:	Endüstriyel Kontrol Sistemleri Siber Acil Müdahale Ekibi (The Industrial Control Systems Cyber Emergency Response Team)
$(ISC)^2$	:	Uluslararası Bilgi Sistemi Güvenlik ve Sertifikasyon Konsorsiyumu
ISO	:	Uluslararası Standardizasyon Organizasyonu (International Organization for Standardization)
ISSA	:	Uluslararası Sistem Güvenliği Mühendisliği Birliği (The International Systems Security Engineering Association)
ITU	:	Uluslararası Telekomünikasyon Birliği (International Telecommunication Union)
NICCS	:	Siber Güvenlilik Eğitimi Ulusal Girişimi (National Initiative for Cybersecurity Careers and Studies)
NSA		ABD Ulusal güvenlik Dairesi (The National Security Agency)
OWASP	:	Açık Web Uygulaması Güvenlik Projesi (The Open Web Application Security Project)
SAMM	:	Yazılım Güvencesi Olgunluk Modeli (Software Assurance Maturiy Model)
SOAP	:	Basit Nesne Erişim Protokolü (Simple Object Access Protocol)
SPICE	:	Yazılım Süreç Yeterliliği ve Organizasyonel Olgunluk Standardı (Software Process Improvement and Capability dEtermination)
SSE-CMM	:	Sistem Güvenlik Mühendisliği Yetenek Olgunluk Modeli (Systems Security Engineering Capability Maturity Model)
TÜBİTAK	:	Türkiye Bilimsel Ve Teknolojik Araştırma Kurumu
UDHB	:	Ulaştırma, Denizcilik ve Haberleşme Bakanlığı
UEKAE	:	Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü

GİRİŞ

Bilgi; verinin belli bir anlam ifade edecek şekilde düzenlenmiş halidir(Eminağaoğlu ve Gökşen 2009). Bir başka deyişle, bir sorunun cevabıdır. Bilgi; araştırma, deney, gözlem veya iletişim yoluyla elde edilebilir. Belirsizliği gidererek davranış, karar veya sonuçları etkileyebilir. Bu sebeple büyük önem taşır.

İnsanlar sahip oldukları bilgiyi korumaları gerektiğini çok uzun zaman önce fark ettiler. Bilgiyi korumak adına gösterilen çabaların ilk örnekleri milattan öncesine dayanmaktadır. Bilgi güvenliğinin ilk zamanlarında savunma yöntemi olarak çoğunlukla kriptolojiye başvurulmuştur; Rosetta Tableti, Sezar Şifreleme, Ebced hesabı gibi. 1950'li yıllarda bilgisayarların geliştirilmesiyle birlikte 1960'lı yıllarda internet kavramı ortaya çıkmış ve bilgi güvenliği konusu yeni bir boyut kazanmıştır. İnternetin gelişmesiyle bilgiye karşı oluşan tehditler siber uzaya taşınmıştır. 1995'te Dünyanın ilk dijital suçlusu olarak bilinen Kevin Mitnick bilgisayar korsanlığı suçundan hapse giren ilk kişi olmuştur.

İnternetin ortaya çıkışından bu yana, kullanıcı profili ve sayısı çok büyük ölçüde değişmiştir. İlk zamanlarda internet erişimi olan kişiler akademik veya askeri personel iken, günümüzde ilkokul öğrencilerinden başlayarak her yaştan ve her meslek grubundan kişiler internet kullanıcısı olmuştur. Ayrıca internet kullanıcılarının sayısı her geçen gün artmaktadır. 1995'te Dünya nüfusunun yaklaşık olarak %1'i internet erişimine sahipken günümüzde bu oran %40'a ulaşmıştır (Number of Internet Users (2016) - Internet Live Stats).

Yaşadığımız iletişim çağında, internet hayatımızın her alanına girmiştir; kişisel mesajlaşma, alışveriş ve bankacılık gibi birçok alanda hemen her dakika işlem yapılmaktadır. İnternetin sağladığı küresel bağlantı dünyasında virüsler, bilgisayar korsanları, elektronik dolandırıcılık, elektronik dinleme gibi birçok tehdit de mevcuttur. Bu tehditlerin fazlalığı bilgi güvenliği konusunun öneminin artmasına sebep olmuştur. Bilgi Güvenliği; bilgilerin izinsiz kullanımından, ifşa edilmesinden, yok edilmesinden, değiştirilmesinden ve bilgilere hasar verilmesinden koruma veya bilgilere yapılacak olan izinsiz erişimleri engelleme işlemidir (Eminağaoğlu ve Gökşen 2009). Siber Güvenlik

ise Uluslararası Standartlar Örgütünün belirlediği tanıma göre siber uzayda (bir iletişim ağı üzerinden bağlantılı bulunan tüm kullanıcılar, ağlar, bilgisayarlar, vb.) gizlilik, bütünlük ve erişilebilirliğin korunması işlemleridir.

Bilişim güvenliği, dijital ortamda depolanan bilgilerin üçüncü şahıslar tarafından ele geçirilmesini önlemek, bilgi transferi sırasında bilginin bütünlüğünün ve yapısının bozulmadan aktarılmasını sağlamak, sistemlere yetkisiz kişilerin erişmesini engellemek, sistemin sürekli olarak erişilebilir olmasını sağlamak için verilmesi gereken uğraşların tümüdür. Gartner Inc. tarafından yapılan bir araştırmaya göre bilişim güvenliği ihlallerinin %80'i yazılım güvenliği sorunlarından kaynaklanmaktadır (Özbilgin, Gökhan Özlü 2010). Bu durum bilgiye yönelik olarak tehditler oluşmasına imkân vermektedir. Yazılım güvenliği sorunlarını aşmak için son yıllarda çalışmalar hızlanmış ve "Güvenli Yazılım Geliştirme" konusu siber güvenlik alanında önemli bir konu haline gelmiştir. "Güvenli Yazılım Geliştirme için atılacak birinci adım endüstri alanında kabul görmüş bir yazılım geliştirme metodolojisini benimsemektir. Bu metodolojiler dünya çapında kabul görmüş standart hazırlama kuruluşları tarafından kullanıma sunulmuştur. Bahsedilen metodolojilerin benimsenmesiyle birlikte güvenli yazılım geliştirme ilkeleri de uygulanmalıdır.

Bu uzmanlık tezi kapsamında güvenli uygulama yazılımı geliştirme ilkeleri ve standartları üzerinde durulacak, zafiyetleri gidermek için izlenmesi gereken yollar hakkında bilgi verilecektir. Bu bağlamda konunun daha iyi anlaşılabilmesi için bilgi güvenliği ve siber güvenlik konuları hakkında bilgilendirme yapılacaktır. Son olarak Çevre ve Şehircilik Bakanlığı bünyesinde geliştirilen ve temin edilen uygulama yazılımları için güvenli uygulama yazılımı geliştirmeye dair değerlendirmeler ve çözüm önerileri yapılacaktır. Bu tez çalışması altı bölümden oluşmakta olup aşağıda detayları verilmiştir.

Birinci bölümde bilgi güvenliği konusu açıklanmış, bilgi güvenliği unsurları ve sık karşılaşılan güvenlik saldırılarından bahsedilmiştir.

İkinci bölümde siber güvenlik hakkında detaylı bilgi verilmiş ve bilgi güvenliği ile olan ilişkisinden bahsedilmiştir. Bu bölümde genel olarak siber

güvenliğin ne olduğu, güncel siber güvenlik konuları ve yaşanmış örnekler hakkında bilgiler verilmiştir.

Üçüncü bölümde güvenli uygulama yazılımı geliştirme konusu incelenmiştir. Bu kapsamda sık karşılaşılan güvenlik saldırılarına ve alınabilecek önlemlere değinilmiştir. Ayrıca temel güvenlik kurallarından bahsedilmiştir. Daha sonra güvenli uygulama yazılımı geliştirme metodolojilerinin ilk basamağı olan yazılım yaşam döngüleri ve yazılım olgunluk modelleri kısaca açıklanmıştır. Daha sonra güvenli yazılım geliştirme modelleri araştırılmıştır.

Dördüncü bölümde güvenli yazılım geliştirme konusunda yazılım geliştiriciler arasında yapılan araştırma, elde edilen bulgular ve yorumlara yer verilmiştir.

Beşinci bölümde Çevre ve Şehircilik Bakanlığına ait uygulama yazılımlarının geliştirme sürecine dair mevcut durum incelenmiştir. Elde edilen bulgular göz önünde bulundurularak Bakanlığımız için güvenli uygulama yazılımı geliştirme metodolojisi önerisinde bulunulmuştur. Ayrıca kullanılması önerilen “Güvenli Uygulama Yazılımı Geliştirme Politikası” ve “Güvenli Uygulama Yazılımı Geliştirme Kontrol Listesi” ne yer verilmiştir.

1. BİLGİ GÜVENLİĞİ

1.1. Bilgi Güvenliği

Bilgi, insan hayatındaki en hassas ve önemli varlıklardan birisidir. Kişiler, kurumlar ve ülkeler için bilgi, elde edilmesi zor, aynı zamanda elde tutulması da zor bir metadır(Canbek ve Sağıroğlu 2006). Kanadalı ekonomist ve diplomat olan John Kenneth Galbraith bir konuşmasında şunları söylemiştir:

"Endüstri toplumuna hız kazandıran şey paradır; fakat bilgi toplumunu hızlandıran ve güce ulaştıran bilgidir, şimdi bilgi sahibi olanlar ve olmayanlar şeklinde yeni bir sınıfsal bölünme ortaya çıktı. Bu yeni sınıf, gücünü paradan ya da sahip olunan topraklardan değil, sadece bilgiden alıyor."

Ünlü ekonomist bu sözleriyle çağımız hakkında haklı bir tespitte bulunmuş ve bilginin neden korunması gerektiğine dair ipucu vermiştir.

Çağımızın getirdiği bilgiye olan bağımlılık, bilginin ve bilgi güvenliği konusunun önemini arttırmıştır. Teknolojinin gelişmesiyle birlikte elektronik olarak saklanan bilgi sayısını ciddi boyutlara ulaştırmıştır. Bu durum beraberinde yeni riskleri ortaya çıkarmıştır. Ortaya çıkan bu riskler ise bilgi güvenliğini sağlamak adına yeni tedbirlerin alınması ihtiyacını doğurmuştur.

Günlük hayatta ve iş hayatında sürekli olarak bilgi güvenliğine ihtiyaç duyulmaktadır. Örneğin; alıcılar satıcıların kimliklerini doğrular, kurumlar çalışanların kimlik bilgilerini doğrular ve kredi veren kurumlar güvenli işlemleri kullanarak herkesin kimliğini doğrular. Bilgi güvenliği, bu bilgilerin ve diğer bilgilerin yetkisiz kişiler tarafından korunması için gereklidir. Bilgiye karşı yapılan saldırılar bilginin zarar görmesine (silinmesi, değiştirilmesi, bütünlüğünün bozulması, vb.) sebep olabilir ve hem itibar kaybına hem de yürütülen işlerde sıkıntılar yaşanmasına sebep olabilir. Bu sebeple bilgi güvenliği konusu kurumların itibarı için önem arz ettiği gibi maddi ve hukuksal açılardan da önemlidir.

Kurumların ve çalışanlarının ellerindeki bilgileri koruması büyük önem arz etmektedir. Bu önem, kurum ve işin sürekliliği, başarısı, kamu, toplum, ticari ve bağımsız organizasyonlara karşı yerine getirilmesi gereken sorumluluklardan kaynaklanmaktadır. Bilgi Güvenliği, kurumların itibarının, güvenilirliğinin, bilgi varlıklarının korunması, faaliyetlerinin en az kesinti ile

devam etmesini hedefler ve başta elektronik olmak üzere, çeşitli ortamlardaki kritik bilgilerinin ve diğer bilgi varlıklarının korunmasını sağlar.

Bilgi Güvenliği Uluslararası Standardizasyon Organizasyonu Standartlarında (ISO / IEC 27000: 2009) verilen tanıma göre, bilgilerin Gizlilik, Bütünlük ve Erişilebilirlik özelliklerini korunmasıdır. Ek olarak, özgünlük, hesap verebilirlik, reddedilemezlik ve güvenilirlik gibi diğer özellikler de söz konusudur. Bu konuda uzman bir kuruluş olan Ulusal Güvenlik Sistemleri Komitesi (CNNS) ise bilgi güvenliğini, gizlilik, bütünlük ve erişilebilirlik sağlamak için bilgi ve bilgi sistemlerinin yetkisiz erişime, kullanıma, açıklanmaya, bozulmaya, değiştirilmeye veya tahrip edilmesine karşı korunması olarak tanımlamıştır. Bilgi Güvenliğinin bir başka tanımı da 2009 tarihli E-Devlet ve Bilgi Toplumu Kanun Tasarısı Taslağında yapılmış ve mevzuatta yer almıştır.

1.2. Bilgi Güvenliği Unsurları

Bilgi güvenliği, bilgilerin elektronik veya fiziksel olarak saklanıp saklanmadığıyla ilgilenmeden, bilgiyi yetkisiz erişime, kullanıma, bozulmaya, değiştirmeye veya yok etmeye karşı korur. Bilgi Güvenliğinin sağlanması için üç temel unsur göz önünde bulundurulmalıdır. Bu unsurlar;

- Gizlilik (Confidentiality)
- Bütünlük (Integrity)
- Erişilebilirlik (Availability)

Bu üç kavram, bilgi güvenliği için hedeflenen unsurları somutlaştırmaktadır.

Şekil 1: Bilgi Güvenliği Üçlüsü (CIA Triad)

(Singh, Vaish, ve Keserwani 2014)

1.2.1. Gizlilik

Bilginin yetkisiz kişilerin erişimine açık olmaması anlamına gelmektedir. Gizlilik konusu veri gizliliği ve mahremiyet olarak ikiye ayırılabilir. Veri gizliliği, özel veya gizli bilginin yetkisiz kişilere açıklanmadığını veya ifşa edilmediğini garanti eder. Mahremiyet, bireylerin kendileri ile ilgili hangi bilgilerin toplanıp saklanabileceğini ve kimlerin kime ve kim tarafından ifşa edilebileceğini kontrol etmesini ya da etkilemesini sağlar.

1.2.2. Bütünlük

Bilginin yetkisiz kişilerce değiştirilememesi, silinememesi veya herhangi bir şekilde zarar görmesine sebep olacak saldırılardan korunuyor olması anlamına gelir. Ayrıca bütünlük bilginin kazara ya da kasıtlı olarak bozulmaması olarak tanımlanabilir.

1.2.3. Erişilebilirlik

Bilginin her ihtiyaç duyulduğu anda erişime açık olması durumuna denir. Kullanıcının yetkileri dâhilinde herhangi bir sorun ya da problem çıkması durumunda bile bilgiye erişebiliyor olması gerekmektedir.

Bilgi Güvenliği konusunda yapılan çalışmaların ilk yıllarında "Bilgi Güvenliği Üçlüsü" temel unsurlar olarak yeterli görülse de zaman içerisinde yetersiz kaldığı düşünülmüştür (Stallings 2011). Bu modelin daha çok teknoloji odaklı olduğunun ve bilgi güvenliğinin insan unsuru üzerinde yeterince odaklanmadığının üstünde durulmuştur (Pender-Bey 2012). Oysaki insan faktörü bilgi güvenliği açısından en büyük tehdidi oluşturmaktadır. Donn B. Parker 1988 yılında Parker Altısı olarak adlandırılan yeni bir set ortaya atmıştır. Parker Altılısı Bilgi güvenliği üçlüsünün üzerine kurulmuş olmasına rağmen bilgi güvenliği konusunu daha kapsamlı bir şekilde ele almaktadır. Parker Altılısı tehditlere karşı koruma sağlamak için yoğunluklu olarak kimlik doğrulama ve şifrelemeye dayanır. Parker Altılısı Şekil-2 de gösterildiği gibidir.

Şekil 2: Parker Altılısı

(Pender-Bey 2012)

Parker Altılısında verilen ek unsurlar:

- Özgünlük (Authenticity): Hakiki olma, doğrulanabilir ve güvenilir olma özelliği; İletimin, mesajın veya mesaj göndereninin geçerliliğine güvenin sağlanabilmesi için gereklidir. Bu, kullanıcıların bulunduklarını söylediklerini ve sisteme gelen her girişin güvenilir bir kaynaktan geldiğini doğrulamak demektir (Pender-Bey 2012).
- Yararlılık (Utility): Maksada uygunluğu ifade eder. Örneğin, yetkisiz girişi veya tespit edilmemiş değişiklikleri önlemek için birisinin disk üzerinde veri şifrelediğini, daha sonra şifre çözme anahtarını kaybettiğini varsayalım. Veriler gizli, kontrollü, özgün ve kullanılabilir olacaktır ancak bu biçimde kullanışlı olmayacaklardır. Bir başka örnek olarak, belirli bir bilgisayar mimarisi için uygun olmayan bir biçimde verilerin depolanması verilebilir (https://en.wikipedia.org/wiki/Parkerian_Hexad).
- Sahiplik /Kontrol (Possession/Control): Bilgi ile fiziksel temasın engellenmesi, kopyalanmasını veya yetkisiz kullanımını engelleme işidir(Kabay,M.E. Whyne,Eric 2009).

Detayları verilen unsurların yanında bilgi güvenliği konusunda öne çıkan bir başka temel özellikte "Hesap verebilirlik" tir (Singh, Vaish, ve Keserwani 2014).

- Hesap verebilirlik(Accountability): Bir varlığın eylemlerinin, o varlık için benzersiz şekilde izlenmesini gerektiren güvenlik hedefidir. Bu hedef, itiraz reddedilme, caydırıcılık, hata izolasyonu, saldırı tespit ve önleme ve eylem sonrası iyileşme ve hukuki eylemi destekler. Sistemler güvenlik ihlallerinin izini sürmek veya işlem anlaşmazlıklarına son verebilmek için daha sonraki adli analizlere izin vermek için faaliyetlerinin kayıtlarını tutmalıdır (Stallings 2011).

1.3. Sık Karşılaşılan Güvenlik Saldırıları

Bilgi güvenliğini tehdit eden saldırılar üç ana başlıkta gruplandırılabilir.

1.3.1. Gizliliği Tehdit Eden Saldırılar:

Genel olarak, iki tür saldırı, bilginin gizliliğini tehdit eder: gözetleme ve trafik analizi. Ağ trafiğini dinleme (Snooping), verilere yetkisiz erişime veya veri kesme işlemine yönelik olarak gerçekleştirilir. Trafik analizi ise bir saldırganın çevrimiçi trafiği izlenerek toplanan değişik türdeki bilgileri elde etme işlemidir.

1.3.2. Bütünlüğü Tehdit Eden Saldırılar:

Verilerin bütünlüğü çeşitli saldırılarla tehdit edilebilir. Bu saldırılara örnek olarak değiştirme, maskeleme, tekrarlama ve reddetme verilebilir.

1.3.3. Erişilebilirliği Tehdit Eden Saldırılar:

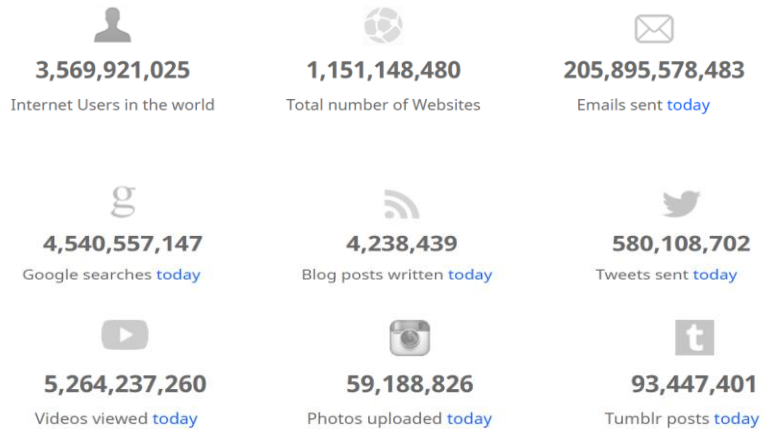
Hizmet dışı bırakma (DOS) saldırıları ve dağıtık hizmet dışı bırakma saldırıları(DDOS), bir sistemin hizmetini yavaşlatabilir veya tamamen kesebilir. Saldırgan bunu başarmak için çeşitli stratejiler kullanabilir. Sistemi çok meşgul edebilir, çökertir ya da bir yönde gönderilen iletileri kesebilir ve gönderme sistemini iletişim veya mesajdaki taraflardan birinin mesajı kaybettiğine ve tekrar gönderilmesi gerektiğine inanmasına neden olabilirler.

2. SİBER GÜVENLİK

2.1. Genel Bakış

İnternette ve Siber uzayda güvenlik giderek artan bir endişe konusu olmuştur. Kullanıcılar, web siteleri aracılığıyla siber dünyada varlıklarını kuruyor ve geçen her saniye siber uzay tarafından sağlanan sanal dünyayı daha fazla kullanmaktadırlar. Şekil 3'te "Internet Live Stats" sitesinin, internet ile ilgili istatistiklerinden örnek verilmiştir. Bu istatistiklerden de anlaşılabileceği gibi her an iş hayatı, sağlık, bankacılık, alışveriş, eğlence ve sosyal medya gibi alanlarda milyonlarca kişi internet üzerinden bilgi aktarımı yapmaktadır.

Şekil 3: Günlük internet kullanımı örneği



("Internet Live Stats - Internet Usage & Social Media Statistics,")

Siber güvenlik konusunda çalışmalar yapan ve uluslararası bir yazılım şirketi olan Symantec'in verilerine göre 2015'te yarım milyar kişisel veri çalınmış veya kaybolmuştur. Ayrıca 2015 yılında her gün bir milyondan fazla web saldırısı gerçekleştirildi. Web sitesi yöneticileri web sitelerini güvence altına alamadığı için, siber suçlular, kullanıcıları etkilemek için meşru web sitelerindeki güvenlik açıklarından faydalanmaya devam etmektedirler. Tüm meşru web sitelerinin yaklaşık yüzde 75'i yamalı güvenlik açıklarına sahip ve kullanıcıları risk altına sokmaktadır ("Internet Security Threat Report 2016 | Symantec,").

Siber uzayın güvenliğini sağlamak ve korumak için yapılan çalışmaların tümüne "Siber Güvenlik" adı verilir. 2013 yılında yayınlanan Resmi Gazetede "Bilişim Güvenliği", dijital ortamda depolanan bilgilerin üçüncü şahıslar tarafından ele geçirilmesini önlemek, bilgi transferi sırasında bilginin bütünlüğünün ve yapısının bozulmadan aktarılmasını sağlamak, sistemlere yetkisiz kişilerin erişmesini engellemek, sistemin sürekli olarak erişilebilir olmasını sağlamak için verilmesi gereken uğraşların tümü olarak tanımlanmıştır.

Uluslararası Telekomünikasyon Birliği (ITU) tarafından hazırlanan "ITU-T X.1205: Overview of cybersecurity" standardında belirtildiği şekliyle siber güvenlik, siber çevreyi ve kuruluşun ve kullanıcının varlıklarını korumak için kullanılabilecek araçlar, politikalar, güvenlik kavramları, güvenlik önlemleri, yönergeler, risk yönetimi yaklaşımları, eylemler, eğitim, en iyi uygulamalar, güvence ve teknolojilerin toplanmasıdır. Organizasyon ve kullanıcı varlıkları arasında, bağlı bilgisayar cihazları, personel, altyapı, uygulamalar, hizmetler, telekomünikasyon sistemleri ve siber ortamda iletilen ve / veya saklanan bilgilerin tamamı bulunur. Genel güvenlik hedefleri aşağıdakileri içermektedir:

- Mevcudiyet
- Doğruluk ve reddedilme olmak üzere bütünlük
- Gizlilik

2.2. Siber Güvenlikle İlgili Temel Kavramlar

Siber güvenliği daha iyi anlayabilmek için siber güvenlikle ilgili temel kavramların anlaşılması gerekmektedir. Bu kavramlar arasında siber uzay, siber saldırı, siber silah, siber savaş, siber suç ve siber güvenlik bulunmaktadır.

2.2.1. Siber Uzay

İnternet veya başka bir iletişim ağı üzerinden erişilebilir tüm bilgi yapılarını içerir. Bu, doğrudan veya dolaylı olarak ağlara bağlanabilen kullanıcılar, ağlar, cihazlar, tüm yazılımlar, süreçler, depolama ya da transit bilgileri, uygulamaları, hizmetleri ve sistemleri kapsar (ITU 2008). Uluslararası bir standart olan ISO 27032'de siber uzay; fiziksel biçimde var olmayan, teknoloji cihazları ve ağları vasıtasıyla internetteki insanların, yazılımların ve hizmetlerin etkileşiminden kaynaklanan karmaşık bir ortam

olarak tanımlanmıştır. Siber uzayın bileşenleri arasında kamu sektörü, özel sektörler ve bireyler bulunmaktadır.

Politik açıdan siber uzay, "beşinci alan" (arazi, deniz, hava ve uzay sonra) olarak bilinir ve ülkeleri yakından ilgilendiren bir konudur (Hamilton 2014). Günümüzde siber uzay yeni bir savaş alanı olarak kabul edilmektedir.

Şekil 4: Siber Uzay Bileşenleri

2.2.2. Siber Saldırı

Hedef seçilen kurum veya kişilerin bilgi sistemlerinin işleyişinin engellenmesi, bozulması, değiştirilmesi yoluyla; iş, idare veya toplumsal hayat üzerinde olumsuz etki oluşturulmasıdır. Ulusal veya uluslararası düzeyde ticari, politik veya askeri amaçlı olarak planlı ve koordineli bir faaliyet olarak gerçekleştirilebilir.

Bir veri iletişim sistemine yönelik tehditler şunları içerir:

- Bilgi ve / veya diğer kaynakların tahrip edilmesi
- Bozulma veya bilginin değiştirilmesi
- Bilgi ve / veya diğer kaynakların çalınması, kaldırılması veya kaybolması
- Bilgilerin ifşası

Siber saldırı türleri örnekleri aşağıda listelenmiştir;

- Açık mikrofon dinleme
- Yemleme (phishing)
- Kabloya saplama yapma
- Oturum çalma
- Bilgi kirliliği

- IP aldatmacası
- Ağ tarama
- Yığın e-posta gönderme (spam)
- İnternet servis saldırıları
- Kriptografik saldırılar
- Trafik analizi
- Zararlı yazılım
- Sosyal mühendislik (social engineering)
- Yerine geçme
- Zamanlama saldırıları
- Tuzak kapı (trapdoor)
- Hizmet dışı bırakma

2.2.3. Siber Silah

Siber saldırıları gerçekleştirmek için kullanılan siber ortam araçlarına siber silah adı verilir. Siber silahlar savaşlarda kullanılan güdümlü füzeler gibi üç ana unsurdan oluşmaktadır. İlki bir iletim aracı (roket motoru), ikincisi navigasyon sistemi, üçüncü olarak da roketin taşıdığı patlayıcı yük. Siber silahın hedefe ulaşmasını sağlayan web siteleri, USB diskler, mailler gibi iletim araçları. Hedefteki sistemde ilgili zafiyetin bulunması navigasyon ve zafiyeti bulunan sistemin istismar edilerek zararlı yazılımın çalışması gerçek silahlardaki patlayıcı kısmına denk gelmektedir (Bakır 2012). Siber silah örnekleri aşağıda listelenmiştir;

- Klavye İzleme (Key Logger)
- Bilgisayar Virüsleri
- İstemsiz ticari tanıtım (Adware)
- Kurtçuk (Worm)
- Casus yazılımlar (Spyware)
- Truva atı (Trojan)
- Yemleme(Phishing)
- Ağ trafiğinin dinlenmesi (Sniffing/Monitoring)
- İstemsiz elektronik posta (Spam)

2.2.4. Siber Savaş

Bir devletin, başka bir devletin bilgisayar sistemlerine veya ağlarına hasar vermek ya da kesinti yaratmak üzere gerçekleştirilen sızma faaliyetleridir.

2.2.5. Siber Suç

Siber uzayda hizmetlerin/ uygulamaların bir suç için kullanıldığı veya siber uzayın bir suçun kaynağı, aracı, hedefi olduğu suç faaliyetidir.

2.2.6. Siber Güvenlik

Siber Güvenlik, gerçekleştirilen siber saldırılardan kaynaklanabilecek zararlara karşı alınan önlemlerin tümüne verilen isimdir. ISO 27032' de Genel olarak güvenlik, bazı etkenlerin tanımlanan koşullar altında olumsuz etkilere neden olmayacağından emin olma durumu olarak tanımlanmaktadır. Siber güvenlik; kurumların ve kullanıcıların varlıklarını korumak için kullanılabilecek araçları, politikaları, güvenlik kavramlarını, güvenlik önlemlerini, yönergeleri, risk yönetimi yaklaşımlarını, eylemleri, eğitimleri, güvenceleri ve teknolojilerin tamamını kapsar.

2.2.7. Siber Savunma

Devletlerin siber saldırıları engelleme çabalarına Siber Savunma adı verilir. Siber tehditler bilgi varlıklara karşıdır, bu nedenle siber savunmanın ilk adımı korunması gereken varlıkları listelemektir. Bu yüzden devletler siber savunma amacıyla kritik altyapılarını belirlerler. Kritik altyapılar; zarar görmesi veya yok olması durumunda toplumsal düzenin ve kamu hizmetlerinin devamının sağlanmasında güçlük yaratacak; işlevleri kısmen veya tamamen yerine getiremediğinde vatandaşların sağlığına, emniyetine, güvenliğine ve ekonomik refahına olumsuz etki edecek yapılardır. Kritik altyapılar olarak;

- Bilgi ve iletişim
- Enerji
- Finans
- Sağlık
- Gıda ve Su
- Ulaşım
- Savunma

- Kamu güvenliği
- Nükleer, biyolojik, kimyasal ve radyoaktif maddeler sayılabilir (Alkan 2013).

Sonraki basamak bir tehdit analizi ve bir güvenlik açığı analizi (etki değerlendirmesi dâhil) yapmaktır. Son aşamada gerekli önlemler alınmalı ve güvenlik mekanizmaları oluşturulmalıdır. Siber savunma basamakları Tablo 1'de verilmiştir.

Tablo 1: Siber Savunma Basamakları

Basamak	Açıklama
1	Sistemin zayıf noktalarını belirlemek;
2	Belirlenen güvenlik açıklarına yönelik tehdit olasılığını analiz etmek;
3	Her bir tehdidin başarılı bir şekilde uygulanması halinde gerçekleşecek sonuçların değerlendirilmesi;
4	Her bir saldırının maliyetini tahmin etmek;
5	Potansiyel karşı önlemlerin maliyetinin hesaplanması;
6	Güvenlik mekanizmalarının seçilmesi (maliyet fayda analizi kullanılarak).

(Alkan 2013)

2.3. Ülkemizde ve Dünyada Siber Güvenlik

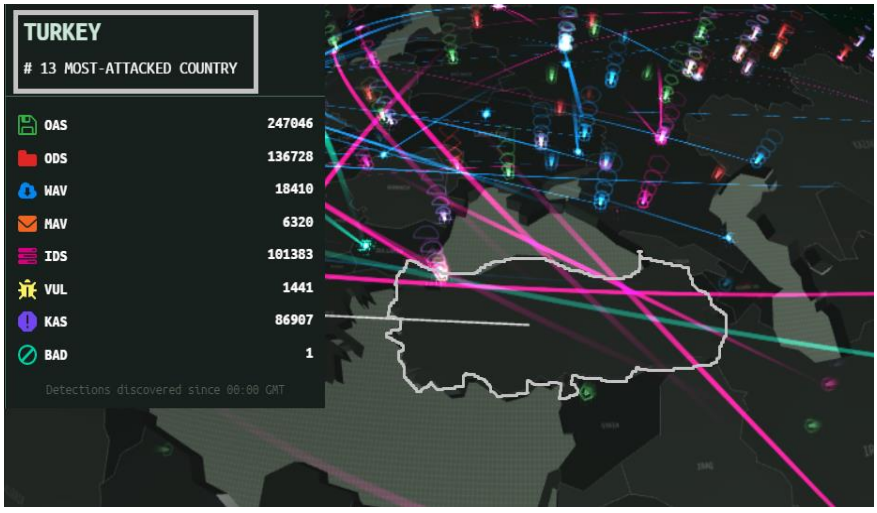
Kullanıcıları internette gündelik işlerini gerçekleştirirken genellikle güvende olduklarını düşünürler ancak bu büyük bir yanılgıdır. Kaspersky Lab tarafından yapılan bir açıklamada, her gün Kaspersky Lab antivirüs uygulamalarının üç yüz binden fazla kötü amaçlı nesne ile karşılaştığı belirtilmiştir. Ayrıca dünyada her gün milyonlarca siber saldırı gerçekleşmektedir ve bu saldırılar dünya ekonomisine yaklaşık olarak 400 milyar dolara mal olmaktadır. Dünya çapında gerçekleşen saldırıların örnekleri siber saldırı haritaları olarak adlandırılmaktadır. Siber saldırı haritalarının örnekleri Şekil 5' te gösterilmiştir.

Şekil 5: Siber Saldırı Haritası

(“Kaspersky Cyberthreat real-time map,”)

Yukarıda örnekleri verilmiş olan siber saldırı haritaları kullanarak hangi ülkelerin hedeflendiğini ve hangilerinin saldırıya başladığını gerçek zamanlı olarak görmek mümkündür. Örneğin erişim anında haritadan ileri geri gidip gelen birçok saldırıyla ABD ve Çin arasında görünüşte devam eden bir mücadele olduğu anlaşılabilmektedir. Gerçek zamanlı olarak tespit edilen farklı tehdit türleri farklı renklerle işaretlenmiştir. Anlık olarak bakıldığında erişim tarihinde Türkiye, siber saldırı alan ülkeler sıralamasında 13. Sırada görüntülenmiştir.

Şekil 6: Türkiye Siber Saldırı Haritası



(“Kaspersky Cyberthreat real-time map”)

2.3.1. Siber Güvenliğin Tarihçesi

Günümüzde “Siber Saldırı”, ”Siber Savaş” ve “Siber Güvenlik” gibi terimlerle sıkça karşılaşmaktayız. Başlangıcı Soğuk Savaş dönemine dayanan siber saldırılar, teknolojinin ilerlemesiyle birlikte artış göstermiştir. İnternetin yaygınlaşmasıyla birlikte siber saldırı gerçekleştirmek çok daha kolay bir hale gelmiş, herkes tarafından uygulanabilir olmuştur. Bu durum siber güvenlik ve siber savunma kavramlarının da önemini arttırmıştır.

Tarih boyunca karşılaşılan etkili siber saldırılar ve sebep oldukları zararların örnekleri aşağıda listelenmiştir.

1988

Dünyanın ilk gelişmekte olan siber altyapısını etkileyen ilk solucanlardan biri olan Morris solucanı, ABD'de yayılmış ve bilgisayarları kullanılamaz hale gelene kadar yavaşlatmıştır. Solucan, İnternet'in ne kadar büyük olduğunu ölçmeye çalışan Robert Tapan Morris tarafından geliştirilmiştir. Morris ABD'de bilgisayar dolandırıcılığı ve istismar eylemiyle mahkûm edilen ilk kişi olmuştur. Şu anda Massachusetts Teknoloji Enstitüsünde çalışmaktadır.

- *Bu olay günümüzde karşılaşılan dağıtık hizmet reddi (DDoS) saldırılarının temelini oluşturmuştur.*

1999

Melissa virüsü 80 milyon dolar zarara mal olan çok basit bir virüstür. Virüs dünya çapında birçok e-posta sunucusunu kilitlemiştir. Virüsün amacı, Microsoft Word belgelerine bulaşarak kendisini otomatik olarak bir elektronik posta eki olarak yaymaktır. Anti-virüs yazılımı satışları o yıl çok ciddi boyutlara ulaşmıştır.

2006

NASA, siber saldırıya uğrayacakları korkusuyla uzay mekiği gönderilmeden önce eklentileri olan e-postaları engellemek zorunda kalmıştır. Business Week, en son geliştirilen uzay araçlarına ait planların bilgisayar korsanları tarafından ele geçirildiğini bildirilmiştir.

2007

Estonya hükümet ağları, ülkenin bir savaş anıtının kaldırılmasıyla ilgili Rusya'yla anlaşmazlığa düşmesini takiben, bilgisayar korsanların yaptığı

hizmet reddi saldırısına (DDoS) maruz kalmıştır. Saldırılarda resmi kuruluşların, finans ve basın-yayın kuruluşlarının bütün iletişimi 3 hafta süreyle kesintiye uğramıştır.

2008

Yüzden fazla ülkede çeşitli hükümetlere ait kritik sistemlere sızan "GhostNet" adlı gizli bir casus ağ keşfedilmiştir. Bu ağın merkezinin Çin olduğu 10 aylık bir çalışmanın sonucunda bulunmuştur.

ABD' de Cumhuriyetçi ve Demokrat başkanlık kampanyalarının veri tabanları saldırıya uğramış ve bilgisayar korsanları tarafından indirilmiştir.

Gürcistan'ın bilgisayar ağları, ülkenin Rusya ile karşı karşıya geldiği bir dönemde saldırıya uğramıştır.

- *Estonya ve Gürcistan'a karşı gerçekleştirilen siber saldırılar üzerine NATO,2008 yılında Estonya'nın başkenti olan Talinn'de bir siber savunma merkezi kurdu.*

2010

İran'ın nükleer programına karşı geliştirildiği söylenen Stuxnet adlı solucan yazılım ortaya çıkmıştır. Natanz tesislerinde çalışan bir mühendisin diz üstü bilgisayarına kafedeyken bulduğu bir ajan tarafından bırakılmış olan USB bellek ile bulaştığı söylenmektedir. Kendini kopyalama yeteneğine sahip olan yazılım öncelikle motorlar ve sıcaklık kontrol merkezi olan mantık kontrol birimini ele geçirmiştir. Özellikle nükleer yakıt zenginleştirme tesislerini hedef alan bu saldırı, santrifüjlerin aşırı hızlanmasına yol açarak nükleer tesislere büyük zararlar vermiştir. İran'da hedef reaktörler aktif olmadığı için nükleer bir facia yaşanmamıştır. Eğer reaktörler aktif olsaydı Çernobil benzeri bir felaket yaşanabileceği belirtilmektedir.

- *Stuxnet ile ilk defa siber saldırıların sadece kritik altyapılara fiziksel olarak zarar verilebileceği görülmüştür.*
- *Stuxnet ile internete kapalı elektronik ortamlara diğer veri giriş yollarıyla (Cd, USB, vb.) saldırı yapılabileceği anlaşılmıştır.*
- *Stuxnet, yazılımın geliştiricileri tarafından henüz farkına varılmamış ve yazılımın kullanıma sürüldüğü halindeki açıklardan faydalanan saldırıları (zero day attack) kullanarak sistemleri ele geçirmiştir. Bu açıklar*

yazılımın geliştiricisi tarafından fark edildikleri ve yamanarak düzeltildikleri ana kadar kullanılabilirler.

2006 yılında kurulmuş olan WikiLeaks, kaynaklarının gizliliğini koruyarak hükümetlerin ve diğer organizasyonların hassas belgelerini yayınlayan, İsveç merkezli bir uluslararası organizasyondur ("Wikileaks"). 2010 yılında açıkladıkları diplomatik belgeler ile dünya çapında skandal yaratmıştır.

2011

Kanada hükümeti savunma birimlerine karşı önemli bir siber saldırı düzenlendiğini bildirmiştir. Saldırı, Kanada'nın ana ekonomi kuruluşları olan Maliye ve Hazine Kurulunun İnternet'ten ayrılmasına yol açmıştır.

2012

Rus firması Kaspersky, 2007'den beri faaliyet gösteren "Kırmızı Ekim" olarak adlandırılan, dünya çapında bir siber saldırıyı keşfetmiştir. Bilgisayar korsanları, Microsoft Word ve Excel programlarındaki güvenlik açıkları aracılığıyla bilgi toplamışlardır. Saldırının birincil hedefleri Doğu Avrupa, eski SSCB ve Orta Asya ülkelerinde olmakla birlikte, Batı Avrupa ve Kuzey Amerika da mağdurlar olduğu bildirdi. Virüs hükümet elçilikleri, araştırma firmaları, askeri tesisler, enerji sağlayıcıları, nükleer ve diğer kritik altyapılardan bilgi toplamıştır("Siber saldırıların tarihçesi").

2014

HSBC Türkiye'ye yapılan siber saldırı sonucu 2.7 milyon kullanıcının kredi kartı ve banka kartı bilgileri çalınmıştır. HSBC bu saldırıyı resmi olarak kabul etmiş ve müşterilere ait kart ve kartın bağlı bulunduğu hesap numarası, kart son kullanım tarihi ve kart sahibi ismine ulaşıldığını açıklamıştır.

2015

Türkiye'nin internetteki imzası olan ".tr" ile biten bütün internet sitelerini etkileyen 10 gün süren DDoS saldırı atağı gerçekleştirilmiştir. Bu saldırıların 12 farklı ülkeden eş zamanlı olarak yapıldığı belirlenmiştir.

23 Aralık 2015'te, Ukrayna'daki bölgesel enerji dağıtım şirketleri, koordine edilmiş bir siber saldırıya uğramış ve yaklaşık 225.000 müşteri çeşitli bölgelerde elektriksiz kalmıştır. Saldırganlar, endüstriyel kontrol sistemlerine uzaktan erişmek, bunları manipüle etmek, birden fazla Ukrayna merkezi ve

bölgesel tesisteki gücü kapatmak için çalıntı kullanıcı kimlik bilgilerini kullanmıştır.

2016

4 Şubat 2016'da Bangladeş merkez bankasında meydana gelen bir siber saldırı 81 Milyon dolar zarara neden olmuş ve işlemlerde 850 Milyon Dolara kadar başka işlemler yapılmasını engellemiştir.

Bordro şirketi ADP Mayıs 2016'da ABD Ulusal Bankası da dâhil olmak üzere yaklaşık 640.000 şirketten gelen bordro, vergi ve fayda bilgilerini açıklayan bir ihlali yaşamıştır. Şirketin müşteri portalındaki bir güvenlik açığını kullanarak bu bilgilere erişildiği anlaşılmıştır.

2017

Kendisini bir hava durumu uygulaması gibi gösteren Good Weather isimli kötü amaçlı yazılım tespit edilmiştir. Bu kötü amaçlı yazılım, ekran kilitleme kapasitesine de sahip ve hedef aldığı Android kullanıcılarının özellikle banka bilgilerini çalmıştır. Söz konusu kötücül yazılımım hedefleri arasında 22 Türk bankasına ait mobil uygulama da bulunduğu ve yaklaşık 5000 kullanıcının etkilendiği tahmin edilmektedir. Kötü amaçlı yazılımın, bir diğer yararlı hava durumu uygulaması olan Good Weather uygulamasının trojanlanmış versiyonu veya kopyası olduğu belirtiliyor.

(List of cyber attacks y.y., Siber Bülten – Dikkat: 22 Türk bankasının müşterileri siber saldırıya hedef oldu y.y., Siber saldırıların tarihçesi y.y., The three biggest cyber-attacks of 2016 y.y., Top 10 most notorious cyber attacks in history y.y.)

2.3.2. Dünyada Siber Güvenlik

Siber saldırılar siber uzay içerisinde gerçekleşmesine rağmen verilen zararlar gerçek hayatı etkilemekte ve sıkıntılara sebep olabilmektedir. Siber saldırılarla bir ülkenin trafik ışıklarından güç şebekelerine kadar tüm altyapılarını tahrip etmek mümkündür (Alkan 2013).Teknolojide görülen gelişmeler siber saldırıların hedeflerinin de büyümesine neden olmuştur. Siber saldırıların ilk dönemlerinde görülen bireysel hedefler seçen saldırılar yerine büyük çaplı saldırılar gerçekleştirilmeye başlanmıştır. Sonuç olarak ülkelerin ekonomileri, kritik altyapıları ve askeri güçleri tehlike altına girmiştir.

Siber saldırılarda öncelikli olarak gelişmiş ülkelerin hedef seçilmiş olması bu ülkelerde siber güvenlik ile ilgili farkındalık oluşmasını sağlamıştır. Bu ülkeler siber saldırılara karşı önlemler almış, kaynak ayırmış ve araştırmalar yapmışlardır. Örneğin ABD eski Başkanı Obama, ülkenin dijital altyapısını bir "stratejik ulusal varlık" ilan etmiştir. ABD Savunma Bakanlığı, ABD siber uzayını savunmakla görevli bir organ olan ABD Siber Komutanlığı'nı (USCYBERCOM) kurmuştur. İngiltere Kasım 2011'de Birleşik Krallık Siber "Güvenlik Stratejisi: Birleşik Krallığı dijital Dünya'da korumak ve desteklemek" isimli siber güvenlik belgesini ortaya koymuştur. Bu kapsamda siber güvenlik için 2009-2013 yılları arasında 650 milyon Sterlin bütçe ayrılmıştır. Çin Halk Cumhuriyeti, 2011 yılında Mavi Ordu isimli siber savaş biriminin varlığını açıklamıştır (Siber Savaş y.y.). Almanya siber güvenlik konusunda yasal ve düzenleyici önlemler alarak siber saldırılara karşı alınan önlemleri resmileştirmiştir. Ayrıca Rusya'nın da siber olaylarla ilgilenen birimleri mevcuttur.

Dünya çapında çalışmalar yapılarak ülkelerin siber güçleri araştırılmaktadır. Siber güç, ülkelerin siber saldırılara dayanma ve üretken ve güvenli bir ekonomi için gerekli dijital altyapıyı dağıtma becerisi olarak tanımlanmıştır (Hamilton 2014). Bu nedenle, siber güç kavramı, dijital kaynaklara güvenmenin faydalarını ve potansiyel zorluklarını kapsamaktadır. 2015 yılında ITU tarafından yayınlanan "Küresel Siber Güvenlik Göstergesi ve Siber Sıhhat Kesitleri" adlı raporda ülkelerin siber güçlerinin karşılaştırması yapılarak ülkelerin siber güçleri hakkında değerlendirme yapmasına yardımcı olmak hedeflenmiştir (ITU 2015). Söz konusu raporda ülkelerin siber güvenlikte; yasal önlemler, teknik önlemler, kurumsal önlemler, yetkinlik inşası ve uluslararası işbirliği konularında hangi noktada olduklarına dair puanlamalar yapılmıştır. Yapılan genel siber güç sıralamasında birçok ülke, aynı puanlara sahip olduklarından aynı sırayı paylaşmaktadır. Tablo-2'de gösterilen genel siber güç sıralamasında ABD listenin en başında gelmektedir. Aynı raporun "İyi Uygulamalar" başlığı altında ABD'nin gerçekleştirdiği bazı etkinlikler neden listenin başında olduğunu açıklamaktadır. Bu etkinlikler;

- Endüstriyel Kontrol Sistemleri Siber Acil Müdahale Ekibi (ICS-CERT) oluşturulmuştur.

- Ulusal Standartlar ve Teknoloji Enstitüsü kurulmuştur.
    - Kritik Altyapı Siber Güvenlik Sürüm 1.0 Geliştirme Çerçevesi hazırlanmıştır
    - Özel Yayın 800 serisi
    - Federal Bilgi İşleme Standardı mevcuttur.
    - Akıllı Üretim Sistemleri için Siber Güvenlik
- Siber Güvenlilik Eğitimi Ulusal Girişimi (NICCS)
    - Mesleki Sertifikalar verilmektedir.
    - Ulusal Siber Güvenlik İşgücü Çerçevesi oluşturulmuştur.

Tablo 2: Genel siber güç sıralaması ilk 5 derecedeki ülkeler

SIRALAMA	ÜLKE	İNDEKS
1	Amerika Birleşik Devletleri	0,824
2	Kanada	0,794
3	Avustralya	0765
3	Malezya	0,765
3	Umman	0,765
4	Yeni Zelanda	0,735
4	Norveç	0,735
5	Brezilya	0,706
5	Estonya	0,706
5	Almanya	0,706
5	Hindistan	0,706
5	Japonya	0,706
5	Kore Cumhuriyeti	0,706
5	Birleşik Krallık	0,706

(ITU 2015)

2.3.3. Türkiye'de Siber Güvenlik

Türkiye'de internet 1993 yılında kullanılmaya başlanmıştır ve günümüze kadar yaygınlığı çok büyük ölçüde artmıştır. Sonuç olarak bilgi güvenliği konusu gündeme gelmiş ve bilgi güvenliğinin idari, teknik, ekonomik ve hukuki yönleri ilgili çalışmalar yapılmaya başlanmıştır. Bu

çerçevede 2003 yılında "2003/10 sayılı Başbakanlık Genelgesi" yayınlanarak güvenlik konusunda ilk adımlar atılmıştır. 2004 yılında elektronik ortamdaki bilgilerin gizlilik ve bütünlüğünün garanti altına alarak kullanılması ve yapılan işlemlerin hukuki geçerliliğinin sağlanması için "Elektronik İmza Kanunu", sayıları ve çeşitleri artan bilişim suçlarına yönelik cezaları da içeren "Türk Ceza Kanunu" çıkarılmıştır (Ünver 2015).

2005 yılında da çalışmalar devam etmiş ve 25989 sayılı Resmi Gazete 'de "Telekomünikasyon Yoluyla Yapılan İletişimin Tespiti, Dinlenmesi, Sinyal Bilgilerinin Değerlendirilmesi ve Kayda Alınmasına Dair Usul ve Esaslar İle Telekomünikasyon İletişim Başkanlığının Kuruluş, Görev ve Yetkileri Hakkında Yönetmelik" in yayımlanmıştır.

2007 yılında siber saldırıları engellemek ve kötü niyetli içeriklerin yayınlanmasını engellemek adına 26530 sayılı Resmi Gazete 'de "İnternet Ortamında Yapılan Yayınların Düzenlenmesi ve Bu Yayınlar Yoluyla İşlenen Suçlarla Mücadele Edilmesi Hakkında Kanun" ve 26687 sayılı Resmi Gazete 'de "İnternet Toplu Kullanım Sağlayıcıları Hakkında Yönetmelik" yayımlanmıştır. Ayrıca bilgi güvenliği hakkında toplumsal farkındalığı geliştirmek amacıyla Bilgi Güvenliği Derneği " kurulmuştur.

Türkiye Bilgisayar Olayları Müdahale Ekibi (TR-BOME) koordinatörlüğünde 20 - 21 Kasım 2008 tarihlerinde ülkemizde ilk defa bilgi sistemleri güvenliği tatbikatı olan BOME 2008 Tatbikatı düzenlenmiştir(Bilgi Sistemleri Güvenliği Tatbikatı BOME 2008 - Ulusal Bilgi Güvenliği Kapısı y.y.). Cumhurbaşkanlığı, Başbakanlık, Adalet Bakanlığı, Sayıştay Başkanlığı, Hazine Müsteşarlığı, Merkez Bankası, Sermaye Piyasası Kurulu ve Tapu Kadastro Genel Müdürlüğü'nün katılımıyla gerçekleştirilen BOME 2008 tatbikatı, gerek kurum içi gerek kurum dışı olay müdahale süreçlerindeki eksikliklerin ortaya çıkartılması açısından çok faydalı bulunmuştur.

2009 yılının mayıs ayında Bilgi Teknolojileri ve İletişim Kurumu tarafından hazırlanan, "Siber Güvenliğin Sağlanması: Türkiye'deki Mevcut Durum Ve Alınması Gereken Tedbirler", TÜBİTAK –UEKAE bünyesinde hazırlanan "Ulusal Sanal Ortam Güvenlik Politikası" gibi yayınlarla siber güvenlik politikası oluşturulmaya başlanmıştır.

2010 yılında Siber güvenlik konusunda yapılan ulusal çalışmalara ek olarak uluslararası işbirliği çerçevesinde Avrupa Konseyi Sanal Suçlar Sözleşmesi imzalanmıştır. 22 Nisan 2014 tarih ve 6533 sayılı Sanal Ortamda İşlenen Suçlar Sözleşmesinin Onaylanmasının Uygun Bulunduğuna Dair Kanun ile onaylanmıştır. Bu sözleşme, bilgisayar suçlarını ve internet suçlarını gözeten ilk uluslararası sözleşmedir. Ulusal kanunların uyumlu hale gelmesini sağlayarak, araştırma tekniklerini geliştirerek ve ülkeler arası işbirliğini arttırmayı hedeflemektedir. Milli Güvenlik Kurulunun 27 Ekim 2010 tarihli bildirisinde siber tehdidin küresel düzeyde ulaştığı boyut ve bu tehdidin ulusal güvenliğe etkileri kapsamlı surette ele alındığı belirtilmiş ve bu bağlamda, siber tehdidin engellenebilmesi açısından milli düzeyde yürütülen çalışmalar değerlendirilmiştir.

2011 yılında siber güvenlikle ilgili çalışmalara devam edilmiş ve Ulusal Siber Güvenlik tatbikatı gerçekleştirilmiştir.

2012 yılında "Siber Güvenlik Hukuku Çalıştayı" düzenlenmiştir. Bu yılın devamında yapılan bir diğer çalışma ise siber saldırılara karşı önlem alma yeteneğinin kazandırılması amacıyla gerçekleştirilen Siber Kalkan 2012 tatbikatıdır. TSK Siber Güvenlik Merkezi ve TÜBİTAK bünyesinde Siber Güvenlik Enstitüsü kurulmuştur. Ayrıca 2012 yılının son çeyreğinde Ulaştırma, Denizcilik ve Haberleşme Bakanlığı (UDHB) "Ulusal Siber Güvenlik Çalışmalarının Yürütülmesi, Yönetilmesi ve Koordinasyonuna İlişkin Bakanlar Kurulu Kararı (BKK)" Resmi Gazete' de yayımlanmış, böylelikle siber güvenlik için hukuki zemin oluşturulmuştur. BKK ile Siber Güvenlik Kurulu oluşturulmuş ve Ulusal Siber Güvenliğin sağlanması görevi UDHB'ye verilmiştir.

2013 yılında UDHB bünyesinde "Siber Güvenlik Dairesi", Bilgi Teknolojileri ve İletişim Kurumu bünyesinde de "Ulusal Siber Olaylara Müdahale Merkezi (USOM)" kurulmuştur. UDHB tarafından hazırlanan "Ulusal Siber Güvenlik Stratejisi ve 2013-2014 Eylem Planı" 20.06.2013 tarihli Resmi Gazete 'de yayımlanmıştır. Eylem planında 7 ana başlık 29 ana eylem 86 alt eylem tanımlanmış ve bu eylemlerden sorumlu ve ilgili kurumlar belirlenmiştir. Ana başlıklar aşağıdaki gibidir:

- Siber güvenlik konusunda yasal düzenleme yapılması

- Uluslararası hukuktan kaynaklanan hakların kullanılması
- Ulusal siber olaylara müdahale organizasyonunun oluşturulması
- Ulusal siber güvenlik altyapısının güçlendirilmesi
- Siber güvenlik alanında insan kaynağının yetiştirilmesi
- Siber güvenlikte yerli teknolojilerin geliştirilmesi
- Ulusal güvenlik mekanizmalarının kapsamının genişletilmesi

Ulusal Siber Güvenlik Stratejisi ve 2013-2014 Eylem Planı'nın ilgili maddesi gereği UDHB, "SOME'lerin Kuruluş, Görev ve Çalışmalarına Dair Usul ve Esaslar Hakkında Tebliğini" 11.11.2013 tarihli Resmi Gazete 'de yayınlamıştır.

6.02.2014 tarihli ve 6518 sayılı kanunun 106ncı maddesiyle 5809 sayılı kanuna eklenen madde ile "Siber Güvenlik Kurulu" kurulmuş ve görevleri belirtilmiştir. Aynı kanunun 102.nci maddesiyle 5809 ayılı Kanunun UDHB'nin görevlerini belirleyen 5inci maddesine UDHB'nin görevleri arasına siber güvenlik kurulu sekretaryasını yapmanın yanı sıra ciddi görevler eklenmiştir.

2016-2019 Ulusal Siber Güvenlik Stratejisi ve Eylem Planı yayımlanmıştır. Strateji ve eylem planının temel amacının, siber güvenliğin, ulusal güvenliğin ayrılmaz bir parçası olduğu anlayışının tüm kesimlerde yerleşmesini sağlamak olduğu belirtilmiştir. (Ulastirma Bakanligi .). 5 ana eylem ve 41 alt eylemden oluşan strateji ve eylem planını ile ulusal siber uzayda bulunan sistem ve paydaşların tamamının güvenliğinin sağlanması hedeflenmektedir.

2017 yılının başında Ulusal Siber Olaylara Müdahale Merkezi, olası siber tehditlerin belirlenmesi ve güvenliğin sağlanması amacıyla devlet kadrosuna alım yapmak amacıyla "Siber Yıldız" ismi ile bir siber güvenlik yarışması düzenledi.

2003 yılından itibaren Türkiye'de yapılan siber güvenlik çalışmaları göz önünde bulundurulduğunda ülkemizin siber güvenlik konusunu ciddiye aldığı ve gelişme kaydetmek için uğraş verdiği görülmektedir. ITU tarafından yayınlanan siber güç endekslerinin bulunduğu raporda Türkiye'nin 195 ülke

arasından, 22. sırada ve Letonya ve İsveç ile beraber 7'nci en iyi göstergeye sahip olduğu görülmektedir (Şekil 7).

Şekil 7: Türkiye'nin Siber Güç Sıralamasındaki Yeri

(Canbek 2016)

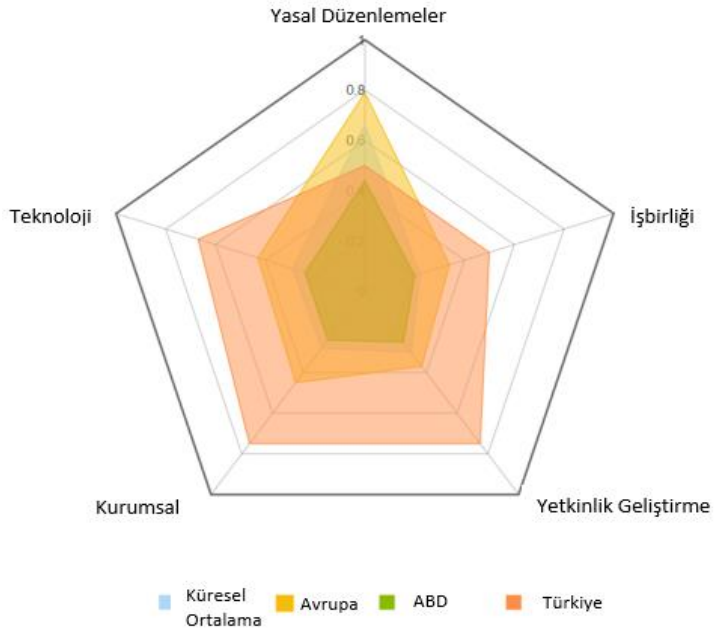
Bu rapor içinde Türkiye'nin durumuna bakıldığında kurumsal ve teknolojik olarak iyi durumda bulunulduğu görülmektedir. İyi Uygulamalar başlığı altında Türkiye'nin kurumsal uygulamaları yer almıştır. Bu uygulamalar;

- Ulusal Siber Güvenlik Stratejisi ve Eylem Planı 2013-2014.
  - Eylem planı 29 ana eylem ve 95 alt eylemi kapsamaktadır ve bu eylemler mevzuat, kapasite geliştirme, teknik altyapı geliştirme gibi konularda hedefler belirlemektedir.
- Siber güvenlikle ilgili önlemleri belirlemek, hazırlanan planları, programları, raporları, prosedürleri, ilkeleri ve standartları onaylamak ve bunların uygulanmasını ve koordinasyonunu sağlamak için Siber Güvenlik Kurulu kurulmuştur.
- Son üç yılda hem kamu hem de özel sektör katılımcıları ile ulusal düzeyde üç siber güvenlik tatbikatı düzenlenmiştir.

Tatbikatlar, siber güvenlik bilincinin artırılmasında büyük rol oynadı ve ayrıca siber güvenlik gelişimini ölçmek için mükemmel bir araç olmuştur.

İyi uygulamalarının yanı sıra Türkiye'nin yasal yaptırımlar konusunda diğer ülkelere kıyasla geride kaldığı gözlemlenmiştir (Şekil-8). Bu durumu çözebilmek için ceza hukukunda siber suçlara yer verilmeli ve veri koruma, ihlal bildirimi, belgelendirme / standardizasyon gereksinimleri gibi konularda yönetmelikler çıkarılmalıdır. Ülkemizin puan konusunda geride kaldığı bir diğer kriter "İşbirliği" kriteridir. Siber güvenlik, tüm sektörlerden ve disiplinlerden gelen girdileri gerektirir ve bu nedenle çok paydaşlı bir yaklaşımla ele alınması gerekmektedir (Canbek 2016). İşbirliği konusuna önem verilerek kurum ve kuruluşlar arasında bilgi alışverişi ve koordinasyonu artırarak daha kapsamlı bir siber güvenlik alanının oluşturulması hedeflenmelidir. 2013-2014 Ulusal Siber Güvenlik Stratejisi ve Eylem Planında işbirliğini destekleyen hedefler bulunmaktadır.

Şekil 8: Türkiye'nin Dünya ülkeleri ile karşılaştırılması



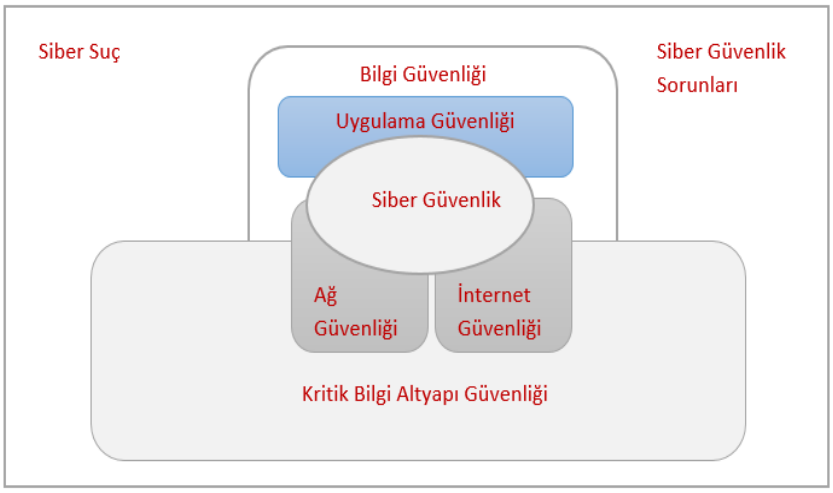
(Canbek 2016)

3. GÜVENLİ UYGULAMA YAZILIMI GELİŞTİRME

Bilgi toplumu, yoğun bir şekilde bilginin üretildiği, işlendiği, toplanıp dağıtıldığı ve buna bağlı olarak yeni bir sosyal değişimin oluştuğu bir yapıyı ifade etmektedir (Bozer 2013). Bilgi toplumu olma yolunda yazılımların insan hayatındaki yeri ve önemi sürekli olarak artmıştır. Günümüzde hemen herkesin kullandığı sıradan bir cep telefonunda yaklaşık 15 milyon satır, bir arabada ise 50 milyon satır kod bulunması yazılımların ne kadar sıklıkla kullanıldığının bir göstergesidir (Barış Can Kaşıkçı 2009).

Yazılımların geliştirilmeye başlandığı ilk dönemlerden itibaren göz önünde bulundurulan güvenlik konusu, internetin gelişmesi ve internet ortamındaki tehditlerin de artmasıyla birlikte günden güne önem kazanmıştır. Hewlett Packard Enterprise tarafından hazırlanan ve her yıl yayınlanan "Siber Risk Raporu" adlı bir çalışmanın 2016 versiyonunda, saldırganların uygulama yazılımlarını hassas kurumsal verilere erişmenin en kolay yolu olarak gördüğü belirtilmektedir (Hewlett Packard Enterprise 2016). Buradan da anlaşılabileceği üzere uygulama yazılımları güvenliği, bilgi güvenliğinin ve siber güvenliğin en önemli unsurlarından birisidir. Uygulama yazılımlarında bulunan tek bir zafiyet hassas verilerin kötü niyetli kişilerin eline geçmesine veya bütün sistemin ele geçirilmesine sebep olabilmektedir. Uygulama yazılımlarında bulunan güvenlik açıklıkları çok uzun süreler fark edilmeden kalabilmekte ve fark edilinceye kadar verilerin gizlilik, bütünlük ve erişilebilirlik özelliklerinin ihlal edilmesine yol açmaktadır. Siber Risk Raporunda 2015 yılında en çok kullanılan uygulama yazılımı güvenlik açığının beş yaşın üstünde olduğu ve aynı açığın 2014 yılında da en çok kullanılan açık olduğu belirtilmiştir (Hewlett Packard Enterprise 2016). Bu güvenlik açığı uygulama yazılımı sahibi tarafından iki kere yama yapılarak kapatılmaya çalışılmış ancak başarılı olunamamıştır. Bilgi güvenliği, siber güvenlik ve uygulama yazılımı güvenliğinin birbirleriyle olan ilişkisi Şekil 9'da gösterilmiştir.

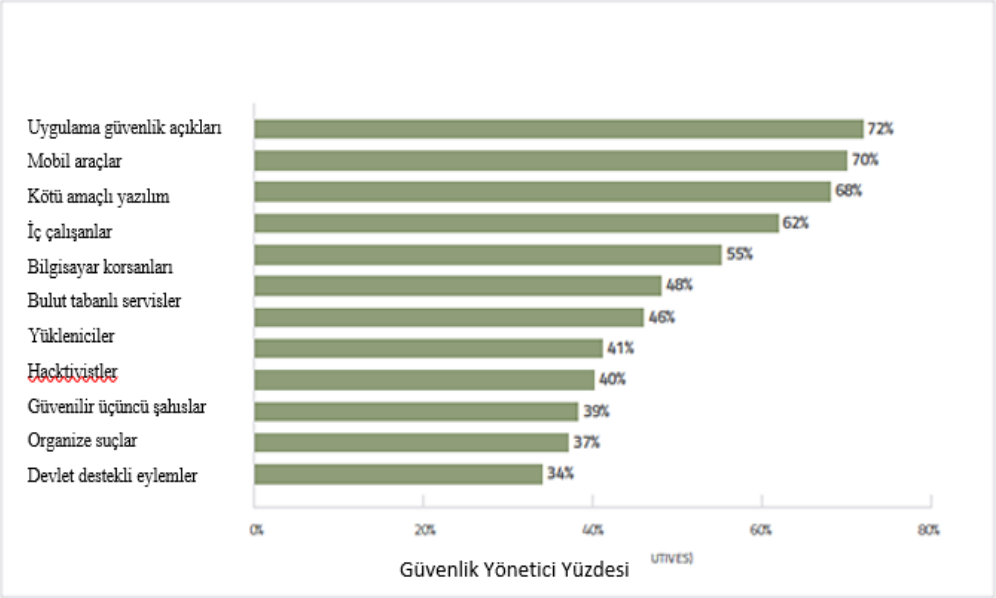
Şekil 9: Bilgi Güvenliği, Siber Güvenlik ve Uygulama Güvenliği İlişkisi



(ISO 2012)

Kâr amacı gütmeyen bilgi güvenliği eğitimi ve sertifikaları konusunda uzmanlaşmış bir organizasyon olan Uluslararası Bilgi Sistemi Güvenlik Sertifikasyon Konsorsiyumu (ISC)² tarafından 2013 yılında güvenlik yöneticileri arasında yapılan bir araştırmada, siber güvenlik konusunda en çok endişeye sebep olan konunun uygulama yazılımlarında bulunan güvenlik açıkları olduğu tespit edilmiştir. Araştırmaya konu olan güvenlik açıklıkları ve sıralamaları Şekil 10' da gösterildiği gibidir.

Şekil 10: Güvenlik Tehditleri: En Çok Endişe Edilen Konular



(Richardson 2013)

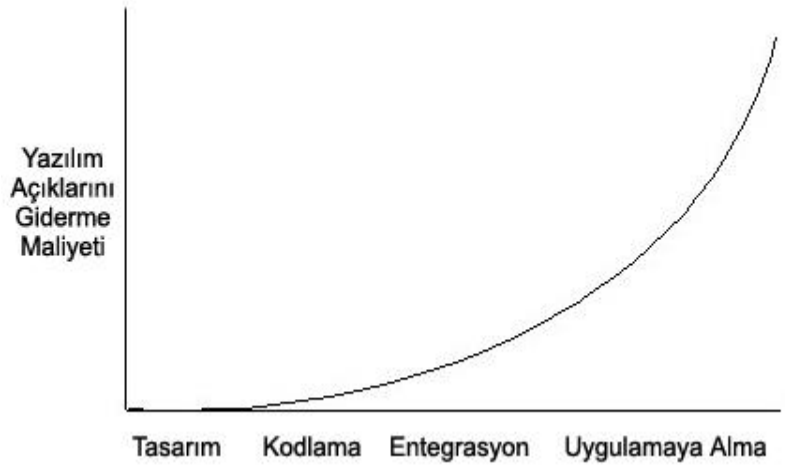
Siber uzaydaki önemli risklerden birisi olan uygulama yazılımı güvenlik açıklıklarını engellemenin yolu güvenli yazılımların geliştirilmesidir. Güvenli yazılım; işlediği ve ulaşabildiği verinin bütünlüğünü, gizliliğini korumalıdır. Bununla birlikte bünyesinde barındırdığı bilgiye erişimin de devamlılığını sağlamalıdır (Demir 2015). Bir başka deyişle yazılımın kötü niyetli saldırılara ve kurcalamalara karşı belirlenmiş seviyede dayanıklılığa sahip olmalıdır.

Yazılım güvenliği, yazılım geliştirme tamamlandıktan sonra güvenlik açıklarını ortadan kaldırmak ve sızma testleri yapmaktan ibaret değildir. Yazılım geliştirme aşamaları tamamlandıktan sonra yapılan sızma testleri sadece minimum seviyede güvenlik sağlamaya yardımcı olmaktadır. Bu durum Savunucunun İkilemi (Defender's Dilemma) ile açıklanabilir. Saldırganın yalnızca bir giriş noktasına ihtiyacı olduğu halde, savunma yapan kişinin tüm noktaları savunması gerekmektedir. Bazı açıkları kapatmak güvenliği sağlamak için yeterli olmamaktadır. Bunun yerine saldırganların kullanılabileceği tüm açıkların kapatılmasını gerektirmektedir. Bu da ancak yazılım geliştirme sürecinin her aşamasında güvenlik konusunun dikkate alınmış olması ve bilgi güvenliği kontrollerinin sağlanmış olmasıyla gerçekleştirilebilir. Bahsedilen bilgi güvenliği kontrolleri ISO 27001 ve ISO 27002 standartlarında toplanmıştır. Standartlarda verilen kontrollere ek olarak Güvenli yazılım geliştirme süreçlerinde; geliştirme, test ve üretim ortamı

ayrışımı, geliştirme ortamında test verilerinin kullanılması, üretim ortamına almadan önce kaynak kodların incelenmesi, güvenli programlama teknikleri kullanımı, uygulama güvenlik duvarı kullanımı ve yapılandırma yönetimi, geliştirilen yazılımın güvenliğine destek olacak hususlardır.

Yazılım geliştirme süreçlerinde erken fark edilen yazılım açıklarının kapatılmasının daha ileri bir evrede fark edilen açıkların kapatılmasına nazaran daha az maliyetli olacağı yazılım endüstrisince yaygın olarak kabul edilen bir ilkedir (Özbilgin, Gökhan Özlü 2010). Güvenlik açığı, geliştirme sürecinde ne kadar geç bulunursa, güvenlik açığını ortadan kaldırmak için sorunun tam doğasını tanımlamakta aynı derecede uzun sürecek ve revize edilecek daha fazla kod ortaya çıkacaktır. Güvenli yazılım geliştirme ilkelerinin yazılım projelerine ilk aşamadan itibaren dâhil edilmesi yazılım geliştirme maliyetini, süresini azaltmayı ve yazılımın kalitesini arttırmayı mümkün kılmaktadır. Ayrıca yazılım kullanımı sırasında ortaya çıkabilecek pek çok güvenlik sorununu da kaynağında engellemenin en etkili yoludur.

Şekil 11: Yazılım geliştirme aşamalarına göre yazılım açıklarını giderme maliyeti (Zaman ve masraf)



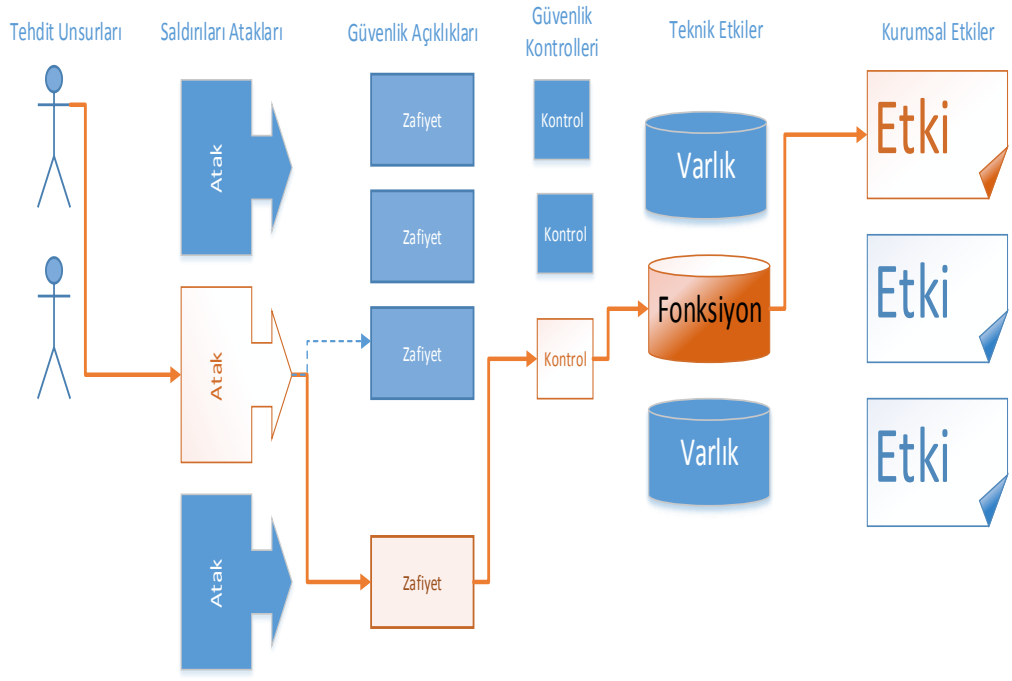
(C. C., Micheal, van Wyk, Ken 2005)

3.1. En Sık Karşılaşılan Uygulama Güvenlik Riskleri:

Güvenlik riski, belirli bir tehdidin bir varlık veya varlık grubunun güvenlik açıklarını kullanması ve böylece kuruluşa zarar verme potansiyelidir.

'Güvenlik açığı/ Zafiyet' terimi, çoğunlukla yazılımların en zayıf oldukları noktalar için kullanılan terimdir. 'Güvenlik açığı/ Zafiyet' bir veya daha fazla tehdit tarafından sömürülebilecek bir varlık veya varlık grubunun bir zayıflığı olarak tanımlanmaktadır. Tehdit unsurları, saldırı yaptıkları kurumlara zarar vermek için uygulamalar aracılığıyla birçok farklı yol kullanabilmektedirler. Bu yolların her biri, bir riski temsil etmektedir.

Şekil 12: OWASP -Tehdit unsurları ile Kurumsal etkiler arasındaki ilişki



(Owasp 2013)

Kurumların güvenli uygulamalar tasarlamasını, geliştirmesini, elde etmesini, kullanmasını ve bakımını sağlamaya adanmış kar amacı gütmeyen bir örgütlenme olan Açık Web Uygulaması Güvenlik Projesi kapsamında belirli aralıklarla küresel olarak en kritik güvenlik risklerini sıralandığı

OWASP Listesini yayınlamaktadır. OWASP Listesi yazılım geliştiriciler tarafından referans olarak kabul görmektedir. Bu listenin amacı kurumların karşılaştığı en kritik risklerden bazılarını belirleyerek uygulama güvenliği konusunda farkındalık yaratmaktır(Owasp 2013). Liste en son 2013 yılında güncellenmiştir ve Tablo-3'te gösterilmiştir.

Tablo 3: OWASP Uygulamalarda En Sık Karşılaşılan 10 Güvenlik Açığı Listesi

SIRA	**ADI**
A1	Enjeksiyon Açıkları
A2	İhlal Edilmiş Kimlik Doğrulama ve Oturum Yönetimi
A3	Siteler Arası Betik Yazma (XSS)
A4	Emniyetsiz Doğrudan Nesne Referansı
A5	Yanlış Güvenlik Yapılandırması
A6	Hassas Bilgi Sızıntısı
A7	Eksik İşlev Seviyesi Erişim Kontrolü
A8	Siteler Ötesi İstek Sahteciliği (CSRF)
A9	Bilinen Güvenlik Açığı olan Bileşenleri Kullanma
A10	Doğrulanmamış yönlendirmeler

(Owasp 2013)

Tablo 3'te verilen güvenlik açıklarına ve alınması gereken aksiyonlara dair açıklamalar aşağıda verilmiştir.

3.1.1. A1- Enjeksiyon Açıkları:

SQL, OS komutları ve LDAP enjeksiyonu gibi enjeksiyon kusurları, uygulama ara yüzünden alınan verilerin kontrol edilmeden komut olarak işlenmesiyle ortaya çıkmaktadır. Saldırganlar, uygulamaları istenmeyen komutları işletmesi veya uygun yetkilendirme olmadan veriye erişmek, değiştirmek, silmek, vb. için kandırabilmektedirler.

Örneğin aşağıdaki kod parçasına saldırgan, tarayıcısında 'id' parametre değerini 'ya da' 1 '=' 1 olarak göndermek üzere değiştirebilir.

```
2  string query = "SELECT * FROM hesaplar WHERE kullaniciID='" + request.getParameter("id")+ "'";
```

```
1  http://ornek.com/app/hesapView?id= 'veya' 1 '=' 1
```

Bu sayede, hesaplar tablosundaki tüm kayıtları döndürmek için sorgunun anlamı değiştirilmiş olmaktadır. Daha tehlikeli saldırılar veriyi değiştirebilmekte veya saklı yordamlar çağırabilmektedir.

Alınabilecek Aksiyonlar

- Girilen verilerin doğrulanması gerekmektedir.
- API'lerde güçlü sorgu parametre tipleri kullanılmalıdır.
- En düşük ayrıcalık verilmelidir. (Veritabanına sadece okuma yetkisi gibi)
- Detaylı hata mesajlarından kaçınılmalıdır.

3.1.2. A2- İhlal Edilmiş Kimlik Doğrulama ve Oturum Yönetimi

Saldırganların, Kimlik Doğrulama ve Oturum Yönetim fonksiyonlarında bulunan açıkları kullanarak hesap bilgilerini (kullanıcı adı, şifre vb.) ve oturum parametrelerini ele geçirmeleridir. Saldırganlar kullanıcının diğer bilgilerini elde etmek için şifreleri ve kimlik denetimi anahtarlarını kullanabilmektedirler.

Kimlik doğrulama mekanizmasında bulunan kusurlar bu saldırılara olanak sağlamaktadır. Ayrıca; çıkış, parola yönetimi, zaman aşımı, beni hatırla, gizli soru ve hesap güncelleştirmesi gibi yardımcı kimlik doğrulama fonksiyonlarının denetlenmesi bu saldırı türünün engellenmesinde önemli rol oynamaktadır.

Kimlik doğrulama işlemi, güvenli haberleşme ve kimlik bilgilerinin güvenli olarak saklanmasına dayanmaktadır.

Alınabilecek Aksiyonlar

- Kullanıcılar tekil olarak tanımlanmalıdır. Bir başka deyişle veritabanı seviyesinde “sa”, işletim sistemi seviyesinde “root”, uygulama seviyesinde “admin” gibi genel kullanıcı isimleri kullanılmamalıdır.
    - Aksi durumda sorgulana bilirlik ve hesap verebilirlik açısından sorun teşkil edebilir.
    - Bu hesaplar kullanılacaksa belirli aralıklarla yetki kontrolleri yapılmalı ve parolaları değiştirilmelidir.
- Kullanıcı parolalarının 60/90 gün içinde zaman aşımına uğraması, minimum 6/8 karakter uzunluğunda olması, karmaşık şifre kontrollerinin uygulanması, en çok 3 hatalı giriş sonunda kullanıcı hesaplarının kilitlenmesi, parolaların 1 gün içinde en fazla 1 kere değiştirilebilir olması gibi özelliklerin bulunması gerekmektedir. Ayrıca daha önce kullanılan son 3 parolanın tekrar kullanılmasının engellenmesi güvenlik açısından önemlidir.
- Merkezi ve güvenli bir kimlik doğrulama mekanizması olmalıdır.
- Tüm kimlik bilgileri, kriptografik özet veya şifrelenen formda depolanmalıdır.
- Her sayfada “oturumu kapat, güvenli çıkış, vb.” gibi oturumu kapatmayı sağlayacak bağlantı noktası ve önceden belirlenmiş bir sürede oturumu otomatik kapatmayı sağlayacak zaman aşımı mekanizması bulunmalıdır.

3.1.3. A3- Siteler Arası Betik Yazma (XSS)

Web uygulamalarının tamamı, Siteler Arası Betik Yazma saldırıları için uygun ortamlardır. Web tarayıcıları üzerindeki uygulamaların alınan verileri gerekli doğrulamalar ve denetimler yapılmadan çalıştırması sonucunda saldırganların web tarayıcısı üzerinden zararlı kodları çalıştırmalarıdır. XSS saldırganın kurbanın tarayıcısında kullanıcı oturumları bilgilerin çalınmasına, web sitesinin tahrif edilmesine veya solucan yüklenmesine sebep olan betik çalıştırmasına izin vermektedir.

Saldırılarda çoğunlukla betik dili olan Javascript kullanılır. Saldırganlar Javascript kullanarak, web sayfasının görünümü değiştirebilmektedir. Örneğin saldırganın sitesine kullanıcı bilgileri gönderecek bir oturum açma elemanı

eklenmesi mümkün olabilmektedir. Ayrıca Javascript, XmlHttpRequest komutunun kullanımına olanak vermektedir ve bu komut genel olarak AJAX kullanan uygulamalarda çalıştırılıyor olmasına rağmen AJAX kullanmayan siteler de XSS saldırılarına hedef olmaktadır. XmlHttpRequest kullanımıyla, kurbanların verileri saldırganın sitesine yönlendirilebilmekte ve karmaşık solucan ve zararlı zombi kodları oluşturularak web tarayıcısı çalıştığı sürece kullanılabilmektedir.

Alınabilecek Aksiyonlar

- Gelen verilerin doğrulanması gerekmektedir.
- Çıkan veriler güçlü bir şekilde şifrelenmelidir. Bütün kullanıcı kaynaklı veriler gönderilmeden önce (HTML veya XML'e bağlı bir çıkış mekanizmasına) uygun bir şekilde kodlanmalıdır.
- Belirli Çıkış verisi kodlaması kullanılmalıdır.(IS0 8859-1 veya UTF 8 gibi)
- Kara liste araçları yerine beyaz liste araçları tercih edilmelidir.
- Varsayılan hata çıktıları izlenmelidir.

3.1.4. A4- Emniyetsiz Doğrudan Nesne Referansı

Yazılım geliştiricilerin bir dosya, dizin veya veri tabanı kaydı gibi bir iç uygulama nesnesine doğrudan erişim sağlayarak ekleme, güncelleme, silme işlemlerinde bilgiyi URL veya form parametresi olarak aldığı durumlarda gerçekleşmektedir. Bu durumda erişim kontrolü sağlanmamışsa, saldırganlar referansı manipüle ederek sunucu üzerinde bulunan nesnelere yetkisiz erişim sağlayabilmektedir.

Örneğin, eğer kod, dosya adları veya yolları belirtmesi için kullanıcı girişine izin veriyorsa, uygulamanın çalışma dizininden dışarı ulaşması ve diğer kaynaklara erişmesi için saldırganlara da izin verebilir. Aşağıdaki örnekte yetkisiz erişim için açık bir bağlantı olmasa da saldırgan "ogrenciID" parametresini değiştirerek herhangi bir öğrenci bilgisine ulaşabilir.

```
1   int ogrenciID = Integer.parseInt( request.getParameter( "ogrenciID" ) );
```

Alınabilecek Aksiyonlar

- Kullanıcılara özel nesne referansları açıklanmamalıdır (birincil anahtarlar, dosya adları, vb.)
- Nesne referansları için beyaz liste yaklaşımı uygulanmalıdır.
- Referans gösterilen nesneler için kimlik doğrulaması ve erişim kontrolü yapılmalıdır.

3.1.5. A5- Yanlış Güvenlik Yapılandırması

Güvenlik; uygulama, çerçeveler, uygulama sunucusu, web sunucusu, veri tabanı sunucusu ve platform için tanımlanmış ve konuşlandırılmış güvenli bir yapılandırmaya sahip olmayı gerektirmektedir. Yanlış güvenlik yapılandırmaları, saldırganlar tarafından ayrıcalıklı verilere erişmelerine izin verecek zayıf alanları tespit etmek için kolayca kullanılabilmektedir. Varsayılan değerler genellikle yeterli güvenlik sağlayamamaktadır.

Alınabilecek Aksiyonlar

- Sunucular, platformlar vb. dâhil tüm uygulama ortamının konfigürasyonu doğru bir şekilde tanımlanmalı, uygulanmalı ve denetlenmelidir.
- Yazılımlar en güncel sürümleriyle kullanılmalıdır.

3.1.6. A6- Hassas Bilgi Sızıntısı

Pek çok uygulama kimlik bilgileri, vergi bilgileri gibi hassas verileri bünyesinde bulundurur. SSL ve HTTPS gibi güvenlik kontrolleri düzgün şekilde uygulanmadığında, bu hassas veriler güvenlik açıkları vasıtasıyla sızdırılabilmekte veya çalınabilmektedir. Saldırganlar, zayıf şekilde korunan bu verileri dolandırıcılık, kimlik hırsızlığı veya diğer suçları işlemek için kullanabilmektedir.

Alınabilecek Aksiyonlar

- Erişim kontrolü yapılmalıdır.
- Hassas veriler şifrelenmeli, şifreli olarak saklanmalı ve taşınmalıdır.
- Kullanılan SSL sertifikaları güncel olmalı ve doğruluğu kontrol edilmelidir.

- Veri kaybı/sızıntısı önleme yazılımları kullanılmalıdır. DLP yazılımları ile kurum dışına çıkarılması istenmeyen verilerin çıkışını önlenebilir ya da belirlenen dosyaların kullanım durumları izlenebilmektedir.

3.1.7. A7- Eksik İşlev Seviyesi Erişim Kontrolü

Uygulamalarda, web ara yüzüne erişim denetimi sağlandığı ancak fonksiyon seviyesinde erişim denetiminin yapılmadığı durumlarda ortaya çıkmaktadır. Bununla birlikte, uygulamalar, her bir işleve erişildiğinde sunucuda aynı erişim kontrolü denetimlerini gerçekleştirmelidir. İstekler doğrulanmazsa, saldırganlar yetkilendirmeden işlevlere erişebilmek için istekte bulunabilir.

Örneğin aşağıda verilmiş olan URL'ler kimlik doğrulaması gerektirmektedir. "admin_getappInfo" sayfasına erişebilmek için yönetici haklarına sahip bir kullanıcı ile giriş yapılması gerekmektedir.

http://example.com/app/getappInfo

http://example.com/app/admin_getappInfo

Kimliği doğrulanmamış bir kullanıcı her iki sayfaya da erişebilirse, bu durum bir zafiyet olarak tanımlanabilmektedir. Kimliği doğrulanmış, yönetici olmayan bir kullanıcının "admin_getappInfo" sayfasına erişmesi halinde, bu da bir zafiyet olup, saldırganın uygulamanın daha derinlerinde bulunan birçok veriye erişmesine, değiştirmesine veya veriye zarar vermesine olanak sağlanmış olmaktadır.

Alınabilecek Aksiyonlar

- Uygulamalarda varsayılan olarak tüm erişim reddedilmeli ve her işleve erişim için belirli rollere açık bir şekilde izin verilmelidir.
- İşlev bir iş akışı içeriyorsa, erişime izin vermek için koşulların yerine getirildiğinden emin olunmalıdır.

3.1.8. A8- Siteler Ötesi İstek Sahteciliği (CSRF)

Bir CSRF saldırısı, oturum açıldığında, saldırıya maruz kalan bir uygulamada, kullanıcının kimlik doğrulaması bilgilerinin başka bir web sayfasında da kullanılarak yetkisiz erişim sağlanmasıdır.

Alınabilecek Aksiyonlar

- CSRF saldırılarının önüne geçebilmek için, uygulamalarda öncelikle XSS açıklıklarının belirlenmesi gerekmektedir. Bunun sebebi XSS açıkları CSRF savunmalarını aşmak için kullanılabilmektedir.
- Uygulamalarda, tarayıcılar tarafından otomatik olarak yollanan izin belgesi veya işaretlerine güvenilmemelidir. CSRF saldırısına karşı korunma sağlayabilmek için, tarayıcının hatırlayamayacağı özel bir işaret (rastgele sayılar gibi) kullanılmalıdır. Kullanılan özel işaretlerin doğruluğu kontrol edilmelidir. Aşağıdaki kod parçasında bu duruma örnek verilmiştir.

```
1  <form action="/transfer.do" method="post">
2    <input type="hidden" name="479043245" value="454578621">
3    …
4  </form>
```

- Hassas veriler için, tekrar kimlik doğrulanması yapılmalı veya isteğin gerçek olduğunun ispatı için işlem işareti kullanılmalıdır.

3.1.9. A9 - Bilinen Güvenlik Açığı Olan Bileşenleri Kullanma

Bileşenler, örneğin kütüphaneler, çerçeveler ve diğer yazılım modülleri, neredeyse her zaman tam ayrıcalıklarla çalışır. Güvenlik açığı bulunan bir bileşen kullanılması durumunda, burada yer alan açıkları kullanarak verilere yetkisiz erişim sağlamak mümkün olabilmektedir. Bu sebeple önceki yıllarda geliştirilmiş ve bilinen açıkları olan kütüphane veya uygulama yazılım modüllerinin kullanılmasından kaçınılmalıdır.

3.1.10. A-10 Doğrulanmamış Yönlendirmeler

Web uygulamaları kullanıcıları başka sayfalara ve web sitelerine yönlendirebilmektedirler. Bu yönlendirmelerin denetlenmediği durumlarda kullanıcılar kimlik avı veya kötü amaçlı yazılım sitelerine yönlendirilebilmekte, kişisel kimlik bilgilerinden vazgeçirmek veya yetkisiz sayfalara erişmek için iletileri kullanılabilmektedir.

Örneğin, "url" adında tek bir parametre alan "redirect.jsp" adlı bir web sayfası bulunan uygulamada saldırgan, kullanıcıları kimlik avı gerçekleştiren

ve kötü amaçlı yazılım yükleyen siteye yönlendiren kötü amaçlı bir URL hazırlayabilmektedir.

```
4   http://www.ornek.com/redirect.jsp?url=kotuamacliyazilim.com
```

Alınabilecek Aksiyonlar

- Yönlendirmelerden kaçınılmalıdır.
- Eğer yönlendirmeler kullanılmışsa hedef noktası parametre olarak alınmamalıdır.
- Parametre kullanımı kaçınılmaz ise yönlendirme parametresinin geçerliliği ve kullanıcı için yetki durumu kontrol edilmelidir.
- Bilgi güvenliği hususunda kurum bünyesinde farkındalığı arttıracak çalışmalar desteklenmelidir.

3.2. Uygulama Yazılımı Geliştirme Süreçlerinde Ele Alınması Gereken Temel Güvenlik Konuları:

Güvenli uygulama yazılımı geliştirebilmek için mutlaka ele alınması gereken bazı temel güvenlik hususları bulunmaktadır. Bu hususların kullanıcı tarafından talep edilmeden ve yazılımcının inisiyatifine bırakılmadan ele alınması gerekmektedir. Ancak bu hususların yazılımın analiz aşamasından başlayarak tüm yaşam döngüsü boyunca geliştirme süreçlerine dâhil edilmesi durumunda gerçek güvenlik söz konusu olacaktır.

Temel güvenlik konuları aşağıda listelenmiştir ve takip eden bölümlerde açıklanmıştır.

- Girdi doğrulama
- Kimlik doğrulama
- Yetkilendirme
- Oturum Yönetimi
- Konfigürasyon dosyaları yönetimi
- Hassas bilgi
- Kriptografi
- Parametre manipülasyonu
- Hata Yönetimi
- Kayıt tutma ve Denetim

3.2.1. Girdi Doğrulama

Günümüzde kullanılan uygulamaların en belirgin özelliklerinden biri kullanıcıdan bir girdi (istek) alarak buna karşılık olarak bir cevap üretmeleridir (Demir 2015). Yazılımlarda karşılaşılan güvenlik açıklarının büyük bir kısmı doğrulama hatalarından kaynaklanmaktadır. Girdi doğrulamasından kaynaklanan güvenlik açıkları, birçok problem ve zafiyete neden olabilir. Bu sebeple girdi doğrulama işinin düzgün yapılması güvenlik konusunda büyük önem taşır. Aşağıda, yalnızca girdiyi doğrulayarak çözülebilecek bazı güvenlik açıklarının listesi verilmiştir.

- Arabellek taşmaları
- Enjeksiyon atakları
- Hizmet Dışı Bırakma (DoS) saldırıları
- Bellek sızıntısı
- Bilgi ifşası

Girdi doğrulama, hatalı biçimlendirilmiş verilerin sisteme girmesini en aza indirgemek için gerçekleştirilir. Uygulamalarda girdi noktalarında bulunabilecek açıklar bütün sistemin ele geçirilmesine neden olabilir. Uygulamalarda girdi noktaları HTTP protokolü parçalarından (Url, http başlıkları, http gövdesi, vb.), Uzak Metot Çağrısı gibi farklı protokollerden veya Basit Nesne Erişim Protokolü (SOAP) gibi alt protokollerden oluşabilmektedir. Aşağıda örnek girdi noktaları verilmiştir.

```
1  <a href="/kitap.do?kitap_id=5">Kitap oku</a>
2
3  <form action="kayit.asp" method="post">
4  ad:<input type=text name="name"/><br/>
5  Soyad:<input type=text name="lastname"/><br/>
6  <input type=hidden name="user" value="1"/>
7  <input type="submit" value="sent"/>
8  </form>
9
```

Girdi doğrulama için kara liste veya beyaz liste uygulaması yapılabilmektedir. Kara liste uygulamasında; girdinin kabul edilebilir veri içerip içermediği kontrol edilir, eğer uygun veri içermiyorsa girdi kara listeye eklenir. Beyaz liste uygulamasında ise sadece beklenen veri özelliklerine sahip verilerin kabul edilmesine dayalı bir doğrulama rutini işletilmektedir. Bunu yapmak için bütün kabul edilebilir girdi değerlerinin ya da koşullarının bir

listesi oluşturulmaktadır. Beyaz liste uygulamasına örnek olarak bir kitap kategori numarası beklenen uygulama alanı için şu örnek kod yazılabilir:

```
1  var kategoryID = Request.QueryString["KetegoriID"];
2  var positiveIntDeg = new Deg(@"^0[1-9][0-9]$");
3
4  if(!positiveIntDeg.IsMatch(kategoryID))
5  {
6    lblResults.Text = "Geçersiz kategori numarası girilmiştir.";
7    return;
8  }
```

Tavsiye edilen girdi doğrulama yöntemi beyaz liste uygulamasıdır. Böylece neyin istenildiğini ve kabul edildiği belirtilmiş olmaktadır.

3.2.2. Kimlik Doğrulama

Uygulamayı kullanacak kişinin geçerli ve tanımlı bir kullanıcı olup olmadığının kontrol edilmesi işine "Kimlik Doğrulama" denmektedir. Kimlik doğrulama uygulamalarda yetkisiz erişimi engellemek güvenlik unsurlarının vazgeçilmez bir parçasıdır. Kimlik doğrulama mekanizmasında meydana gelecek bir hata saldırganın uygulamayı kontrol etmesine ve yönetici seviyesinde yetki sahibi olmasına sebep olabilmektedir. Kimlik doğrulama işleminin 3 temel faktörü bulunmaktadır (Demir 2015).

1. Kullanıcının bildikleri; Kullanıcı adı, parola, vb.
2. Kullanıcının sahip olduğu; Cep Telefonu, OTP (One Time Password) cihazları, e-imza tokenı, yeni çipli T. C. Kimliği, vb
3. Kullanıcının kendisi; Parmak izi, Retina, Yüz tanıma vb.

Bu faktörler tek başlarına veya kombine edilerek kimlik doğrulama işlemi için kullanılabilmektedir. Kimlik doğrulama işleminin kullanılacağı uygulamanın sahip olduğu bilgilerin kritiklik seviyesine göre kullanılacak kimlik doğrulama faktörlerinin seviyesi belirlenmelidir.

Kimlik doğrulama işlemleri için farklı yöntemler kullanılabilmektedir. Bunların en sık kullanılanları aşağıda verildiği gibidir:

1. Temel Kimlik Doğrulama Yöntemi (Basic):

İnternet ile ilgili çalışmalarda kaliteyi sağlamak için standartlar belirleyen bir topluluk olan İnternet Mühendisliği Görev Birimi (Internet Engineering Task Force) tarafından oluşturulmuş olan RFC 2617 adlı standartta Temel Kimlik Doğrulama Yöntemi açıklanmıştır (Franks,j, Hallem-

Baker,P. , Hostetler 1999). Bu yöntemde sunuculara yetkisiz erişimin engellenmesi hedeflenmektedir. Kullanıcı adı ve parola ile kontrol yapılır.

Bu yöntemin avantajı yöntemin basit olması ve kısa sürede uygulanabilmesidir. Dezavantajlarının başında kullanıcı adı ve parolanın şifreli olmaması gelir. HTTP protokolü ile kullanıldığı zaman kullanıcı adı ve parola elde edilmesi mümkündür. Bunu engellemek için HTTPS protokolü ile birlikte kullanılması önerilmektedir. Bir diğer dezavantajı ise oturum kapama özelliğinin olmayışıdır. Kullanıcı oturumu kapatmadığı sürece bağlantı açık kalmaktadır.

2. Özet Kimlik Doğrulama Yöntemi (Digest):

Bu yöntemde RFC 2617 standardında mevcuttur (Franks,j, Hallem-Baker,P. , Hostetler 1999). Temel Kimlik Doğrulama Yönteminden daha güvenlidir. Bunun sebebi kullanıcı adı ve parola MD5 hash algoritması ile şifrelenir. MD5 hash algoritması tek başına güvenlik için yeterli olmadığından HTTPS protokolü ile birlikte kullanılması önerilmektedir.

3. Sertifika Tabanlı Kimlik Doğrulama Yöntemi:

Sertifika tabanlı kimlik doğrulama yöntemi yaygın olarak sunucunun istemci tarafından kimliğinin doğrulanması için kullanılmaktadır. Çok kullanıcılı uygulamalarda, tüm kullanıcılara sertifika, token, smart card vb. vermek ve bunların yönetimini, bakımını sağlamak zor olmaktadır (Demir 2015). Bu nedenle sertifika tabanlı kimlik doğrulama yöntemini iki taraflı kullanmak tercih edilmemektedir.

4. Form Tabanlı Kimlik Doğrulama Yöntemi:

Form tabanlı kimlik doğrulama karşılaşılan en yaygın doğrulama yöntemidir. Bu yöntem birkaç kimlik doğrulama faktörünün birlikte kullanılmasını mümkün kılmaktadır. Form tabanlı kimlik doğrulama yöntemi kullanıcı kimlik bilgilerinin doğrulanmasında ve yönetilmesinde kolaylık sağladığı için tercih edilmektedir.

Kimlik doğrulama, bireyin hangi görevleri yapabileceğini veya kişinin görebileceği dosyaları belirlememektedir. Kimlik doğrulama, sadece kullanıcının kim olduğunu tanımlamakta ve doğrulamakta kullanılmaktadır.

3.2.3. Kriptografi

Kriptoloji, bilgi gizliliğini ve / veya özgünlüğünü sağlamak için kullanılan tekniklerle ilgili çalışmadır (Stallings 2011). Kriptolojinin kriptografi ve kriptanaliz olmak üzere iki ana unsuru bulunmaktadır. Kriptografi, gizlilik, kimlik denetimi, bütünlük gibi bilgi güvenliği kavramlarını sağlamak için çalışan matematiksel yöntemler bütünüdür (Kriptografi - Vikipedi). Kriptanaliz ise şifrelenmiş metinlerin çözümü ve anahtarın bilinmediği durumlarda bilgiyi kurtarmak için yapılan çalışmalardır. Kriptografi çok uzun zamandır bilgiyi korumanın yollarından biri olarak görülmüştür. Kriptografi siber uzayda bulunan bilgileri korumak için sıklıkla kullanılan bir yöntemdir. Uygulamaların bünyesinde bulunan hassas bilgiler, bilinen ve test edilmiş şifreleme yöntemleri ile saklanmalıdır. Bilişim alanında görülen gelişmeler sonucunda eskiden güvenilir bulunan bazı şifreleme algoritmaları artık çok kısa sürelerde çözülebilmektedir. Bu nedenle uygulamada kullanılacak şifreleme algoritmaları verinin hassaslığına göre seçilmeli ve güncellenmelidir. Verilerinin korunması için uygun yöntemi belirlemek mimari bir karardır. Aşağıda kullanılan genel uygulamalar ve gerekli asgari anahtar uzunluğu listelenmiştir (Cryptographic Storage Cheat Sheet - OWASP 2017).

- Anahtar değişimi: En az 2048 bitlik Diffie-Hellman anahtar değişimi
- Mesaj Bütünlüğü: HMAC-SHA2
- Mesaj Hash Algoritması: SHA2 (en az 256 bit)
- Asimetrik şifreleme: RSA 2048 bit
- Simetrik anahtar algoritması: AES 128 bit
- Parola Hash Algoritmaları: PBKDF2, Scrypt, Bcrypt.

Simetrik Şifreleme Algoritmaları

Simetrik şifreleme, şifrelemenin ve şifre çözmenin aynı anahtarı kullanarak gerçekleştirildiği bir kriptoloji formudur ve geleneksel şifreleme olarak bilinmektedir (Stallings 2011). Simetrik şifreleme, gizli bir anahtar ve bir şifreleme algoritması kullanarak düz metinleri şifrelenmiş metin haline dönüştürür. Aynı anahtar ve şifre çözme algoritması kullanılarak, düz metin elde edilmektedir.

Asimetrik Şifreleme Algoritmaları

Asimetrik şifreleme, şifrelemenin ve şifre çözmenin farklı anahtarlar kullanılarak gerçekleştirildiği algoritmalardır (Stallings 2011). Haberleşen taraflardan her birinde birer çift anahtar bulunur. Bu anahtar çiftlerini oluşturan anahtarlardan biri gizli anahtar diğeri açık (gizli olmayan) anahtardır. Bu anahtarlardan bir tanesiyle şifreleme yapılırken diğeriyle de şifre çözme işlemi gerçekleştirilir. Bu iki anahtar çifti matematiksel olarak birbirleriyle bağlantılıdır (Açık anahtarlı şifreleme 2015). Asimetrik şifreleme algoritmalarında güvenli bir "ilk anahtar değişimi" ihtiyacı bulunmamaktadır.

Asimetrik şifreleme işlemleri simetrik şifreleme işlemlerine kıyasla daha yavaş çalışmaktadır. Bunun sebebi, uzun anahtarlar kullanılması ve gerçekleştirilen işlemlerin karmaşık olmasıdır. Asimetrik şifrelemelerde güvenlik tek-yönlü-fonksiyonlara dayandırılmaktadır (Açık anahtarlı şifreleme 2015). Tek yönlü çalışan fonksiyonların hesaplamaları kolaylıkla yapılabilmektedir. Ancak tersinin hesaplanmasının polinom zaman (Bir bilgisayarın bir sorunu çözmesi için gereken süre) içerisinde imkânsız olduğu kabul edilmektedir.

Anahtar Değişimi Algoritmaları

Anahtar değişim algoritmaları (SSL için Diffie-Hellman gibi) bilinmeyen bir tarafla şifreleme anahtarlarını güvenle değiş tokuş etmek için kullanılmaktadır. Bu protokol, iki kullanıcının ayrık logaritmalara dayalı bir genel anahtar düzeni kullanarak gizli bir anahtar oluşturmasını sağlar. Protokol, iki tarafın kimlik doğrulamasının gerçekleştirildiği durumlarda güvenlidir (Stallings 2011).

Hash Algoritmaları

Hashing algoritmaları verileri (bir anahtar, parola vb.) benzersiz bir karma veya sağlama toplamı üretir. Bu fonksiyon, değişken uzunlukta bir mesajı sabit uzunlukta bir karma değere veya mesaj özetine eşlemektedir (Stallings 2011). Bir başka deyişle Hash fonksiyonları, değişken uzunluklu veri kümelerini, sabit uzunluklu veri kümelerine haritalayan algoritma veya alt programlardır (Hash Fonksiyonu y.y.).

Hemen hemen tüm hash fonksiyonları, bir sıkıştırma işlevinin yinelemeli olarak kullanılmasını içermektedir. Bu algoritmalar tek yönlü olarak

çalıştıklarından, müdahale tespitinde kullanılabilmektedir. MD5 ve SHA-2 algoritmaları günümüzde yaygın olarak kullanılmaktadır.

3.2.4. Yetkilendirme

Yetkilendirme işleminin amacı izin verilen işlemlerin yetki derecelerine uygun olarak sadece yetkilendirilmiş kullanıcılar tarafından gerçekleştirilebilmesini sağlamaktır. Uygulama bünyesinde bulunan bilgilere erişimin, rol bazında yetkilendirme yapılarak sağlanması tavsiye edilmektedir. Kullanıcılara rol bazında yetki verirken işleme "Hepsini reddet" ile başlayarak gerekli oldukça rollerin ve erişim haklarının eklenmesi en iyi uygulama olacaktır. Yetkilendirme işleminin dinamik olarak yapılması ve yetki kaldırıldığında nesneye veya servislere erişimin engellenmiş olması gerekmektedir.

Yetkilendirme kapsamına aşağıdaki nesneler(veriler) alınabilir (Enstitüsü 201Ì4):

- İşletim sistemi üzerinde bulunan ve uygulama tarafından oluşturulan erişilen dosyalar, dizinler
- Dinamik içerik sağlayan sayfalar
- İstemci tarafından gönderilen POST ve GET parametreleri, HTTP başlık (header)* alanları
- Web servisleri ve web servisinin sunduğu fonksiyonlar (operations)
- Statik içerik (HTML, PDF, veritabanı yedek dosyaları v.s.)
- Bilgi varlıkları ve iş nesneleri
- Yazılım nesneleri (object, class tanımları)
- Yazılım nesne metodları (function, procedure vs.)
- Önemli fonksiyon icra eden kod parçacıkları

3.2.5. Oturum Yönetimi

Oturum bilgisi, uygulamaların istemcileri tanımak amacıyla kullandıkları bir bilgidir. Uygulamada kimlik doğrulama gerçekleştikten sonra kullanıcıya, kendisine özel olan (unique), oturum anahtarı verilir. Daha sonra istemci (tarayıcı) yapılan her istekte bu anahtarı göndererek kullanıcının kim

olduğunu sunucuya bildirir ve sunucu bu bilgiye göre gelen isteği değerlendirir. Bu bilginin taşınması sırasında çerezler, URL veya gizli form elemanları kullanılabilir (Demir 2015). Şekil 13' te çerezlerin nasıl işlediği görselleştirilmiştir.

Şekil 13: Çerez işleyiş örneği

Oturum anahtarları belirli uzunlukta olan karakter kümesidir. Oturum anahtarlarının yeterli uzunlukta olmamaları veya yeterli zorlukta olmamaları uygulamaların tahmine dayalı saldırılara karşı zayıf olmalarına sebep olmaktadır. Zayıf anahtarlara örnek olarak "Session_ID=145356344" verilebilir. Bu anahtarın zayıf olmasının sebebi kısa olması ve sadece sayılardan oluşmasıdır.

Güvenli oturum anahtarlarının sahip olması gereken özellikler aşağıdaki gibidir:

1. Oturum anahtarları sayı, büyük-küçük harf içermelidir.
2. Oturum anahtarları en az 16 karakterden oluşmalıdır. 24 veya daha fazla karakter içeren anahtarlar tercih edilmelidir.
3. Oturum anahtarları kullanıcı adı, e-posta adresi ve zaman fonksiyonları gibi tahmin edilebilir değerler içermemelidir.

4. Oturum anahtarlarına atanacak değerler rastgele değer üreten fonksiyonlarla oluşturulmalıdır.

Oturum güvenliğini sağlamak için alınması gereken bir diğer önlemde oturum bilgisinin zaman aşımına uğrayacak şekilde yapılandırılmasıdır. Böylece kullanıcı belirlenmiş bir süre sistemde işlem yapmadığında oturumun kapatılması sağlanmaktadır.

3.2.6. Konfigürasyon Dosyaları Yönetimi

Konfigürasyon dosyaları, uygulama ile ilgili hassas bilgileri içermektedir (Enstitüsü 2014). Örneğin veri tabanına erişim için kullanılan bağlantı bilgilerinin bulunduğu dosyalar konfigürasyon dosyaları kapsamına girmektedir. Konfigürasyon dosyalarında yapılacak saldırılar ile uygulamanın erişilebilirliğini engelleyebilmek veya işleyişini değiştirmek mümkün olmaktadır. Konfigürasyon dosyalarının sunucularda saklanıyor olması yeterli güvenliği sağlayamamaktadır. Konfigürasyon dosyaları hassas bilgi olarak nitelendirilmelidir. Bu dosyalar şifrelenmiş olarak saklanmalı ve gerçekleştirilen erişimler listelenerek kayıt altına alınmalıdır.

3.2.7. Hassas Bilgi

Yazılım geliştiriciler genellikle işin niteliğini tam olarak bilememekte sadece uygulama sahibinden gelen taleplere göre uygulamayı oluşturmaktadırlar. Uygulama bünyesinde bulunacak hassas bilginin belirlenmesi ve korunabilmesi için yazılım geliştiricilerin ve uygulama sahiplerinin bir arada çalışması gerekmektedir. Hassas bilgilerin ne olduğu ve hangi derecede korunması gerektiği belirlendikten sonra bu bilgileri korumak için bir politika belirlenmelidir.

3.2.8. Parametre Manipülasyonu

Dağıtık algoritmalar kullanılan uygulamalarda parametre gönderimi söz konusudur. Gönderilen parametrelere yönelik saldırılar gerçekleştirildiğinde sistemde bulunan verilerin bütünlüğü bozulabilir ve veriler değiştirilebilir. Örneğin internet alışverişlerinde kullanılan formda bulunan alışveriş tutarının http proxy kullanılarak manipüle edilmesi sonucunda borç düşük veya yüksek olacak şekilde değişiklik yapılabilir.

3.2.9. Hata Yönetimi

Uygulamalarda hata meydana geldiğinde ya da beklenmedik bir durum ortaya çıktığında, üretilen hata mesajlarında teknik detayların ve hatalara ait günlük kayıtlarının kullanıcıya gösterilmemesi gerekmektedir. Ayrıntılı olarak kullanıcıya gösterilen hata mesajları saldırgana kullanışlı bilgilerle dolu bir kaynak sunmaktadır (OWASP 2017). Kullanıcıya HTTP konum cevap kodları(404, 500 vb.) gibi tanıtımlık mesajlar gösterilmelidir. Ayrıca hataların kayıt altına alınması ve gerçek hataya sadece yazılım geliştiricilerin ulaşmasını sağlayacak bir süreç oluşturulması gerekmektedir (Özbilgin, Gökhan Özlü 2010).

3.2.10. Kayıt Tutma ve Denetim

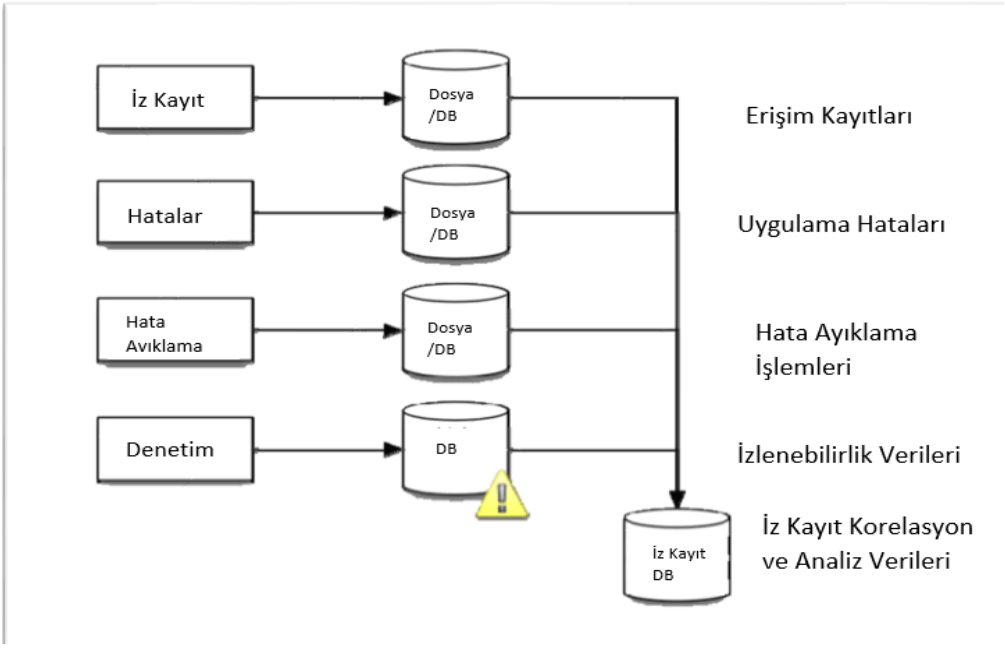
5651 sayılı kanunla tüm kurum ve kuruşlara internet trafiğinin kayıt altına alınması zorunluluğu getirilmiştir. Bu yasal zorunluluğa ek olarak uygulamalara ait erişim kayıtları, uygulama hataları, hata ayıklama işlemleri, izlenebilirlik verileri ve uygulamada yapılacak değişikliklerin üretim ortamına aktarım sürecine ait veriler de kayıt altına alınmalıdır. Bu kayıtlar, kayıt dosyalarında tutulmalı ve üzerinde silme, değişiklik gibi işlemlere izin verilmeden sadece yeni kayıtlar eklenerek güncellenmelidir. Kayıt dosyalarının düzenli olarak yedeklerinin ve kopyalarının alınması gerekmektedir. Ayrıca bu kayıtlar zaman zaman yasal işlemlerde kanıt olarak kullanılabilmektedir. Bu sebeple, kayıt verilerinin güncel olması ve bozulmamış olması güvenlik açısından önemli bir unsurdur. İz kayıtlarının, yeterli güvenlik düzeyine sahip ortamlarda korunması ve yedeklerinin alınması suretiyle, yaşanacak olası felaketler sonrasında da öngörülen süre için erişilebilir olmaları temin edilmelidir.

Ayrıca bilgi sistemleri üzerindeki riskler, sistemlerin boyutu ve faaliyetlerin karmaşıklığı göz önünde bulundurularak bilgi sistemlerinin kullanımına ilişkin etkin bir denetim izi kayıt mekanizması tesis edilmelidir. Bu sayede, bilgi sistemleri dâhilinde gerçekleşen ve Kurum faaliyetlerine ait kayıtlarda değişikliğe sebep olan işlemlere ilişkin denetim izlerinin yeterli detayda ve açıklıkta tutulması temin edilir. Denetim izlerinin bütünlüğünün bozulmasının önlenmesi ve herhangi bir bozulma durumunda bunun tespit edilebilmesi için gerekli teknikler kullanılmaktadır. Kayıt sisteminin her türlü

yetkisiz sistemsel ve kullanıcı müdahalesine karşı korunmasına yönelik önlemler alınmalıdır. Kurum faaliyetlerine ait kayıtlarda değişikliğe sebep olan işlemler için asgari olarak;

- Bu kapsamdaki işlemlere ilişkin yetkisiz erişim teşebbüslerine,
- İşlemi gerçekleştiren uygulamaya,
- İşlemi gerçekleştiren kişinin kimliğine,
- Yapılan işlemlerin zamanına, ilişkin bilgileri içeren denetim izleri tutulmalıdır.

Şekil 14: Kayıt Altına alınması gereken bilgiler



3.3. Yazılım Geliştirme Süreç ve Modelleri

Bu tez çalışmasında daha önce de belirtildiği gibi güvenlik, geliştirme sürecinin bir parçası olmalıdır. Güvenlik entegrasyonunun nasıl olacağını anlayabilmek için öncelikle yazılım geliştirme süreçlerinin ve yazılım yaşam döngülerinin anlaşılması gerekmektedir.

3.3.1. Yazılım Yaşam Döngüleri

Yazılım yaşam döngüleri, bir uygulamanın talep edilmesinden kullanımdan kaldırılmasına kadar geçen süreçleri tanımlayan çerçevelerdir. Yazılımların ortaya çıkmasından itibaren bu süreçleri yönetmek için birçok yazılım yaşam döngüsü modeli kullanılmıştır.

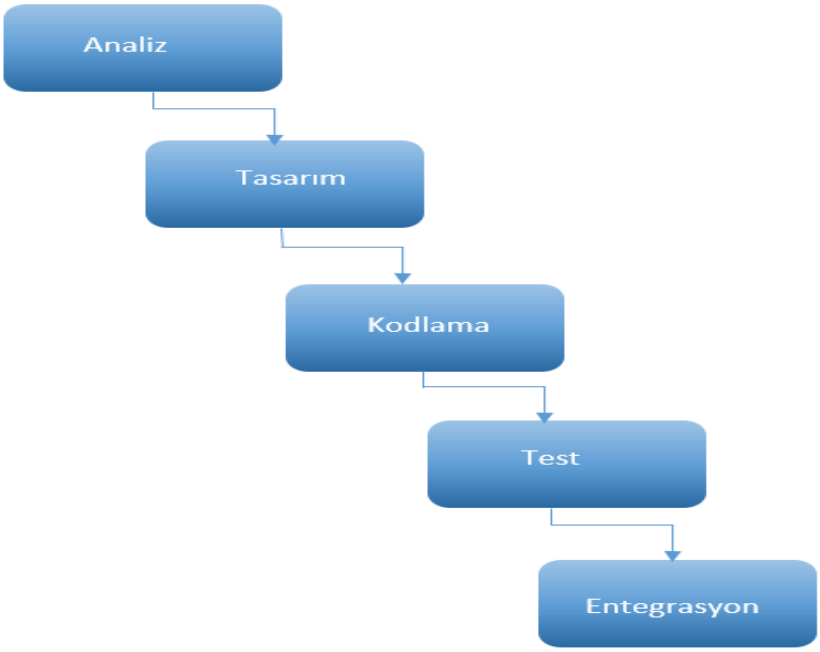
Dünyada başlıca kullanılan yazılım yaşam döngüsü modelleri;

- Şelale Modeli
- Çevik Model
- Artırımlı Model
- Spiral Model

3.3.1.1. Şelale Modeli

Şelale modeli, ilerlemenin istikrarlı bir şekilde aşağı doğru aktığı (şelale gibi) yazılım geliştirme süreçlerinde kullanılan ardışık bir tasarım sürecidir. Yazılım mühendisliğindeki diğer modellere temel teşkil eden "Şelale Modeli" yazılım yaşam döngüsünü analiz, tasarım, kodlama, test ve bakım olmak üzere beş aşamada ele almaktadır.

Şekil 15:Şelale (Waterfall) Yazılım Geliştirme Modeli



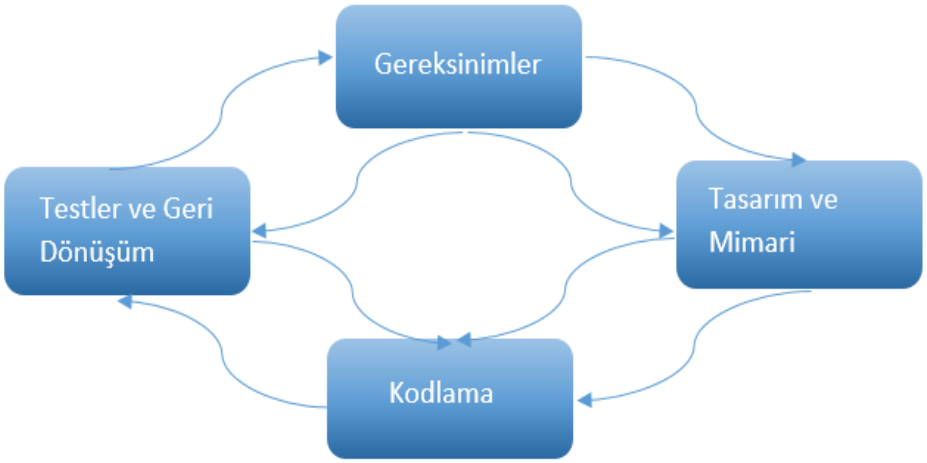
(Yazılım geliştirme süreci - Vikipedi 2016)

Şelale modelinin uzun süreli projeler için tercih edilmesi risk oluşmasına sebep olabilmektedir. Bunun sebebi uygulama gereksinimlerinin zaman içerisinde değişime uğrayabilecek olmasıdır. Proje geliştirme sürecinde gereksinimlerde oluşacak bir değişim geliştirme aşamalarının en baştan uygulanmasını gerektirir ve bu durum proje maliyetini ciddi ölçüde arttırabilmektedir. Şelale modeli koşulların iyi bilindiği, ürün tanımının net olarak yapılabildiği, belirsiz gereksinimleri olmayacak, esnek projelerde kullanılmalıdır. Projenin kalitesinin, projenin süreç ve maliyetinden daha önemli olduğu koşullarda tercih edilmelidir.

3.3.1.2. Çevik Model

Çevik yazılım geliştirme, gereksinimlerin ve çözümlerin kendi kendini düzenleyen ve işlevler arası ekipler arasındaki işbirliği sayesinde gelişen bir grup yazılım geliştirme yöntemidir. Çevik yazılım geliştirme modelinde hedef hızlı ve kaliteli bir yazılım geliştirerek teslimat yapmaktır. Projenin ilerleyen aşamalarında bile taleplerde meydana gelebilecek değişiklikler büyük zararlara yol açmayacak şekilde karşılanabilmektedir. Proje süresince yazılım geliştiriciler ile uygulama sahipleri birlikte çalışmalıdır. Bu yaklaşım; takım çalışmasıyla gelen liderlik psikolojisi, kendi kendini düzene sokma (örgütleme), sorumluluk, yüksek kalitedeki yazılımların hızlı dağıtımını onaylayan en iyi mühendislik örnekleri ve iş yaşamında müşteri ihtiyaçlarıyla şirketlerin temel amaçlarını koordine etme işlevi de görmektedir (Atik yazılım geliştirme - Vikipedi 2016).

Şekil 16: Çevik Yazılım Geliştirme Modeli



(Temur 2013)

Çevik model tekrarlanan yazılım geliştirme süreçleri sayesinde proje riskleri azaltıp, hata oranlarını düşürmeyi hedeflemektedir. Geliştirme süreçlerinde uygulama sahibinin sürekli işe dâhil edilmesiyle gereksinimlerin iyi analiz edilmesi sağlanmaktadır. Ancak dokümantasyon yönünden eksik kalmaktadır.

3.3.1.3. Spiral Model

Spiral model, yazılım projeleri için risk odaklı bir süreç modelidir. Bu model, diğer modellerden farklı olarak süreci oluşturan aşamalardan tekrar tekrar geçilmesini ve her geçişte projenin ilerleme kat etmesini hedeflemektedir (Seker 2015). Üretim süreci boyunca ortaya çıkan ara ürünün son kullanıcı tarafından sınanması temeline dayanır. Spiral Modelin Planlama, Risk Analizi, Mühendislik ve Değerlendirme olmak üzere dört aşaması bulunmaktadır.

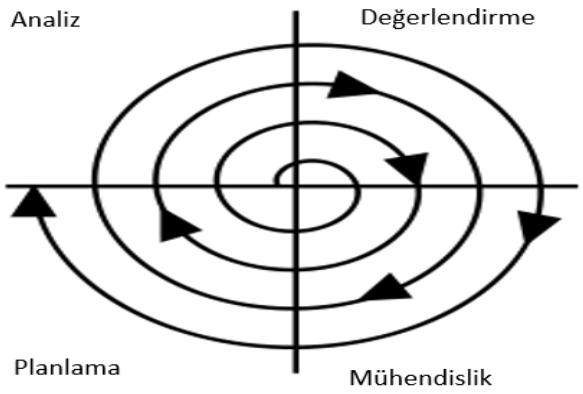
Planlama aşamasında gereksinimler ortaya çıkartılmakta ve alternatif yaklaşımlar belirlenmektedir.

Risk analizi aşamasında olası alternatif çözümler yazılım geliştirici tarafından incelenmekte ve riskler tespit edilmektedir. Bu aşamanın sonunda bir prototip üretilir.

Mühendislik aşamasında detaylı gereksinimlerin belirlenmesiyle, test ve yazılım geliştirme süreci eş zamanlı olarak yürütülmektedir.

Değerlendirme aşamasında mühendislik aşamasında oluşturulan sürüm analiz edilmekte ve bir sonraki spiral döngüye girmeden önce uygulama sahibi tarafından değerlendirilmektedir.

Şekil 17: Spiral Yazılım Geliştirme Modeli



(Software Development Spiral.svg - Wikipedia y.y.)

Spiral Geliştirme Modelinin büyük ve kritik projeler için tercih edilmesi önerilmektedir. Yazılım geliştirme sürecinde yüksek risk analizi yapıldığından diğer yazılım geliştirme modellerine göre avantaj sahibidir. Yazılımı kullanacak personelin sürece erken katılması ileride oluşabilecek istenmeyen durumları engellemektedir (YILMAZ 2007). Projenin başarılı olması risk analiz aşamasına oldukça bağlıdır. Küçük projeler için iyi çalışmaz ve bu modelin kullanımı diğer yazılım geliştirme modellerine göre daha maliyetlidir.

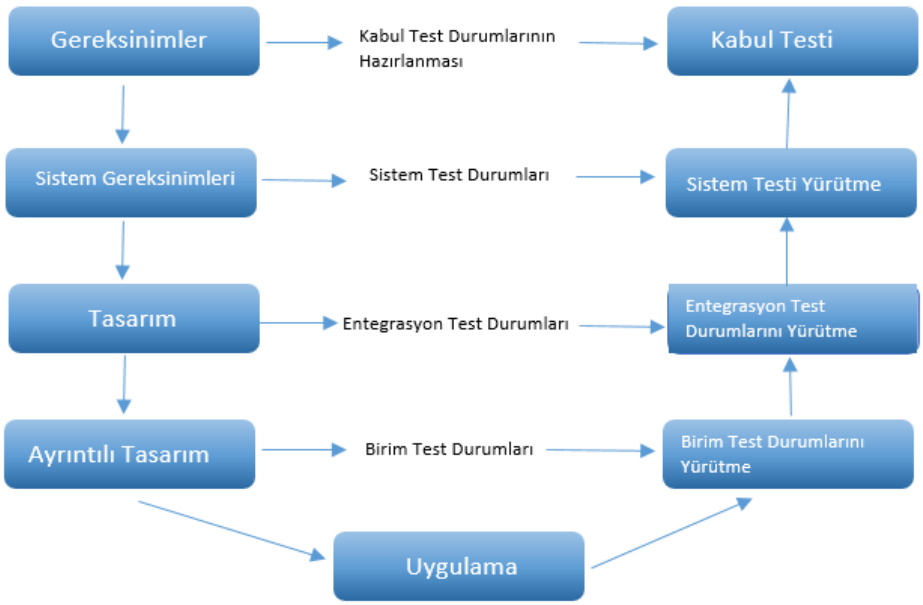
3.3.1.4. V Model

Doğrulama ve geçerleme (Verification and Validation) modeli V model olarak anılmaktadır. Şelale modelinde olduğu gibi sistem gereksinimlerinin önceden belirlenmesi gereken bir yazılım yaşam döngüsü modelidir. Şelale modelinden farkı ise yazılım geliştirmeye başlanmadan test planı oluşturulmasıdır. V modelin sol tarafı yazılım geliştirme faaliyetlerini, sağ tarafı test aşamalarını göstermektedir. Bu modelde yazılım geliştirme faaliyetlerine başlanıldığında test aşamalarına da başlanmaktadır.

Projede gereksinimlerinde bir değişiklik olduğu zaman test dokümanlarının güncellenmesi gerekmektedir. Bakım, onarım ve projeden

vazgeçme V modeline dâhil değildir. Gereksinimleri açık ve belli olan küçük, orta ve büyük boyut projelerde kullanılabilir(Temur 2013).

Şekil 18: V Model



(Temur 2013)

3.3.2. Süreç İyileştirme ve Olgunluk Modelleri

Yazılım geliştirme projeleri çoğunlukla yetersiz planlama, geliştirme süreçlerin tam anlaşılmaması, iyi bir yönetim çerçevesinin olmayışı gibi problemlerle karşı karşıya kalmakta ve başarısızlığa uğramaktadır. Bu durum bilişim teknolojilerinin nasıl daha iyi yönetileceği sorusunu ortaya çıkarmıştır. Süreçlerin iyileştirilmesi ve bilgi altyapısı üzerinde uygun seviyede yönetim ve kontrolün sağlanması için gelişme kaydetmek amacıyla olgunluk modelleri ortaya çıkmıştır (COBIT 4.1 Executive Summary). Bu modellere süreç iyileştirme ve olgunluk modelleri denilmiştir. Herhangi bir yazılım yaşam döngüsü ile bütünleştirilerek kullanılabilen bu modellerin en sık kullanılanları çalışmanın ilerleyen bölümlerinde anlatılmıştır.

3.3.2.1. Yazılım Süreç Yeterliliği ve Organizasyonel Olgunluk Standardı (SPICE ISO/IEC 15504)

Yazılım Süreç Yeterliliği ve Organizasyonel Olgunluk Standardı (SPICE) ISO tarafından yayınlanan ve sürekli güncellenen ISO 15504 yazılım

süreci iyileştirme ve yetenek belirleme standardıdır. Ülkemizde bazı özel kurumlar tarafından sertifikası alınmıştır. 1995 yılında ISO ve IEC tarafından çıkarılmıştır. SPICE da yazılım kalitesini hedefleyen standartlardan biri olup, yazılım süreçlerini iyileştirmek ve süreç yeteneklerini belirlemek için çerçeve oluşturmaktadır. Uluslararası Standartlar Örgütü ve Uluslararası Elektroteknik Komisyonu'nun ortak çalışması ile 1995 yılında standart olarak belirlenmiştir. SPICE, iki boyutlu bir model olup içe dönük süreç iyileştirme ile içe ve dışa dönük yetenek belirleme amacını taşır. Birinci boyutta süreçler, ikinci boyutta yetenek seviyeleri bulunmaktadır.

SPICE Boyutları:

- Birinci Boyut - Süreç Boyutu: Süreç bir işi yapma yöntemi olarak tanımlanmaktadır. Süreçler çoğunlukla alt süreç ve işlemlerden oluşmaktadır. İşletilen süreçler belgelenmekte ve tekrarlı olarak işletilmektedir. Bütün süreçlerin girdileri ve çıktıları vardır. SPICE standardında 5 süreç boyutu tanımlanmıştır. Bunlar:
  - Müşteri-tedarikçiye direkt etkisi olan süreçler (Customer)
  - Mühendislik süreçleri (Engineering)
  - Yönetim süreçleri (Management)
  - Destek süreçleri (Support)
  - Organizasyon süreçleri (Oganization)
- İkinci Boyut - Yetenek Seviyeleri: Her bir işlem için bir yetenek seviyesi belirlenmiştir.

Tablo 4: SPICE Seviye Adları ve Nitelikleri

Seviye	Seviye adı	Nitelikler
5	Optimizasyon Süreci	Süreç değişiminin yönetilmesi ve sürekli iyileşme
4	Öngörülebilir Süreç	Ürün ve süreç ölçümlerinin başarıyla kullanılması ve süreç denetimi

3	Kuruluş Süreci	Standart süreç tanımı, uyarlama kurallarının varlığı ve süreç kaynakları
2	Yönetilen süreç	Etkinliklerin başarı ile planlanması ve yönetilmesi
1	Yapılan işlem	Temel uygulamaların varlığı (süreci yerine getiren)
0	Eksik işlem	Nitelik Yok

ISO/IEC 15504'ün nihai hedefi, bilişim sektöründe kaliteye ulaşmaya yardımcı olacak süreçleri işletmek ve iyileştirme sağlamaktır. Bu standardın kullanım alanları süreç iyileştirme ve yetenek belirleme olarak belirtilmiştir (ISO 15504 (SPICE) 2014). SPICE İlkeleri aşağıda verilmiştir;

- Standartlaşma
- Değerlendirme, yetenek belirleme ve iyileştirme
- Diğer modellere uyum sağlama
- Gelişmeyi ölçme
- Nesnel, tutarlı ve tekrarlanabilir olma
- Sertifikasyon amacı taşımaz

3.3.2.2. Tümleşik Yetenek Olgunluk Modeli (CMMI)

Tümleşik Yetenek Olgunluk Modeli (CMMI: Capability Maturity Model Integration) bir süreç modeli olup, örgütlerin yazılım süreçlerinin (Yazılım planlama, geliştirme, yapılandırma vb.) olgunluğunu değerlendirme modelidir (CMMI - Vikipedi). CMMI, Carnegie Mellon Üniversitesi'ne bağlı Yazılım Mühendisliği Enstitüsü tarafından Amerikan Savunma Bakanlığı'nın isteği üzerine 1986 yılında geliştirilmeye başlanmıştır. Kuruluşların % 70'inden fazlası yetkinlik boşluklarını, karşılaştıkları en büyük beş zorluktan biri olarak göstermektedir (Get Started | CMMI Institute 2017). Bu modelin uzun süreli kullanımında iş performansının olgunluğunun artmasına yardımcı olması hedeflenmektedir.

CMMI modelinde 5 olgunluk seviyesi ve 6 yeterlilik düzeyi bulunmaktadır. Bunlar Tablo 5 ve Tablo 6 de gösterilmiştir.

Tablo 5: CMMI Olgunluk Seviyeleri

Seviye	Ad
1	Başlangıç
2	Yönetilen
3	Tanımlı
4	Nicel olarak yönetilen
5	İyileştirici

Tablo 6: CMMI Yeterlilik Düzeyleri

Düzey	Ad
0	Yetersiz
1	İfa edilen
2	Yönetilen
3	Tanımlı
4	Nicel olarak yönetilen
5	İyileştirici

CMMI modeli 22 işlem alanından ve tüm kuruluşların izlemesi beklenen üç genel hedeften oluşmaktadır (CMMI İlkeleri ve Değerleri 2013).

Genel Hedefler:

- İşlem, tanımlanabilir giriş iş ürünlerini tanımlanabilir çıkış iş ürünlerine dönüştürerek süreç alanının belirli amaçlarının başarılmasını destekler ve sağlar.
- İşlem yönetilen bir işlem olarak kurumsallaştırılmıştır.
- İşlem, tanımlanmış bir işlem olarak kurumsallaştırılmıştır.

Tümleşik Yetenek Olgunluk Modelinde süreç iyileştirme ve değerlendirme; Proje Yönetimi, İşletme Yönetimi, Mühendislik ve Destek olmak üzere dört kategoride incelenir. CMMI süreçleri içerisinde doğrudan güvenlikle ilgili bir süreç bulunmamaktadır (Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan 2009).

3.3.2.3. Tümleşik Test Olgunluk Modeli (TMMI)

Tümleşik Test Olgunluk Modeli (TMMI- Test Maturity Model Integration) , Tümleşik Yetenek Olgunluk Modeli (CMMI) temel alınarak

geliştirilen bir modeldir. TMMI'ın amacı test işlemlerinin olgunluğunu belirlemek için bir sistem yaratmak ve testin olgunluğunu geliştirmek için hedefler belirlemektir.

CMMI'da test aşamalarına yeteri kadar önem verilmediği düşünülerek bu modelde gelişme yoluna gidilmiştir ve böylece ortaya TMMI çıkmıştır. TMMI için CMMI'ın daha detaylı ve geliştirilmiş bir versiyonu demek mümkündür.

Tablo 7: TMMI Olgunluk Seviyeleri

Seviye	Ad	Açıklama
1	Başlangıç	Test için plansız metotlar kullanılmaktadır, kalite standardı yoktur.
2	Yönetilmiş	Test yönetimli bir sürece girer ve hata ayıklama işinden ayrılır, test stratejileri ve test planları belirlenir
3	Tanımlanmış	Testler Geliştirme döngüsüne tamamen entegre, test planı projenin başında yapılır.
4	Ölçülmüş	Test aktiviteleri döngünün bütün aşamalarında vardır, test kalitesini ölçmek için bir test ölçüm programı kullanılır.
5	İyileştirme	Test sürecinin performansını geliştirmek için çalışmalar yürütülür.

3.3.3. TS ISO/IEC 12207 :2010 Yazılım Yaşam Döngüsü Süreçleri

Bu standart; yazılım endüstrisi tarafından kaynak gösterilebilecek, iyi tanımlanmış terminolojisiyle, yazılım yaşam döngüsü süreçleri için ortak bir çerçeve oluşturmaktadır. Bu standart; yazılım, bağımsız yazılım ürünü ve yazılım hizmeti içeren bir sistemin satın alınması ve yazılım ürünlerinin tedarik, geliştirme, işletme ve bakımı süresince uygulanacak süreçleri, faaliyetleri ve görevleri kapsamaktadır (TSE 2007). Ayrıca yazılım yaşam döngüsü süreçlerinin tanımlanması, kontrol edilmesi ve geliştirilmesi için kullanılabilecek süreçleri de sağlamaktadır. En son 2013 yılında gözden geçirilmiş ve onaylanmıştır.

Bu standart, yazılım yaşam döngüsü boyunca uygulanabilecek faaliyetleri beş ana süreç, sekiz destek süreci ve dört kurumsal süreç olarak gruplandırmaktadır.

Tablo 8: Yazılım Yaşam Döngüsü Süreçleri

Süreç	Açıklama
Edinme	Yazılım ürünü veya yazılım hizmetini edinen kurumun faaliyetlerini tanımlar
Tedarik	Yazılım ürünü veya yazılım hizmetini sağlayan satıcı kurumun faaliyetlerini tanımlar
Geliştirme	Geliştirici kurumun faaliyetlerini tanımlar
İşletim	Kullanıcılar için gerçek ortamda yazılımın işletilmesini sağlayan kurumun faaliyetlerini tanımlar
Bakım	Yazılım ürününe bakım hizmeti (ki, bu yazılım ürününü her an hazır ve çalışır durumda tutmaktır), sağlayan kuruluşun faaliyetlerini tanımlar (Bu süreç yazılım ürününün kullanımdan kalkmasını ve taşınmasını da içermektedir.)

(TSE 2007)

Bir destek süreci, farklı amaçlı tümleşik bir bölüm olarak diğer bir süreci desteklemekte ve yazılım projesinin başarı ve kalitesine katkıda bulunmaktadır (TSE 2007). Bir destek süreci başka bir süreç tarafından, gerektiğinde işletilmektedir. TS ISO/IEC 12207 standardında belirlenmiş olan destek süreçleri aşağıda verilmiştir;

Tablo 9: Yazılım Yaşam Döngüsü Destek Süreçleri

Süreç	Açıklama
Dokümantasyon	Yaşam döngüsü süreci tarafından üretilen bilginin kaydedilmesi faaliyetlerini tanımlar
Konfigürasyon Yönetimi	Yapılandırma yönetimi faaliyetleri tanımlar
Kalite Güvence	Yazılım ürünleri ve süreçlerinin belirtilen ihtiyaçlara uygun olduğunu ve yapılan plânlara bağlı kalındığını tarafsız bir şekilde garantileyen faaliyetleri tanımlar
Doğrulama	Yazılım projesine bağlı olarak değişen büyüklüklerdeki yazılım ürünlerinin doğrulanması faaliyetlerini tanımlar.

Geçerleme	Yazılım projesi yazılım ürünlerinin geçerleme faaliyetlerini tanımlar.
Müşterek Gözden Geçirme	Bir faaliyetin sonucundaki ürünleri ve durumu değerlendirme faaliyetlerini tanımlar.
Denetleme	İhtiyaçlar, plânlar ve sözleşme ile uyumu belirleyen faaliyetleri kapsar.
Problem Çözümleme	Geliştirme, işletme, bakım veya diğer süreçlerin icrası boyunca karşılaşılan, kaynağı veya yapısı nasıl olursa olsun, problemleri (uyumsuzluk dâhil) analiz etme ve ortadan kaldırma sürecini tanımlar.

(TSE 2007)

Kurumsal yaşam döngüsü süreçleri, ilişkilendirilmiş yaşam döngüsü süreçleri ve personelin oluşturduğu temel yapıyı kurmak, gerçekleştirmek, yapıyı ve süreçleri devamlı bir şekilde iyileştirmek için kurum tarafından kullanılır (TSE 2007). Süreçler aşağıda verildiği gibidir;

Tablo-10: Kurumsal Yaşam Döngüsü Süreçleri

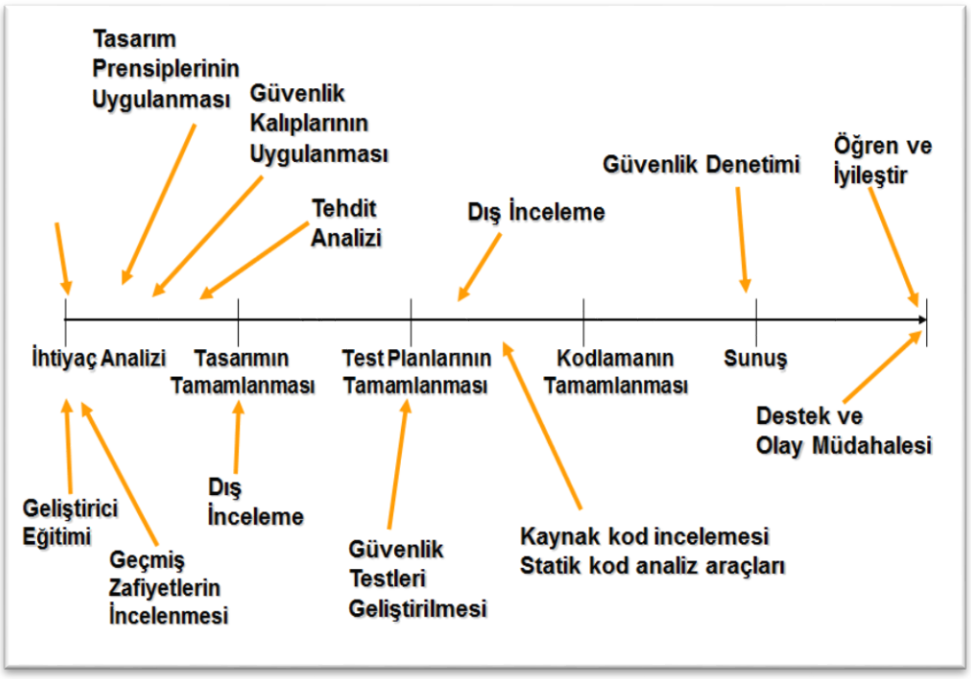
Süreç	Açıklama
Yönetim	Bir yaşam döngüsü süreci boyunca proje yönetimi de dâhil yönetimin temel faaliyetlerini tanımlar
Altyapı	Bir yaşam döngüsü sürecinin temelini oluşturan yapının kurulmasındaki temel faaliyetleri tanımlar.
İlerleme	Bir kurumun yaşam döngüsü sürecini kurma, ölçme, kontrol etme ve iyileştirme temel faaliyetlerini tanımlar.

(TSE 2007)

3.4. Güvenli Yazılım Geliştirme Modelleri

İnternetin ve teknolojinin gelişmesinin doğal bir sonucu olarak siber uzayda bulunan bilgi varlıkları ve bunlara yönelik tehditler karmaşıklaşarak çoğalmıştır. Bu durum güvenlik konusunun ayrı bir disiplin haline gelmesini sağlamıştır. Bu tez çalışmasının 3.3.2 Süreç İyileştirme Ve Olgunluk Modelleri bölümünde anlatılmış olan çalışmalarda güvenlik çoğunlukla ayrı bir başlık olarak ele alınmıştır. Ancak yazılım geliştirmede başvurulan bu yöntemler özellikle güvenlik odaklı olmadıklarından "Güvenli Uygulama Yazılımı Geliştirme" konusunun nasıl ele alınması gerektiği hakkında yol göstericilik yapamamaktadırlar. Güvenlik açıklarını erken bir safhada keşfedebilmek ve bu açıkları azaltabilmek için yazılım yaşam döngüleri ile güvenlik faaliyetlerini bütünleştirmek gerekmektedir. Yazılım geliştirme aşamalarında gerçekleştirilebilecek güvenlik eylemleri Şekil 19'da gösterilmiştir.

Şekil 19: Yazılım geliştirme aşamalarında ve güvenlik işlemleri



(GÜRLER 2007)

Zafiyet, Tehdit ve Risk modellemesinin yazılım geliştirme süreçlerine entegrasyonu ile yazılımın üretim aşamasına geçtiği zaman oluşabilecek güvenlik zafiyetlerinin proaktif olarak önlenmesi Güvenli Yazılım Geliştirme Modellerinin ana fikrini oluşturmaktadır.

Yazılım geliştirme aşamalarında güvenlik eylemlerinin sistematik olarak gerçekleştirilebilmesi için güvenli yazılım geliştirme modelleri konusunda çalışmalar yapılmış ve standartlar, çerçeve yapılar oluşturulmuştur. Bu modeller mevcut yazılım süreçlerini değiştirmeyi gerektirmez, sadece bir dizi yüksek etkili güvenlik etkinliği ekleyerek gelişmelerini sağlamaktadırlar. Bunların başlıcaları ilerleyen bölümlerde anlatılmıştır.

3.4.1. Yazılım Güvencesi Olgunluk Modeli (SAMM)

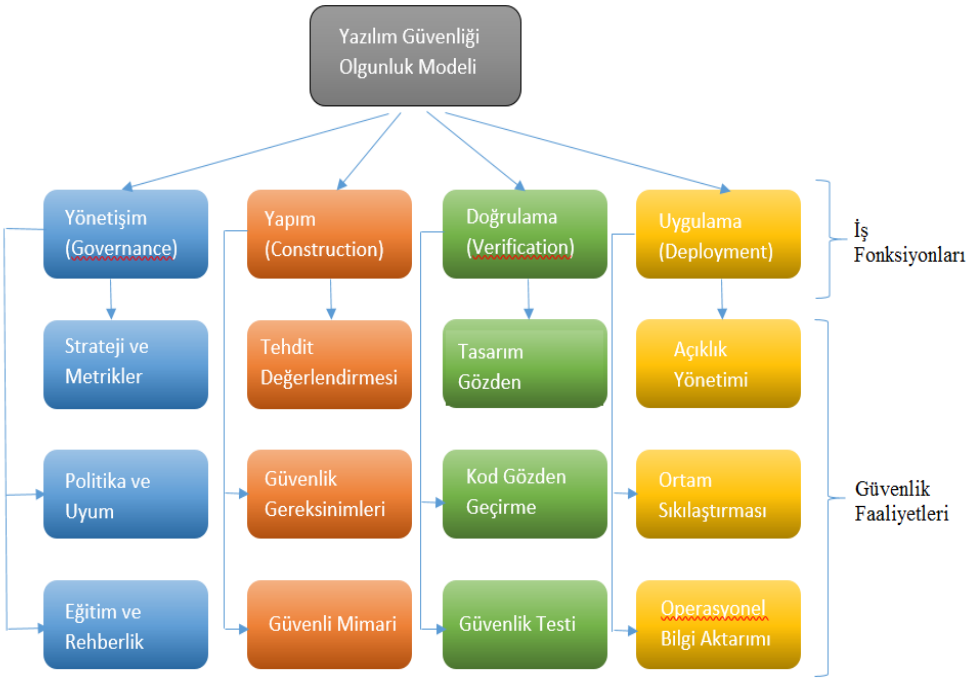
Açık Web Uygulaması Güvenlik Projesi (OWASP), kurumların güvenli uygulamalar tasarlamasını, geliştirmesini, elde etmesini, kullanmasını ve bakımını sağlamaya adanmış kar amacı gütmeyen bir örgütlenmedir. OWASP tarafından desteklenen OWASP Yazılım Güvencesi Olgunluk Modeli (SAMM) projesi kapsamında yazılım garanti olgunluk modeli ortaya

konmuştur (Deleersnyder vd. 2009). Açık kaynaklı bu çalışmada güvenli yazılım geliştirme amacıyla bir çerçeve oluşturulması hedeflenmiştir. Çerçeve, büyüklükten bağımsız bir şekilde her organizasyonun kendine adapte edebileceği, normal yazılım geliştirme döngülerine uyarlayabileceği ve bir organizasyondaki yazılım güvenliği alanında gelişmeyi yönlendirebilecek bir yapıda oluşturulmuştur (Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan 2009). Çerçevede iş fonksiyonlarını içeren dört ana başlık bulunmaktadır;

- Yönetişim (governance) Fonksiyonu: Uygulanacak yazılım güvenliği programının stratejik yönünün belirlenmesi, kurumsal güvenlik çalışmalarının performansını ölçme yöntemlerinin belirlenmesi, belirlenmiş kurumsal standartlara ve diğer düzenlemelere uyum sağlanması, personelin yazılım güvenliği konusunda eğitilmesi ve personele rehberlik yapılması konuları bu başlık altında toplanmıştır.
- Yapım (construction) Fonksiyonu: Güvenli yazılım geliştirilebilmek için gereksinimlerin belirlenmesi, tasarım aşamasında bulunması gereken güvenlik eylemler anlatılmaktadır. Tehditlerin değerlendirilmesi, güvenlik ihtiyaçlarının belirlenmesi ve güvenli mimarinin oluşturulması hedeflenen güvenlik eylemleridir.
- Doğrulama (verification) Fonksiyonu: Tasarım, kodlama ve test aşamalarında gerçekleştirilmesi gereken güvenlik gözden geçirmelerini ve güvenlik testlerini kapsamaktadır.
- Uygulama (deployment) Fonksiyonu: Yazılımın canlı sisteme kurulması ve destek verilmesi aşamalarında gerçekleştirilmesi gereken eylemleri içermektedir. Bahsedilen güvenlik eylemleri; açıklık yönetimi, ortam sıkılaştırması ve operasyonel bilgi aktarımı olarak ele alınmıştır.

Bu ana başlıklar aslında normal yazılım geliştirme döngüsünün temel adımlarına karşılık gelmektedir. Her bir ana başlık altında güvenlik eylemleri yer almaktadır. Bu güvenlik eylemleri Şekil 20' de gösterilmiştir.

Şekil 20: SAMM İş Fonksiyonları ve Güvenlik Eylemleri



(Deleersnyder vd. 2009)

Bu yapıda, güvenli bir yazılım geliştirmek için her bir temel yazılım geliştirme adımına karşılık yapılması gereken güvenlik çalışmaları güvenlik faaliyeti olarak değerlendirilmiştir. Modelde hedeflenen on iki güvenlik faaliyetinin her biri olgunluk alanıdır. Bu olgunluk modelleri için tanımlanan ardışık hedefler, güvenlik için gerekli yapı taşlarını oluşturmaktadır. Hedefleri gerçekleştirdikçe iyileştirilecek uygulamalar seçilmeli ve başarı ölçütlerinde ilgili faaliyetleri gerçekleştirerek seviyeyi yükseltmek iyi bir uygulama olacaktır. Modelde yazılım güvenliğini sağlamak için gerçekleştirilmesi gereken faaliyetlerden aşağıda verilmiştir (Deleersnyder vd. 2009):

YÖNETİŞİM

- *Strateji ve Metrikler*, bir yazılım güvenlik garantisi programı için bir organizasyon çerçevesinin oluşturulması üzerine odaklanmıştır. Bu, güvenlik amaçlarını hem ölçülebilir hem de örgütün gerçek kurumsal riskiyle uyumlu şekilde tanımlamadaki en temel adımdır. Hafif risk profilleri ile başlayarak, bir organizasyon, zaman içinde uygulama ve veri varlıkları için daha gelişmiş risk sınıflandırma şemalarına dönüşür. Göreceli risk önlemlerine ilişkin ilave

bilgilerle bir organizasyon, proje düzeyindeki güvenlik hedeflerini ayarlayabilir ve güvenlik programını daha verimli hale getirmek için ayrıntılı yol haritaları geliştirebilir.

- *Politika ve Uyum*, dış yasal ve düzenleyici gereklilikleri anlamak ve bunlarla buluşmak ve aynı zamanda kuruluşun kurumsal amacı ile uyumluluğu sağlamak için iç güvenlik standartlarını belirlemek üzerine odaklanmıştır. Bu Uygulamada iyileştirme için sürükleyici bir tema, bilgi toplayan proje düzeyinde denetime odaklanmaktadır. Beklentilerin karşılandığını kontrol etmek için örgütün davranışları hakkında bilgi verir.
- *Eğitim ve rehberlik:* Yazılım geliştirme sürecinin tüm aşamalarında yer alan doğru personel için doğru güvenlik eğitimine odaklanmayı içermektedir. Teknik personel için yazılım geliştirme alanlarında bilgi güvenliği ile ilgili sertifikalar önerilmektedir.

YAPIM

- *Güvenlik gereksinimleri*, Yazılımda yerleşik bilgi güvenliği planlamak için, diğer işlevsel veya işlevsel olmayan gereksinimlerle aynı şekilde geliştirilebilecekleri ve test edilebilecekleri gereklilikler ayrıntılı olarak belirlenmelidir. SAMM, tüm yazılım geliştirme projelerinin aynı olmadığını kabul etmektedir. Güvenlik gereksinimleri, geliştirilen yazılım türü, işlenecek veya erişimi olacak olan veriler gibi çeşitli risk faktörlerine göre uyarlanmalıdır.
- *Tehdit Değerlendirme,* geliştirilmekte olan yazılımın işlevselliğine ve çalışma zamanı ortamının özelliklerine dayalı olarak proje düzeyindeki riskleri tanımlama ve anlama üzerine odaklanmaktadır. Her projeye yönelik tehditler ve muhtemel saldırılar modellenerek güvenlik konusunda daha iyi kararlar almayı kolaylaştırmaktadır. Basit tehdit modelleri ile başlayıp daha ayrıntılı tehdit analizi ve ağırlıklandırma yöntemleri oluşturarak bir organizasyonun gelişmesi sağlanmaktadır.
- *Güvenli Mimari,* bir kuruluşun varsayılan olarak güvenli bir yazılım tasarlayıp kurması için proaktif adımlar üzerinde durmaktadır. Yeniden kullanılabilir hizmetler ve bileşenlerle yazılım tasarım

sürecini geliştirerek, yazılım geliştirmeden kaynaklanan genel güvenlik riski önemli ölçüde azaltılabilmektedir. Yazılım çerçeveleri ve güvenli tasarım ilkelerinin açık bir şekilde ele alınmasıyla ilgili basit tavsiyelerden başlayarak, güvenlik işlevleri için tasarım kalıplarını tutarlı bir şekilde kullanmaya doğru ilerlemektedir. Ayrıca, faaliyetler, proje ekiplerini merkezi güvenlik hizmetleri ve altyapı kullanımını artırmaya teşvik eder.

DOĞRULAMA

- *Tasarım Gözden Geçirme,* güvenlikle ilgili sorunlar için yazılım tasarımı ve mimarinin değerlendirilmesine odaklanmıştır. Bu, bir organizasyonun yazılım geliştirmede mimari düzeydeki sorunları tespit etmesini ve dolayısıyla güvenlik endişeleri nedeniyle daha sonra büyük zararlara uğramasını engellemektedir. Bir mimari ile ilgili güvenlikle ilgili ayrıntıları anlamak için hafif faaliyetlerle başlayarak, bir organizasyon güvenlik mekanizmalarının sağlanmasında eksiksizliği doğrulayan daha resmi denetleme yöntemlerine doğru ilerlemektedir.
- *Kod İncelemesi,* güvenlik açıklarını bulmak için yazılımın kaynak kodu düzeyinde incelenmesine odaklanmıştır. Kod düzeyindeki güvenlik açıkları genel olarak kavramsal olarak anlaşılması kolaydır, ancak bilinçli geliştiriciler bile yazılımda hata yapabilmektedir. Başlamak için, bir organizasyon hafif denetim listeleri kullanır ve verimlilik açısından yalnızca en kritik yazılım modüllerini inceler. Bununla birlikte, bir organizasyon geliştikçe, kod inceleme faaliyetlerinin kapsama alanını ve etkinliğini önemli ölçüde artırmak için otomasyon teknolojisini kullanır.
- *Güvenlik testleri,* etkinliği, uzun yıllardır gerçekleştirilmektedir. Buna, kara kutu ve beyaz kutu testi gibi geleneksel penetrasyon testleri dâhildir. SAMM ayrıca, güvenlik gereksinimlerinden türetilen test durumlarına ve otomatikleştirilmiş araçları kullanan bilgi güvenliği ile ilgili konular için kaynak kod

analizine dayanan daha özelleştirilmiş testler yapmayı önermektedir.

UYGULAMA

- *Güvenlik Açığı Yönetimi*, güvenlik açığı raporlarına ve operasyonel olaylara karşı bir organizasyon içindeki süreçlere odaklanmaktadır. Gelişmiş bir formda, güvenlik açığı yönetimi, olayların detaylı bir şekilde incelenmesini ve kurumun aşağı akım davranışına geribildirim için ayrıntılı metrikler ve diğer kök neden bilgilerini toplamak için güvenlik açığı raporları içermektedir.
- *Ortam Sıkılaştırılması*, kuruluşun yazılımını barındıran çalışma ortamı için güvence oluşturmak üzerine yoğunlaşmıştır. Harici bileşenlerdeki sorunlar nedeniyle bir uygulamanın güvenli bir şekilde çalışması bozulabileceğinden, alttaki altyapıyı sıkılaştırmak yazılımın genel güvenlik durumunu doğrudan iyileştirir.
- *Operasyonel Bilgi Aktarımı,* proje ekiplerinden kritik önem taşıyan bilgilerin toplanması ve yazılımın kullanıcılarına iletilmesine odaklanmıştır. Bu bilgi olmadan, en güvenli şekilde tasarlanmış yazılımlar bile risk taşımaktadır. Kullanıcılar ve operatörler için en etkili bilgileri toplayabilmek için hafif dokümantasyondan başlayarak, bir kuruluş her bir sürümle birlikte verilen eksiksiz operasyonel güvenlik kılavuzlarını oluşturma yönünde gelişmelidir.

Her güvenlik uygulamasının altında zamanla gelişmeyi sağlayacak hedefler tanımlanmıştır. Bu hedeflerin gerçekleştirilmesi o uygulamadaki seviyeyi belirler. Güvenlik uygulamaları için biri örtülü olmak üzere dört seviye belirlenmiştir;

Tablo 10: SAMM Olgunluk Seviyeleri

Seviye	Açıklama
0	Uygulamanın tamamlanmadığı örtülü başlangıç noktası
1	Güvenlik Uygulamasının başlangıçtaki anlayışı ve geçici tedariki

2	Güvenlik Uygulamasının verimliliğinin ve/veya etkinliğinin arttırılması
3	Ölçeklenebilir Güvenlik Uygulamasının kapsamlı uzmanlığı

Tablo 9 da gösterilmiş olan her bir seviye için Amaç, Etkinlikler, Sonuçlar, Başarı Metrikleri, Maliyetler, Personel, İlgili Düzeyler modelde tanımlanmıştır.

SAMM Modelini, yazılım geliştirme süreçleri ile bütünleştirebilmek için aşağıda verilen adımlar uygulanmalıdır.

- Güvenlik uygulamalarının her biri için modelde bulunan kılavuzlara göre değerlendirme yapılmalıdır. Yapılan değerlendirme sonuçları başarı ölçütlerine göre kontrol edilmelidir.
- Fark analizi yapılmalıdır.
- Karşılaşılan eksiklikleri karşılamak için başarı ölçütleri göz önünde bulunarak yol haritası belirlenmelidir.

SAMM genele yönelik olarak oluşturulmuş bir modeldir ve her ölçekteki kuruluş için kullanıma uygundur. SAMM çerçevesi hızlı bir şekilde kurulabilmektedir. Uygulama güvenliğini artırmak için uygulanabilir öneriler sunmaktadır ve risk yönetimi açısından sadedir. SAMM projesi açık kaynak olmasından dolayı dinamik bir yapıya sahiptir ve sürekli gelişme göstermektedir. Anlaşılması kolay bir yapıda olması sebebiyle yazılım geliştiriciler arasında kullanımı tavsiye edilmektedir.

3.4.2. Microsoft Güvenlik Geliştirme Yaşam Döngüsü (Microsoft SDL)

Güvenlik Geliştirme Yaşam Döngüsü (SDL), geliştiricilerin daha güvenli bir yazılım oluşturmasına ve güvenlik uyumluluk gereksinimlerini karşılamasına ve geliştirme maliyetini düşürmesine yardımcı olan bir yazılım geliştirme sürecidir (Howard ve Lipner 2006).

Microsoft, 2002'de güvenilir uygulamalar sunmaya olan bağlılığının bir sonucu olarak, Güvenlik Geliştirme Yaşam Döngüsü ’nü ürünlerinde sıkça karşılaştıkları güvenlik sorunlarına çözüm getirmek için tanımlamıştır. SDL, Microsoft'un geliştirme sürecini tamamlayan ve özellikle güvenlik konularını hedefleyen bir dizi etkinlik içermektedir. Bu modelde proje başlangıcından

itibaren güvenli yazlım geliştirme hedeflenmektedir. Bu hedefe ulaşmak için 13 adım belirlenmiştir (Howard ve Lipner 2006);

- Eğitim ve farkındalık,
- Proje başlangıcı,
- En iyi pratikleri tanımla ve uygula,
- Risk analizi yap,
- Risk analizi aracı,
- Risk analizi,
- Yazılım dokümantasyonu araçları ve uygulama sahibi için en iyi uygulamalar,
- Güvenli kodlama politikası,
- Güvenli test politikası,
- Güvenlik ekleme,
- Son güvenlik kontrolü,
- Güvenlik müdahale planlaması,
- Ürün çıkarma güvenlik cevapları ve işletme.

SDL optimizasyon modelinin dört güvenlik olgunluk seviyesi bulunmaktadır ve bu seviyeler Tablo 11' de gösterilmiştir.

Tablo 11: Microsoft SDL Olgunluk Seviyeleri

Seviye	Açıklama
Temel	Güvenlik reaktiftir ve riskler tanımlanmamıştır.
Standart	Güvenlik proaktiftir ve riskler anlaşılmıştır.
İleri	Güvenlik entegre olmuştur ve riskler kontrol altına alınmıştır.
Dinamik	Güvenlik konusunda uzmanlaşılmıştır ve riskler en aza indirilmiştir.

(Howard ve Lipner 2006)

Şekil 21: Güvenlik için Genel Yaklaşım

(Secure SDLC y.y.)

Microsoft SDL'in temel güvenli tasarımı altı temel ilkeye dayanmaktadır. Bu ilkeler;

- Saldırı Yüzeyinin Azaltılması: Saldırı yüzeyi, uygulamanın bir kişi veya başka bir program tarafından erişilebilen herhangi bir parçası olarak tanımlanmaktadır. Bu parçaların her biri potansiyel olarak saldırganlar tarafından kullanılabilmektedir. Saldırganların keşfedebileceği veya yararlanmaya çalışabileceği saldırı yüzey noktalarının sayısını en aza indirilmelidir. Örneğin varsayılan olarak çalışan kodların azaltılması, erişim denetiminin sağlanması, girdi denetimi yapılması vb.
- Temel Gizlilik: Gizlilik bilgi güvenliğinin temel prensiplerinden birisidir. Kullanıcıları bilgilerin kullanımını, toplanmasını ve dağıtımı konusunda bilinçlendirilmelidir. (Kişisel bilgiler, İsimsiz bilgiler,)
- Tehdit Modelleme: Tehdit modelleme, bir uygulamaya yönelik tehditleri ve uygulamada bulunan zafiyetleri anlama sürecidir. Microsoft, tehdit modelleme aracını yayınlamış ve ürün ve hizmetlere yönelik tehditleri değerlendirmek için dâhili olarak kullanmaktadır.
- Derin Savunma: "Bir savunma katmanı ihlal edildiğinde diğer savunma katmanları (varsa) uygulamayı koruyabiliyor mu?" sorusunun cevabıdır. Örneğin uygulamaları korumak için kullanılan

ateş duvarı katmanı saldırganlar tarafından aşıldığında uygulama güvenliğinin sağlanıp sağlanmadığı bilgisidir.

- En Az Ayrıcalık: Bir uygulama saldırıya uğradığı zaman uygulamanın maruz kalabileceği olası zararlar belirlenmeli ve buna göre en aza indirilmelidir. Örnek olarak kullanıcı hesaplarının ele geçirilmesi verilebilir. Yetkili bir yönetici hesabının ele geçirilmesinin sisteme vereceği zarar yetkisiz son kullanıcı hesabının ele geçirilmesi durumunda oluşacak zarardan çok daha fazladır.
- Güvenli Varsayılanlar: Uygulamalarda atanan varsayılanların daha güvenli yapılandırılması gerekmektedir. Güvenlik ve gizlilik düzeylerini azaltmak kullanıcının sorumluluğundadır. Örneğin, varsayılan olarak parola karmaşıklığı yüksek olmalı ve hash algoritmaları çalıştırıldıktan sonra kaydedilmelidir.

SDL üç temel konsepte dayanmaktadır;

1. Eğitim,
2. Sürekli süreç iyileştirme
3. Hesap verebilirlik.

Microsoft SDL, zorunlu güvenlik aktivitelerinden oluşan bir derlemedir. Temel bir konsept olarak, her aşamada üretilen çıktıların kalitesine ve bütünlüğüne odaklanmaktadır (Tiirik 2004). Dolayısıyla çıktı üretilen yol değil ürünün kendisi önem taşımaktadır. Bir geliştirme ekibi, Microsoft SDL sürecine uymak için zorunlu güvenlik faaliyetini başarıyla tamamlamalıdır.

Bu modelde güvenlik bir kalite unsuru olarak ele alınmıştır. Benimsenmesi halinde projelerde iyi tanımlanmış süreçler belirlenebilmektedir. Kendi içinde tutarlı ve net bir metodoloji olan bu modeli uygulamaya geçirmek çoğunlukla kolaydır. Microsoft'un dokümantasyonu içinde pratik uyarı ve etkili uygulama yöntemleri tanımlanmıştır. Tehdit ağaç yapılarının bulunması, kurumsal rollerin net ve detaylı olarak açıklanmış olması modelin diğer olumlu yanları arasındadır. Olumlu yönlerinden sonuncusu ise başarısı kanıtlanmış bir model olmasıdır. Microsoft'un verilerine göre 36 aylık kullanımdan sonra uygulamalarında bulunan güvenlik açıklarında %91'e varan azalma görülmüştür.

Microsoft Güvenlik Geliştirme Yaşam Döngüsünün temel dezavantajı doğrudan proje yöneticisinin liderliğine bağlı olmasıdır (Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan 2009). Bu modelin genel ve her kuruma yönelik bir metot değil, doğrudan Microsoft'un kendi ihtiyaçlarına yönelik olarak geliştirilmiş olması da olumsuz yönleri arasında sayılabilir. Microsoft için işe yaradığı kanıtlanmış ve iyi belgelendirilmiş bir yöntem olmasına rağmen her kuruma uymayabileceği düşünülmektedir. Microsoft SDL modeli kullanımı ağır bir modeldir, büyük yazılım sağlayıcılar için kullanımında iyi sonuçlar alınması beklenmektedir.

3.4.3. Yazılım Güvenliği Temas Noktaları (Touchpoints)

Yazılım Güvenliği Temas Noktaları Modeli (Software Security Touchpoints), en iyi uygulamalardan oluşan bir küme setidir ve 2006 yılında Gary McGraw tarafından geliştirilmiştir. Ulusal Siber Güvenlik Görev Gücü raporu sonucunda ABD İç Güvenlik Departmanı tarafından benimsenmiş ve kullanıma geçirilmiştir.

McGraw'a göre, yazılım güvenliğinin üç temel direği, risk yönetimi, yazılım güvenlik temas noktaları ve bilgi/tecrübedir (Gary McGraw 2006). Üç sütunun kademeli, evrimsel bir şekilde ve eşit ölçüde uygulanmasının mantıklı, uygun maliyetli bir yazılım güvenlik programı oluşturabileceği belirtilmektedir.

Şekil 22: Temas Noktaları Ayakları

McGraw tarafından hazırlanan bu modelde güvenliği sağlamak için uygulanması gereken en iyi uygulamalar temas noktası (Touchpoint) olarak adlandırılmıştır. Yazılım güvenliği en iyi uygulamalarını yazılım geliştirme yaşam döngüsüyle bütünleştirmek, yazılım güvenliğinin üç temel ayağının merkezidir (Gary McGraw 2006). Bahsedilen en iyi uygulamalar karşılaşılabilecek risklerin bilinmesi ve anlaşılması, güvenlik için tasarım yapılması ve tüm yazılımlar için kapsamlı, objektif risk analizleri ve testlerine tabi tutulmasını kapsamaktadır. Bu modelde belirlenmiş yedi adet temas noktası bulunmaktadır. Temas noktaları içinde tanımlanan aktiviteler için bir etkililik sıralaması önerilmektedir (Emiral 2009). Bu sıralama;

- Kod incelemesi
- Mimari risk analizi
- Sızma testleri
- Risk tabanlı güvenlik testleri
- Kötüye kullanım davaları
- Güvenlik gereksinimleri
- Güvenlik işlemleri(Mcgraw 2009)

Modelde anlatılan temas noktaları, geleneksel bir yazılım geliştirme sürecinde zaten üretilen yazılım eserlerine dayanmaktadır. Güvenlik sağlamaya başlamak için tüm temas noktalarının benimsenmesi ve uygulamaya geçirilmesi gerekmemektedir. Ancak tüm temas noktalarının uygulamaya geçirilmesi önemle tavsiye edilmektedir. Bazı temas noktaları diğerlerinden daha etkilidir ve en etkili olanlar ilk olarak benimsenmelidir. Yazılım güvenlik temas noktaları, sistemin asıl tasarımında ve uygulamasında yer almayan kişiler tarafından en iyi şekilde uygulanabilmektedir.

Yazılım güvenliğinin üçüncü ayağı olan bilgi/tecrübe, güvenlik bilgisinin toplanmasını ve paylaşılmasını kapsamaktadır. Yazılım güvenlik bilgi/tecrübe ayağı ilkeler, kılavuzlar, kurallar, güvenlik açıkları, saldırı desenleri ve geçmişte karşılaşılan riskleri kataloglamayı gerektirmektedir.

Yazılım Güvenliği Temas Noktaları Modelinin olumlu yönlerinin başında kavramsal olarak başarılı anlatıma sahip olması gelmektedir. Yazılım güvenliği konusunda ilk okunacak kaynaklardan biri olarak gösterilmektedir.

Yazılım Geliştirme Yaşam Döngülerinden bağımsız olarak tanımlanmıştır ve bu esneklik sayesinde tüm metotlarla birlikte kullanılabilmektedir.

Kavramsal anlatım olarak başarılı olmasına rağmen diğer metotlara nazaran kaynak ve netlik açılarından geride kalması modelin en büyük eksikliklerinden birisidir. Model, Risk Yönetimi'ni yazılım güvenliğini destekleyen 3 kolondan biri olarak tanımlamasına rağmen temas noktaları içinde yer alan aktivitelerle bağlantısını net bir biçimde kuramamıştır.

3.4.4. Sistem Güvenlik Mühendisliği Yetenek Olgunluk Modeli (SSE-CMM)

Uluslararası Sistem Güvenliği Mühendisliği Birliği (ISSA) tarafından sürdürülen Sistem Güvenlik Mühendisliği Yetenek Olgunluk Modeli iyi bir güvenlik mühendisliği sağlamak için var olması gereken bir kuruluşun güvenlik mühendisliği sürecinin temel özelliklerini tanımlayan bir sistemdir (Information Systems Security Association 2017). Orijinal proje ABD Ulusal Güvenlik Dairesi (NSA) sponsorluğunda başlamıştır. Model, güvenlik mühendisliği uygulamalarını değerlendirmek ve hassaslaştırmak için mühendislik kuruluşları tarafından, bir sağlayıcının güvenlik mühendisliği yeteneğini değerlendirmek için müşteriler tarafından ve güvenlik mühendisliği değerlendirme kuruluşları tarafından örgütsel yeteneğe dayalı güveni oluşturmak için kullanılabilir (SSE-CMM (Systems Security Engineering Capability Maturity Model) y.y.). Model, ISO/IEC 21827 adı ile standartlaştırılmış, güvenlik mühendisliği uygulamaları için aşağıdakileri kapsayan bir metriktir:

- Geliştirme, işletme, bakım ve hizmetten çıkarma faaliyetleri de dâhil olmak üzere proje yaşam döngüsü
- Yönetim, kurumsal ve mühendislik faaliyetleri
- Sistem yazılımı ve donanımı, insan faktörleri, test mühendisliği gibi diğer disiplinler ile eşzamanlı etkileşimler; Sistem yönetimi, çalıştırma ve bakım
- Edinme, sistem yönetimi, sertifikasyon, akreditasyon ve değerlendirme dâhil olmak üzere diğer kuruluşlarla olan etkileşimler.

Tablo 12: SSE-CMM Modelinin olgunluk seviyeleri

Seviye	Açıklama
1	Gayri resmi Olarak Gerçekleştirildi
2	Planlı ve İzlenen
3	İyi Tanımlanmış
4	Nicel Kontrollü
5	Sürekli İyileştirme

SSE-CMM proje ve organizasyon süreçleri ve güvenlik mühendisliği olmak üzere iki ana başlık altında toplanmıştır. Güvenlik mühendisliği başlığı toplamda 22 süreç (mühendislik süreçleri, güvenlik süreçleri ve risk süreçleri) içermektedir.

Güvenlik mühendisliği modeli güvenlik değerlendirmesi yapabilmek için genel çerçeveyi oluşturma konusunda eksik kalmaktadır (Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan 2009). SSE-CMM modelinin başlıca amacı uygulamaların performansını belirlemek ve performansta iyileştirmektir.

3.4.5. Ortak Kriterler (ISO 15408)

Ortak Kriterler, Kanada, Fransa, Hollanda, İngiltere, Almanya ve Amerika Birleşik Devletleri'nin ulusal güvenlik organizasyonları ve standartlar enstitüleri ile birlikte ortak bir çalışma sonucunda bu ülkelerde kullanılan güvenlik değerlendirme kriterlerinin yerine kullanılması amacıyla ortaya çıkmıştır (TSE- Ortak Kriter Nedir). Ortak Kriterler - ISO 15408 bilgi teknolojileri ürün ve/veya sistemlerin güvenlik seviyelerinin tespit edilmesi ve test edilebilmesi için geliştirilmiş olan, temelini TCSEC ve ITSEC standartlarından alan ve Uluslararası Standartlar Organizasyonu'nun 1999 yılında Uluslararası Bilgi Teknolojileri Güvenlik Değerlendirme Standardı olarak kabul ettiği güvenlik standardıdır. Türkiye 2003 yılında Ortak Kriterler Tanıma Sözleşmesini imzalamış, 2010 yılında "Sertifika Üretici Ülke" unvanını almıştır (Alkan 2013).

Ortak kriterler birbirleriyle ilişkili üç ayrı bölümden oluşmaktadır (TSE- Ortak Kriter Nedir).

Bölüm 1, Giriş ve Genel Bölüm: Bu bölüm güvenlik değerlendirmelerinin temel görüş ve ilkelerini tanımlamaktadır. Buna ek olarak genel bir değerlendirme modeli önermektedir. Aynı zamanda güvenlik hedeflerinin oluşturulması, güvenlik gereksinimlerinin belirlenmesi ve tanımlanması konusunda bilgiler içermektedir. Ayrıca standartta bulunan bölümlerinin potansiyel kullanıcılar (yönetici, ekip lideri, yazılım geliştirici vb.) için nasıl kullanılacağı bu bölümde tanımlanmaktadır.

Bölüm 2, Güvenlik Fonksiyonel Gereksinimleri: Güvenlik Fonksiyonel Gereksinimleri, değerlendirme hedefinin güvenlik fonksiyonel gereksinimlerinin standart bir dille anlatılabilmesini sağlamak için tanımlanmış olan güvenlik bileşenleri bu bölümde listelenmektedir. Ortak Kriterlerin ikinci bölümünde 60adet güvenlik işlevsel gereksinimini içeren 11 tane fonksiyonel sınıf (Kara 2004). Bu sınıflar;

- FAU: Güvenlik Denetimi (Audit)
- FCO: İletişim (Communication)
- FCS: Kriptografik destek (Cyriptographic support)
- FDP: Kullanıcı Verilerinin Korunması (User data protection)
- FIA: Tanıma ve Kimlik Doğrulama (Identification and authentication)
- FMT: Güvenlik Yönetimi (Security Management)
- FPR: Gizlilik (Privacy)
- FPT: Değerlendirme Hedefi Güvenlik Fonksiyonlarının Korunması (Protection of TSF)
- FRU: Kaynak kullanımı (Resource utilisation)
- FTA: Değerlendirme Hedefi erişimi (TOE access)
- FTP: Güvenilir yollar/kanallar (Trusted path/channels)

Bölüm 3, Güvenlik Garanti Gereksinimleri: Güvenlik Garanti Gereksinimleri, değerlendirme hedefinin güvenlik garanti gereksinimleri bu bölümde listelenmiştir. Standardın üçüncü bölümü garanti bileşenlerini, ailelerini ve sınıflarını kataloglar halinde tanımlamaktadır. Bu bölüm aynı zamanda Koruma Profillerinin ve Güvenlik Hedeflerinin değerlendirme kriterlerini ve değerlendirme garanti seviyelerini oluşturan garanti bileşenlerini

de içermektedir. Ortak Kriterlerin üçüncü bölümünde sekiz adet garanti sınıfı vardır (Kara 2004). Bu sınıflar;

- ACM: Konfigürasyon yönetimi (Configuration Management)
- ADO: Dağıtım ve işletim (Delivery and Operation)
- ADV: Geliştirme (Development)
- AGD: Kılavuz dokümanları (Guidance Documents)
- ALC: Hayat döngüsü desteği (Life Cycle Support)
- ATE: Testler (Tests)
- AVA: Açıklık değerlendirmesi (Vulnerability Assessment)
- AMA: Garantinin sürdürülmesi (Maintenance of Assurance)

Ortak kriterler bünyesinde bulunan bütün sınıflar bir veya daha fazla bileşen içermektedir. Bu bileşenler arasında bir hiyerarşi bulunmaktadır.

Ortak Kriterler Değerlendirme Garanti Seviyesi (Evaluation Assurance Levels EAL) olarak bilinen yedi adet garanti paketi tanımlamaktadır. Bu yedi garanti seviyesi aşağıdaki gibidir (Tiirik 2004);

- EAL1 Bu seviye ürünün veya sistemin doğru çalıştığına dair güvenin yeterli olduğu ve güvenlik tehditlerinin ciddi olmadığı durumlarda kullanılmaktadır.(Fonksiyonel test)
- EAL2 Ürün geliştirici tasarım bilgilerini ve test sonuçlarını değerlendirme laboratuvarına iletmelidir. EAL2 değerlendirmesi, müşteriler veya ürün geliştiriciler, düşük ve orta düzey seviye arasında bir güvenlik gereksinimi duyuyorlar ise ve ürünün geliştirme dokümanlarının tamamına ulaşamıyorlar ise uygulanmaktadır. (Yapısal Test) (Black box)
- EAL3 seviyesinde standart, ürün geliştiriciye tasarım sırasında maksimum garanti sağlayabilmesi için yöntemler önermektedir. EAL3 değerlendirmesi üreticinin test sonuçlarının seçilerek onaylanması ve bilinen açıklıkların üretici tarafından incelendiğinin kanıtlanmasını içeren gri kutu testleri (grey box testing) ile desteklenmektedir. Ayrıca geliştirme ortamı kontrolleri ve ürünün konfigürasyon yönetimi delilleri değerlendirmeler için gerekmektedir. (metodik test ve kontrol)

- EAL4 seviyesi ticari ürün geliştirme yöntemlerinden maksimum garanti sağlayabilmek için ürün geliştiricilere yöntemler önermektedir. EAL4 var olan ürün geliştirme altyapısını değiştirmeden ulaşılabilecek en yüksek garanti seviyesidir. EAL4 değerlendirmesi ürünün alt düzey tasarımı ve uygulamanın alt kümelerinin analizi ile de desteklenen bir süreçtir. Yapılan testler bağımsız açıklık analizleri ile desteklenir. Geliştirme kontrolleri yaşam döngüsü desteği, tanımlama teknik ve araçları, ve otomatik konfigürasyon yönetimi ile güçlendirilir. (metodik tasarım, test ve kontrol)
- EAL5 seviyesi, özel güvenlik tekniklerinin orta düzeyde uygulanması ile desteklenen, ticari ürün geliştirme yöntemlerinden maksimum garanti sağlayabilmek için ürün geliştiricilere yöntemler önermektedir. Bu seviyeye aday bir ürün seviyenin gerektirdiği garantiyi sağlayabilecek bir şekilde tasarlanmalı ve geliştirilmelidir. Bu seviye ürün geliştiricileri ve müşteriler yüksek seviyede güvenlik ve bağımsız bir garanti ihtiyacı duyduklarında kullanılmaktadır. (semiformal dizayn ve test)
- EAL6 seviyesi yüksek değerdeki varlıkları korumakta olan ürünler için yüksek garanti seviyesi sağlayan güvenlik teknikleri önermektedir. (semiformal doğrulanmış dizayn ve test)
- EAL7 seviyesi son derece yüksek risk durumlarında veya korunan varlıkların bu seviyenin getireceği maliyeti karşılayabileceği durumlarda uygulanabilmektedir. EAL7 değerlendirmesinde fonksiyonel spesifikasyonun ve üst düzey tasarımın biçimsel bir sunumu ile biçimsel bir model sunulmaktadır. Ürün geliştiricinin beyaz kutu testlerinin (white box testing) kanıtları ve bu test sonuçlarının bağımsız bir onayı gerekmektedir. Tasarımın karmaşıklığı en düşük değerde olmalıdır.

Ortak kriterler metodolojileri kullanılarak gerçekleştirilen projelerin çoğunluğu EAL1'den EAL3'e kadar olan daha düşük güvence seviyelerinde değerlendirilmesine rağmen, Ortak Kriterler olgun bir uygulamadır. Türkiye sertifika müşterisi olarak Ortak Kriterlere göre değerlendirilmiş ürünleri kabul etmekle birlikte 2005 yılından beri bilgi teknolojileri ürünlerini EAL4 seviyesine kadar değerlendirebilmektedir (Kara 2004). Ortak Kriterler Standardının EAL 4 ve üzeri garanti seviyesi için tasarım sürecine getirdiği disiplin ve metodoloji aşağıdaki maddeleri içermektedir;

- Risk Analizi Faaliyeti
- Metodolojik Tasarım
- Yaşam Döngüsü Modeli
- Geliştirme Araçları
- Geliştirme Ortamı
- Hata Düzeltme
- Fonksiyonel Testler

Ortak Kriterler genellikle bir ürünün veya sistemin güvenlik özellikleri tespit edilirken, güvenlik özellikleri eklenirken, güvenlik özellikleri değerlendirilirken veya güvenlik özellikleri olan bir ürün veya sistem satın alınırken kullanılmaktadır (TSE- Ortak Kriter Nedir). Ortak Kriterler Standardının uygulanması ve değerlendirilmesi çok ayrıntılı olduğu için kullanacak kurum ve kuruluşlar için maliyetli olabilmektedir.

3.4.6. Güvenli Yazılım Geliştirme Modelleri Genel Değerlendirme

Yazılım geliştirme süreçlerinde güvenlik unsurunun ele alınması ve elde edilen yazılım ürününün güvenli olmasını sağlamak adına önerilen modellerden bir kaçı yukarıdaki bölümlerde anlatılmıştır. Tez çalışmasının bu bölümünde yazılım geliştirme döngülerini geliştirmek ve güvenli hale getirmek için kullanılabilecek modeller olan SAMM, Microsoft-SDL, SSE-CMMI ve Ortak Kriterler ile ilgili genel değerlendirmeler yapılmıştır. Değerlendirmeler, Tablo 13'te verilmiştir.

Tablo 13: Güvenli Yazılım Geliştirme Modelleri Genel Değerlendirme

Değerlendirme Kriterleri	SAMM	MSDL	Temas Noktaları	SSE-CMM	Ortak Kriterler

Odak noktası	Yazılım güvenliği	Yazılım güvenliği	Yazılım güvenliği	Sistem geliştirme güvenliği	Ürün güvenliği değerlendirme
Uygulamaya Geçirme (Tavsiye edilen güvenlik eylemleri)	Tek / Hepsi	Hepsi	Tek/ Hepsi	Tek/ Hepsi	Hepsi
Yapı	Esnek	Katı	Esnek	Esnek	Katı
Eğitim ve Farkındalık	+	+	-	+	-
Fiziksel-Mantıksal Güvenlik	-	-	-	+	+
Güvenli Yapılandırma	-	-	+	+	+
Tehdit Modellemesi	+	+	+	+	+
Risk Analizi	+	+	+	+	+
Güvenlik Gereksinimleri	+	+	+	+	+
Güvenli Mimari	+	+	+	+	+
Güvenli Tasarım	+	+	+	+	+
Kaynak Kod Analizi	+	+	+	-	-
Zafiyet Analizi	+	+	+	+	-
Güvenlik Doğrulaması	+	+	-	+	+
Zafiyet Yönetimi	+	+	-	+	+
Güvenli Geliştirme Teknikleri	+	+	+	+	-
Operasyonel Ortama Bilgi Aktarma	+	+	+	+	+
Diğer Bileşenlerle Güvenli Entegre	+	+	+	+	+
Güvenli Teslim Etme	+	-	-	+	+

(Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan 2009)

4. GÜVENLİ UYGULAMA YAZILIMI GELİŞTİRME ARAŞTIRMASI, BULGULAR VE YORUM

4.1. Yöntem

Bu bölümde sırasıyla araştırmanın modeline, araştırma ve çalışma grubuna, verilerin toplanmasına ve verilerin analizine ilişkin bilgiler ile bulgular ve yorumlara yer almaktadır.

4.1.1. Araştırma Modeli

Ülkemizde kamu sektöründe ve özel sektörde yazılım geliştirici olarak çalışan bilişim personelinin "Güvenli Yazılım Geliştirme" konusu hakkındaki

düşüncelerini, uygulamalarını öğrenebilmek ve ülkemizde geliştirilen yazılımlarda hangi metodolojilerin kullanıldığı konusunda fikir edinebilmek amacıyla yapılan bu çalışma tarama araştırma modeli kullanılarak gerçekleştirilmiştir.

Tarama modeli nesnelerin, toplumların, kurumların, olayların doğasını ve özelliklerini tanımlamayı, geçmişte veya halen var olan durumlarını var olduğu şekilde betimlemeyi hedeflemektedir. Tarama modelleri araştırma konusuna ilişkin verilerin toplanması, sınıflandırılıp düzenlenmesi ve çözümlenmesi süreçlerinden oluşur. Bu modelde, araştırmaya katılan bireylerin görüşleri, herhangi bir değiştirme çabası içinde bulunulmadan kendi ortamlarında tanımlanmaya çalışılır (Karasar 2012). Tarama modelinde, katılımcıların yetenekleri, tercihleri, davranışları veya fiziksel ortamların özelliklerini tanımlarlar. Tarama araştırmalarının üç temel özelliği bulunmaktadır (Büyüköztürk, ş. Kılıç Çakmak, E. Akgün 2010).

- Araştırılan konuya ilişkin katılımcıların görüşlerinin ya da özelliklerinin (bilgi, beceri, kaygı, ilgi, vb.) betimlenmesi için, topluluğu temsil edebilecek insanlardan oluşan bir parça seçilir (Evrenden örneklemin seçilmesi).

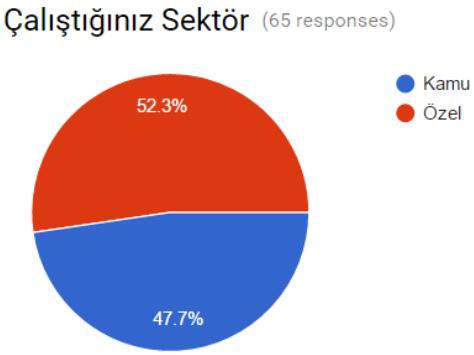
- Araştırma için ihtiyaç duyulan verileri toplama süreci, veri kaynakları olan kişilere yöneltilen sorulara verilen cevaplara dayalıdır.

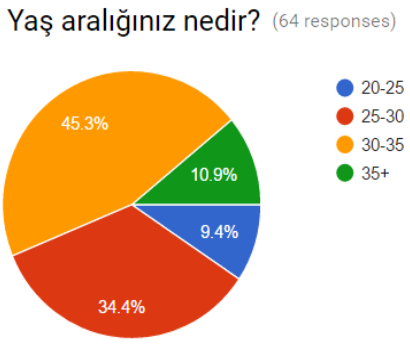
- Veriler, özelliği betimlenecek topluluğun her bireyinden değil, bu topluluğu temsil eden bir parçasından, yani örneklemden toplanır.

4.1.2. Araştırma- Çalışma Grubu

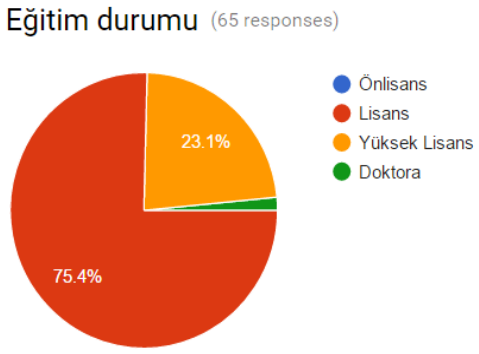
Bilişim dünyasında aktif olarak bulunan ve yazılım geliştiren çalışanların "Güvenli Yazılım Geliştirme" konusu hakkındaki görüşlerinin neler olduğunu belirlemek amacıyla kamu ve özel sektör çalışanlarının oluşturduğu 65 kişi araştırmanın çalışma grubunu oluşturmuştur (Son erişim tarihi 10/03/2017). Anılan çalışma grubundan Google Belgeler aracılığıyla çevrimiçi olarak doldurdukları form kullanılarak görüşleri alınmıştır.

Araştırmaya konu olan yazılım geliştiren bilişim personelinin oluşturduğu çalışma grubunun bazı genel özellikleri ve bu özelliklere ait dağılımlar aşağıdaki gibidir;





Yukarıda görüldüğü gibi araştırmaya dâhil olan katılımcıların %52.3'ü özel sektörde, %47.7'si kamu sektöründe çalışmaktadır. Katılımcıların çalıştıkları kurumlarda bilişim ile ilgili birimlerde ve mühendislik alanlarında görev yaptığı bilgisi edinilmiştir. Toplam 65 kişiden oluşan çalışma grubunun %45.3'ü 30-35 yaş aralığında, %34.4'ü 25-30 yaş aralığında, %10.9'u 35 yaşın üstünde ve %9.4'ü 20-25 yaş aralığında bulunmaktadır.



Katılımcıların eğitim durumu incelendiğinde 49 kişinin lisans, 15 kişinin yüksek lisans ve 1 kişinin doktora mezunu olduğu görülmektedir. Çalışma grubunda bulunan kişilerin 21'i 3-6 yıl arasında bir süredir yazılım geliştirdiğini bildirmiştir. Bunu 10 yıldan daha fazla süredir yazılım geliştirdiğini belirten 17 kişi, 0-3 yıl arası bir süre yazılım geliştiren 16 kişi takip etmektedir. Son olarak 11 kişinin 6-9 yıl arasında yazılım geliştirme tecrübesi olduğu görülmektedir. Buradan da anlaşılabileceği gibi çalışma grubunda bulunan insanların çoğunun (%56.9) yazılım geliştirme tecrübesi 6 yıl veya daha azdır. Yazılım sektörünün ülkemizde genç bir sektör olması bu durumun temel kaynağıdır.

3. Ve 4. Sorularda katılımcıların hangi birimde ve hangi pozisyonlarda çalıştıkları araştırılmıştır. Katılımcıların büyük çoğunluğu bilgi işlem birimlerinde çalıştıklarını belirtmişlerdir. Görev yaptıkları pozisyonlara göre yazılım geliştirme uzmanları, mühendisler, proje ve sistem yöneticileri çalışma grubunda çoğunluğu oluşturmaktadır.

4.1.3. Veri Toplama Aracı

Yazılım geliştirme konusunda çalışan bilişim personelinin görüşlerinin neler olduğunu belirlemek amacıyla "Güvenli Yazılım Geliştirme Anketi" isimli form hazırlanmıştır.

Görüşleri belirlemek amacıyla kullanılan formda aşağıdaki sorular yer almıştır;

1- Kurumunuzda Bilgi Güvenliği ve Siber Güvenlik konularının yeterince ele alındığını düşünüyor musunuz?
2- Kurumunuzda ISO 27001 Sertifikası mevcut mu?
3- Aşağıdaki güvenlik tehditlerinden en çok endişe duyduğunuz konular hangileridir?
4- Güvenli Yazılım Geliştirmenin, Siber Güvenlik açısından önemine bir puan veriniz. (1-En az, 10- En çok)
5- Güvenli Yazılım Geliştirme konusunda eğitim aldınız mı?
6- "Güvenlik" konusu yazılım geliştirme aşamalarının hangisine dâhil edilmelidir?
7- Yazılım geliştirme aşamasında aşağıdaki konulara verdiğiniz önemi puanlayınız. (En az 1- En çok 10)
8- Kurumunuzda yazılım geliştirme projelerinde benimsenen Yazılım Yaşam Döngüsü modeli aşağıdakilerden hangisidir?
9- Kurumunuzda yazılım geliştirme projelerinde benimsenen Güvenli Yazılım Yaşam Döngüsü nedir?
10- Kurumunuzda geliştirilen yazılımlar için güvenlik politikası mevcut mu?
11- Kurumunuzda dışarıdan edinilen yazılımlar için aranılan güvenlik isterleri nelerdir?

12- Kurumunuz bünyesinde bulunan yazılımlar için aşağıdaki testlerden hangileri yapılmaktadır?

13- Görüş ve Önerileriniz.

4.1.4. Verilerin Analizi

Nitel yöntemle yapılan araştırmalarda kullanılan bilgi toplama teknikleri sonucunda elde edilen bilgiler veriye dönüştürüldükten sonra verilerin çözümlenmesi için iki genel yöntem kullanılabilir. Birinci yöntem; derinlemesine analiz gerektirmeyen ve verilerin incelenmesinde kullanılan betimsel analizdir. İkinci yöntem ise elde edilen verileri daha yakından incelemeyi ve bu verileri açıklayan kavram ve temalara ulaşmayı gerektiren içerik analizidir. (Altındağ,2005)

Yazılım geliştiricilerin "Güvenli Yazılım Geliştirme" konusu hakkındaki görüşlerinin neler olduğunu belirlemek amacıyla hazırlanan form vasıtasıyla elde edilen verileri betimlemek için içerik analizi yöntemi kullanılmıştır.

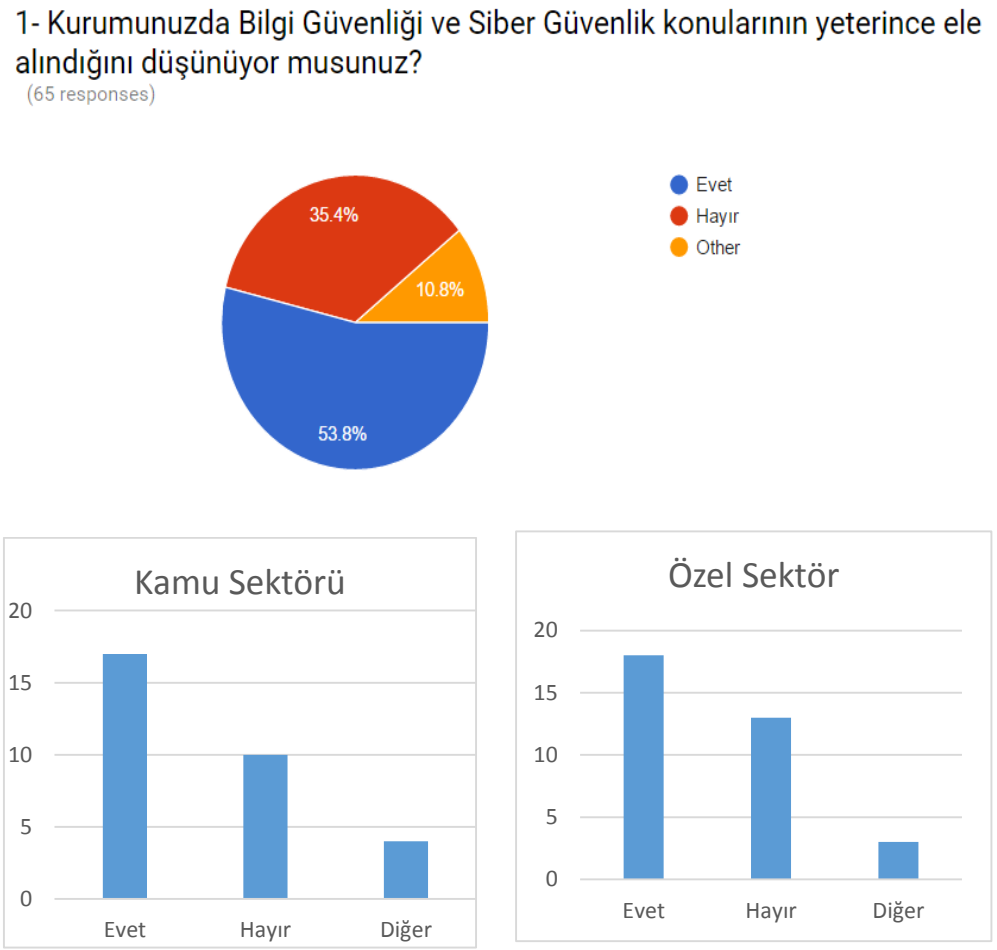
Anket yoluyla elde edilen verileri analiz edebilmek için aşağıdaki adımlar takip edilmiştir;

1. Görüşler konu ile ilgili soruları içeren çevrimiçi soru formu ile toplanmıştır.
2. Soru formundaki sorulara verilen cevaplar benzerliklerine göre kategorize edilmiştir.
3. Her bir soru için belirlenen kategoriler gözden geçirilip tekrarlar giderilerek, birbirine yakın ve benzer kategoriler birleştirilmiştir.
4. Her bir soru için oluşturulan kategoriler, içerikleri dikkate alınarak belli ana başlıklar altında toplanmıştır.
5. Katılımcıların kategorilere ilişkin görüşleri frekans ve yüzde olarak görselleştirilmiştir.

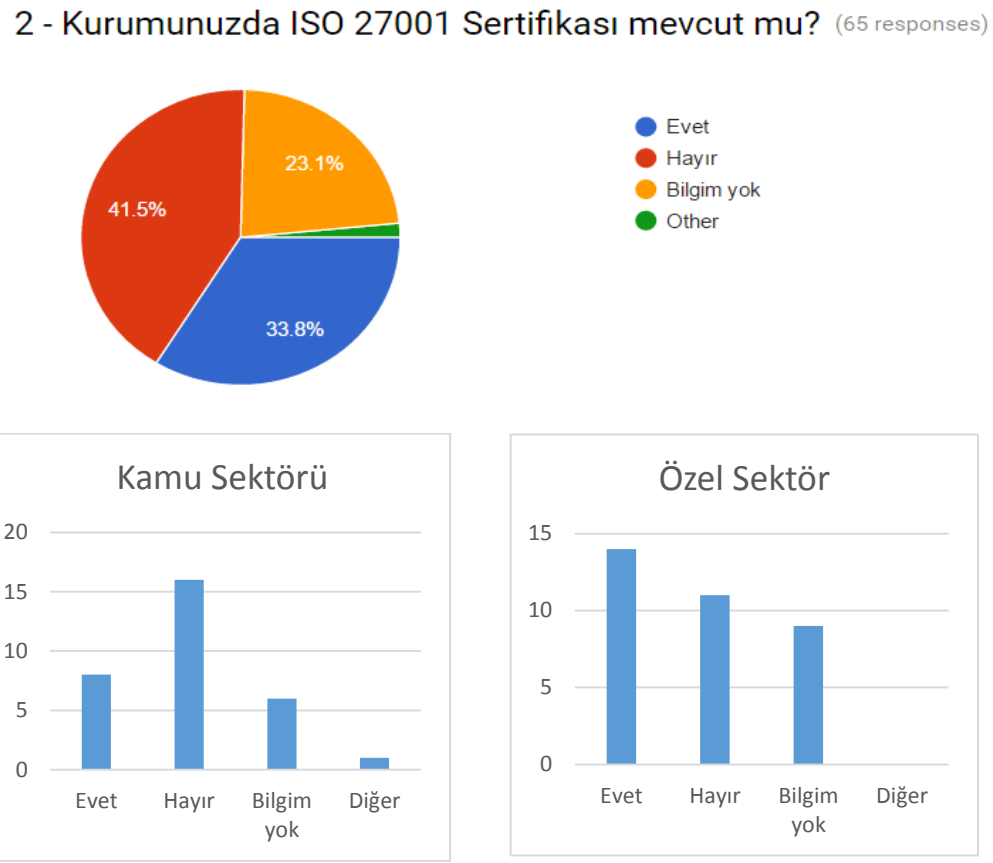
4.2. Bulgular ve Yorum

Araştırmanın bu bölümünde, katılımcıların "Güvenli Yazılım Geliştirme" konusu hakkında görüşlerini belirlemek amacıyla hazırlanan anket formu aracılığıyla yöneltilen sorulara verdikleri cevapları içeren analiz formlarına, her soru özelinde gelen görüşlerin belirlenen kategorilere ayrılarak

bu kategorilerdeki dağılımların gösterildiği analiz sonuçlarına ve eldeki verilerin analizi sonucunda ortaya çıkan bulgular ile bu bulgular baz alınarak yapılan yorumlara yer verilmiştir.

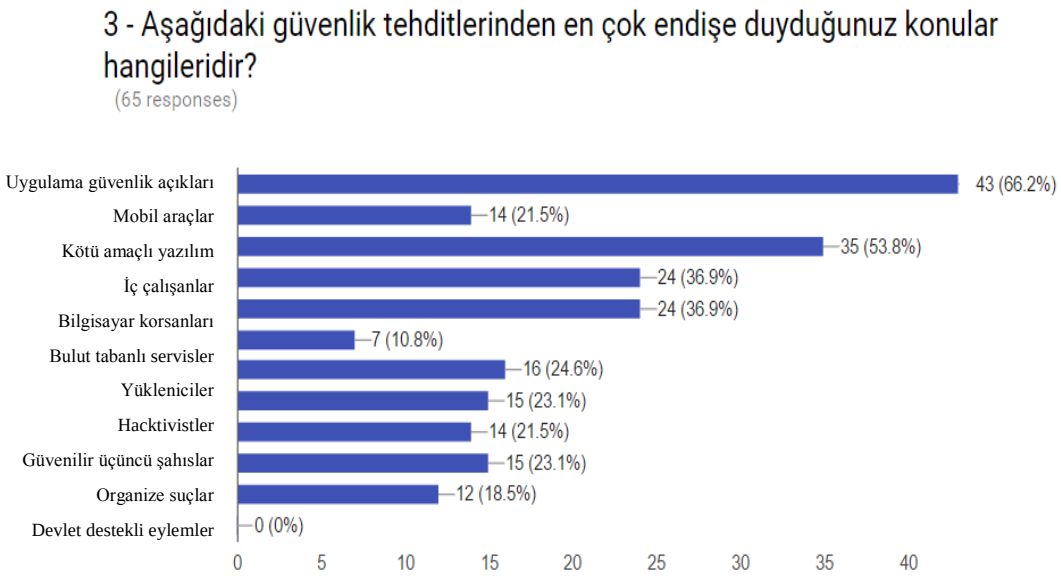


Yukarıdaki grafiklerde de görüldüğü üzere hem kamu sektöründe hem de özel sektörde katılımcıların çoğunluğu (%53.8) çalıştıkları kurumun bilgi güvenliği ve siber güvenlik konusunda yeterli çalışmaların yapıldığını düşünmektedir. Bunun yanında %35.4'lük kesim ele alınmadığını ve %10.8'i oluşturan 7 kişilik grup ise konu ile ilgili bilgisi olmadığını belirtmiştir.

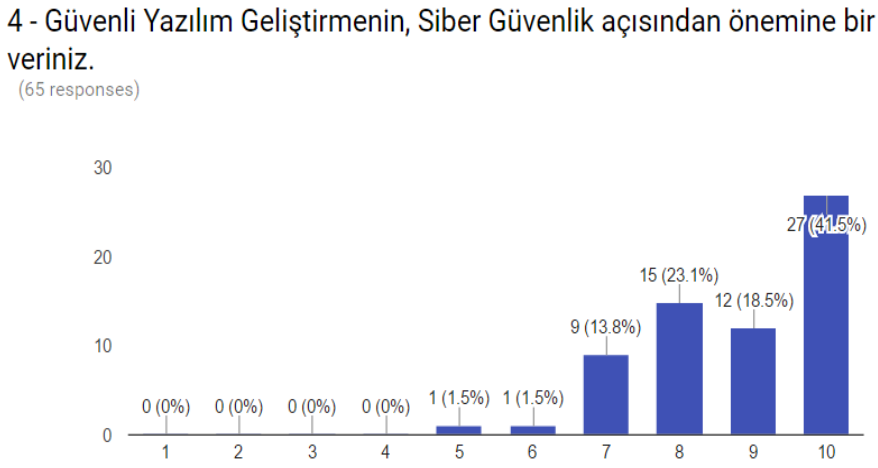


Yapılan araştırma sonucunda TS ISO/IEC 27001:2005 Bilgi Güvenliği Yönetim Sistemi Sertifikasının özel sektörde daha yaygın olarak edinildiği görülmüştür. Katılımcıların %23,1'i ise kurum bünyesinden böyle bir sertifika bulunup bulunmadığından haberdar olmadığını söylemiştir. Kamu sektörüne mensup 1 katılımcıda çalıştığı kurumda sertifika çalışmalarının devam ettiğini belirtmiştir.

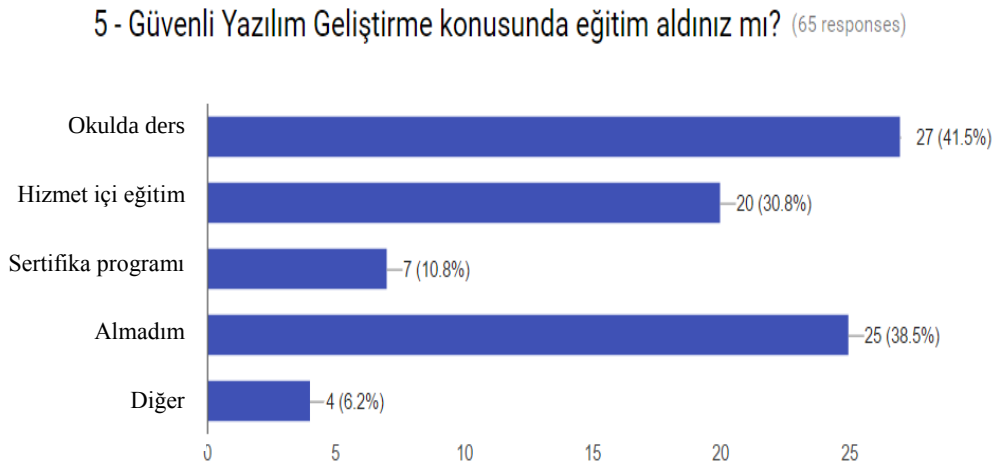
Gerçekleştirilen Güvenli Yazılım Geliştirme Araştırmasında sorulan 3. Soru yazılım geliştiriciler arasında en çok endişeye duyulan konuyu tespit etmeye yöneliktir. Bu soruya verilen cevaplara göre katılımcılar %66,2'lik oranla en çok "Uygulama güvenlik açıkları" seçeneğini işaretlemişlerdir. Bu soruya verilen cevaplar, Uluslararası Bilgi Sistemi Güvenlik Sertifikasyon Konsorsiyumu (ISC)² tarafından 2013 yılında güvenlik yöneticileri arasında yapılan ve Resim-8 de sonuçları gösterilen araştırmayla benzerlik göstermektedir. Buradan ülkemizdeki yazılım geliştiricilerin dünyadaki yazılım geliştiricilerle benzer endişelere sahip olduğu sonucuna ulaşabiliriz. Endişe duyulan diğer konuların sıralaması aşağıda gösterilmiştir.

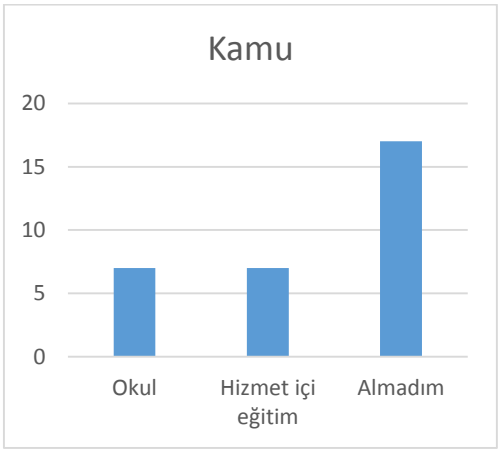


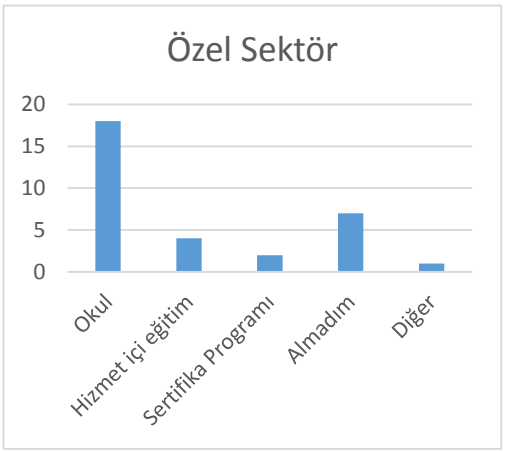
"Güvenli Yazılımı Geliştirme" konusunun siber güvenliği sağlama konusundaki önemine katılımcıların %41.5'i 10 puan (1-En az, 10-En çok) verilmiştir. Ayrıca katılımcıların bu konuya büyük ölçüde önem verdiği aşağıdaki grafikte de görülebilmektedir. Güvenliği sağlamak için atılması gereken ilk adımlardan birisi olan "Güvenli Yazılım Geliştirme" konusunda farkındalığın katılımcılar arasında yüksek olduğu sonucuna varılmıştır.



Katılımcılar arasında farkındalığın yüksek olmasına rağmen eğitim konusunda eksik kalındığı görülmüştür. Çalışma grubundan 25 kişi konu ile ilgili eğitim almadığını, 3 kişi de yetersiz bir eğitim aldığını belirtmiştir.

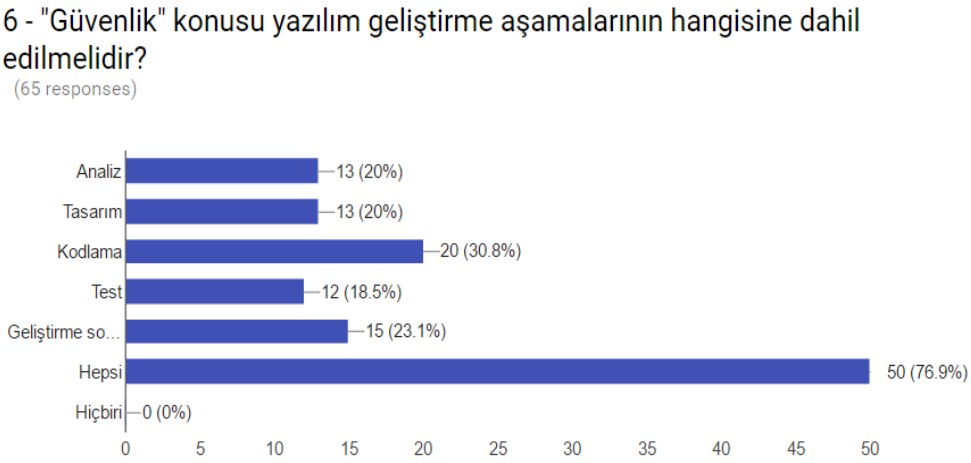




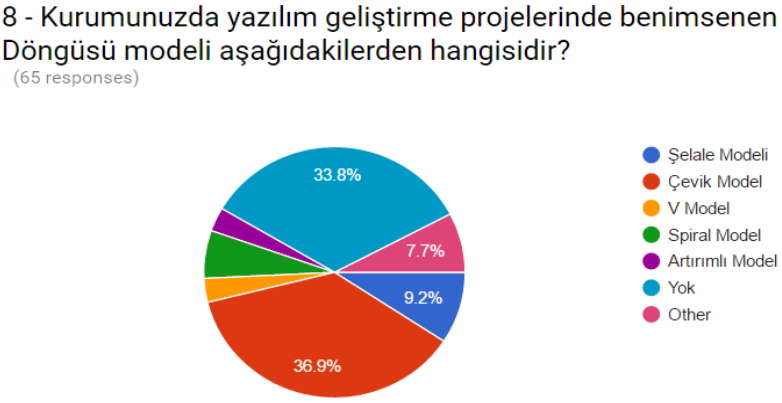


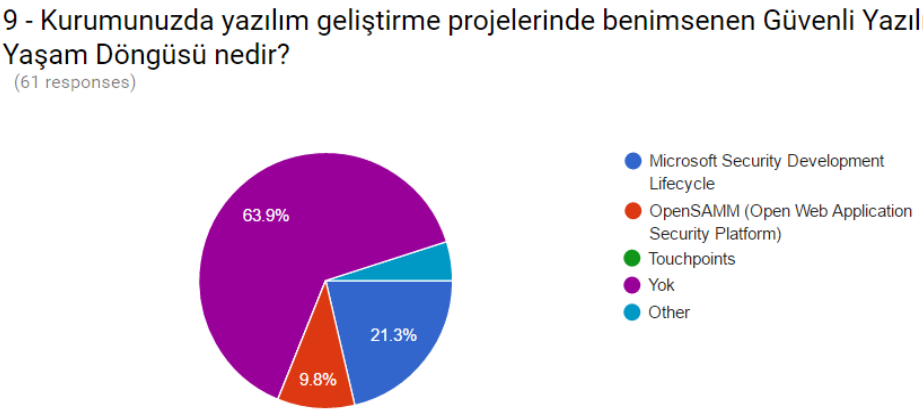
Kamu çalışanları arasında "Güvenli yazılım geliştirme" ile ilgili hiç eğitim almadım diyenlerin sayısının çoğunlukta olduğu görülmüştür. Özel Sektör çalışanları arasında ise tam aksi bir durum olduğu sonuçlardan anlaşılmaktadır.

Yazılım geliştiricilerin konu hakkındaki farkındalıklarını ölçmek için sorulan bir diğer soruda 6. Sorudur. Bu soruya verilen cevaplar yazılım geliştiricilerin güvenlik konusunda bilinçli olduklarını göstermektedir.



Araştırma kapsamında katılımcılara yöneltilen 8. ve 9. Sorularda ülkemizde yazılım geliştirme sırasında kullanılan metodolojiler belirlenmeye çalışılmıştır. Elde edilen sonuçlara göre Çevik Yazılım Geliştirme Yaşam Döngüsünün %36.9'luk oranla en sık tercih edilen model olduğu görülmüştür. Yazılım yaşam döngülerine güvenliği bütünleştirmek için kullanılan modellerin çoğunlukla kullanılmadığı %63.9'luk bir oranla ortaya çıkmıştır.



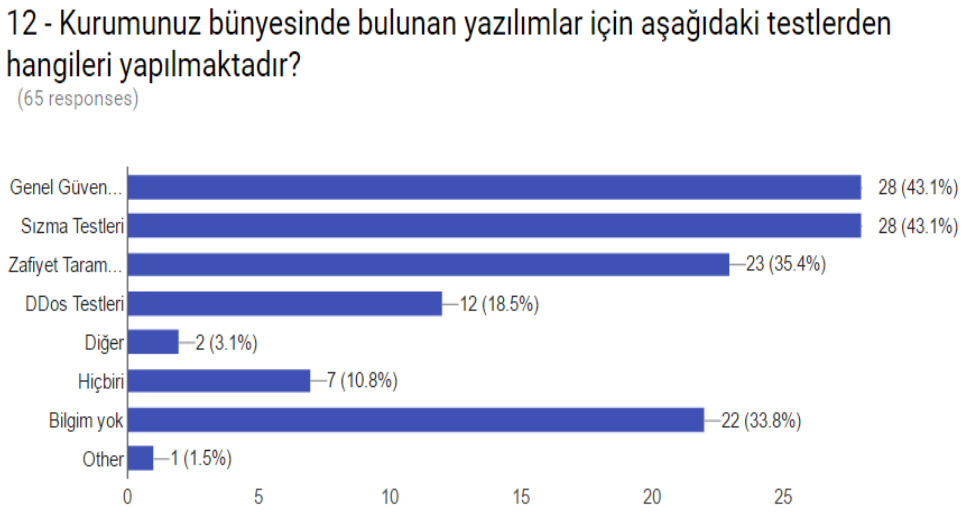


Güvenli yazılım geliştirebilmek için kurumlarda uluslararası düzeyde yapılan çalışmalar sonucu ortaya çıkan modeller ve metodolojiler kullanılmamasına rağmen kurumların kendi bünyelerinde güvenlik politikası oluşturduğu 10. Soruya verilen cevaplarda görülmektedir.

Tablo 14: "Soru 11: Kurumunuzda dışarıdan edinilen yazılımlar için aranılan güvenlik isterleri nelerdir?" İçerik analiz tablosu

Görüşlerden Elde Edilen Temel İfadeler	Toplam Görüş Sayısı	Yüzde (%)
Bilgim yok	9	37.5
Kurumsal güvenlik politikalarına uygunluk	4	16.6
Uluslararası standartlara uygunluk	1	4.16
Diğer birimler tarafından takip ediliyor.	2	8.33
Kullanıcı yönetimi, yetkilendirme modülü, şifreleme, vb.	8	33.3

Kurumların bünyelerinde barındırdığı uygulamalara Genel Güvenlik Testleri, Sızma Testleri ve Zafiyet Tarama Testlerinin diğer testlere oranla daha fazla yaptırdığı görülmüştür.



Tablo 15' te Katılımcılara konu hakkındaki görüş ve önerileri sorulmuştur. Bu soruya verilen cevaplar arasından

Tablo 15: "Soru 13: Görüş ve Önerileriniz" İçerik analiz tablosu

Görüşlerden Elde Edilen Temel İfadeler
Yazılım geliştirmede ele alınması gereken en önemli hususlardan biri güvenlik kısmıdır.
Güvenli uygulamalar geliştirebilmek için tüm testler eksiksiz yapılmalı ve karşılaşılabilecek her tehlikeli durum göz önünde bulundurulmalıdır.
Personelin konu hakkında eğitilmesi ve farkındalık kazandırılması gereklidir.
Yazılım güvenliği konusunda bir kontrol listesi hazırlanmalı
Bilgi güvenliği konusunda odak genellikle ağ ve sistem tarafında kalmaktadır. Yazılım geliştirme süreçleri kapsamında güvenliğin ele alınmasını önemsenmelidir. Anket Sonuçlarının ilgili yerlerle ve meslektaşlarımız ile paylaşılması gerekmektedir.
Güvenli yazılım geliştirmenin ülkemizde yeteri kadar önemsenmediği düşünülmektedir.

Yapılan Güvenli Yazılım Geliştirme Araştırmasının sonucunda ulaşılan en çarpıcı husus katılımcıların sorulara sıklıkla verdiği “Bilgim yok” cevabı olmuştur. Örneğin katılımcıların yaklaşık ¼’lük kısmı kurumlarında ISO 27001 sertifikası bulunup bulunmadığını bilmediğini belirtmiştir. Katılımcılar güvenlik konusunun yazılım geliştirmede önemli bir unsur olduğunu düşünmektedir. Bu da ülkemizde belirli bir düzeyde farkındalığa ulaşılmış olduğunun göstergesidir. Buna rağmen süreçler henüz sistematik bir hale getirilememiştir. Ayrıca ülkemizde güvenli yazılım geliştirme konusunda yeterli eğitim seviyesine ulaşılamadığı saptanmıştır.

5. BAKANLIK İÇİN YOL HARİTASI VE ÖNERİLEN GÜVENLİ YAZILIM GELİŞTİRME MODELİ

2016-2019 Ulusal Siber Güvenlik Eylem Planında mevcut riskleri, belirlenen ilkeler ışığında asgari düzeye indirmeyi hedefleyen stratejik amaçlar yer almıştır. "Stratejik Siber Güvenlik Amaçları ve Eylemleri" başlığı altında 15.maddede verilmiş olan "Güvenli yazılım geliştirme ve tedarik yönetimi kültürünün oluşturulması" hedefi kapsamında güvenli yazılım geliştirme temel kurallarının belirlenmesi gerektiği belirtilmiştir (Ulaştırma Denizcilik ve Haberleşme Bakanlığı 2016). Kurumsal davranışların ve ihtiyaçların zaman içerisinde değişim göstermesi sebebiyle bu kuralların geliştirilmesi aşamasında kurumlar kendi bünyelerinde incelenmelidir. Uzun vadeli hedefler doğrultusunda çalışırken değişimler göz önünde bulundurulmalı ve yinelemeli bir şekilde ele alınmalıdır. Yapılan çalışmalarda tüm kurum ve kuruluşlar için geçerli olacak tek bir tarifi yoktur. Dolayısıyla geliştirilecek olan çözümler kurumların yapısına göre uyarlanmış olmalıdır. Bu tez çalışmasında da Çevre ve Şehircilik Bakanlığı'nın mevcut durumu incelenerek SWOT analizi ile güçlü ve zayıf yönleri ele alınmıştır. Çalışma sonucunda geliştirilen ve/veya temin edilen uygulama yazılımlarına bilgi güvenliği kapsamında siber güvenlik bakış açısının yerleştirilmesi ve güvenliğin sağlanması için faydalı olacağı düşünülen yazılım geliştirme metodolojisi ve yol haritası önerilmiştir.

5.1. Mevcut Durum Değerlendirilmesi

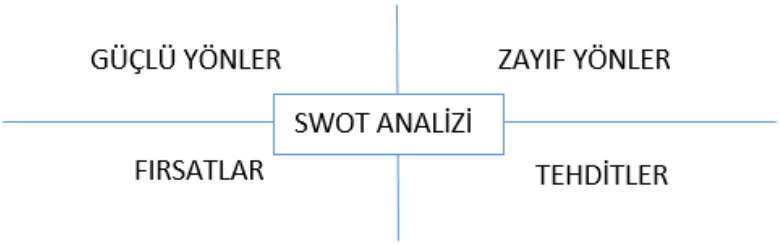
Bakanlığımız bünyesinde geliştirilen ve Bakanlık dışından edinilen her türlü uygulama yazılımı ile ilgili çalışmalar Coğrafi Bilgi Sistemleri Genel Müdürlüğü altında bulunan Yazılım Teknolojileri Dairesi Yazılım Geliştirme Şubesi ve Bilişim Projeleri Koordinasyon Şubeleri tarafından yürütülmektedir. Bilgi Güvenliği kapsamında yapılan çalışmalar ise Bilişim Ağları ve Sistem Dairesi Başkanlığına bağlı olan Bilgi Güvenliği Şubesi tarafından yönetilmektedir. Bakanlığımız bünyesinde bulunan uygulama yazılımlarının nasıl geliştirildiği ve güvenlik durumları tespit edilmeye çalışılırken Yazılım Geliştirme ve Bilgi Güvenliği şubelerinden alınan bilgilere başvurulmuştur. Durum tespiti SWOT analizi yöntemi kullanılarak gerçekleştirilmiştir.

SWOT analizi bir projede ya da bir girişimde kurumun, tekniğin, sürecin, durumun veya kişinin;

- Güçlü (Strengths) yönlerini,
- Zayıf (Weaknesses) yönlerini,
- Fırsatları (Opportunities),
- Tehditleri (Threats),

saptamak için kullanılan stratejik bir tekniktir (SWOT analizi - Vikipedi 2016). Fırsat ve tehditler kurum içinden veya dışından kaynaklanabilmektedir. SWOT analizleri sayesinde açığa çıkmamış fırsatlar görülebilir ve bunlardan yararlanmak mümkün olabilmektedir. Ayrıca kurumun zayıf yönlerinin anlaşılarak tehditlerin kontrol edilmesini ve ortadan kaldırılmasını sağlamaktadır.

Şekil 23: SWOT Analizi



Bakanlığımız için güvenli uygulama yazılımı geliştirme bakış açısıyla gerçekleştirilen SWOT analizine göre güçlü yönler, zayıf yönler, fırsatlar ve tehditler aşağıda listelenmiştir.

Güçlü Yönler:

- Üst yönetim tarafından konu ile ilgili liderlik ve bağlılık gösterilmesi,
- Genel Müdürlüğümüzde görevler ayrılığı ilkesiyle uyumlu şube yapısının bulunması. (Yazılım Geliştirme Şubesi, Veri Yönetimi Şubesi, Yazılım Destek Şubesi vb.)
- Bakanlık bünyesinde Bilgi Güvenliği Şube Müdürlüğü'nün bulunması,
- Bilgi güvenliği ve siber güvenlik konularında farkındalığın artış gösteriyor olması,
- Personelin değişim isteğinden kaynaklanan yüksek motivasyonu,

- Genç ve dinamik personel kaynağı
- Kurumun hitap ettiği kitlenin belirli bir kesimden oluşması.

Zayıf Yönler:

- Bürokratik zorlukların olması,
- Konu ile ilgili yeterli sayıda ve yetkin personelin mevcut olmaması,
- Konu ile ilgili eğitim bütçesinin/ödeneğinin sağlanmamış olması,
- Bakanlık bünyesinde geliştirilmiş/edinilmiş çok sayıda uygulama yazılımı bulunması,
- Bakanlık bünyesinde Uygulama Yazılımları ve diğer BT bileşenleri edinimlerinin Birimler tarafından yapılıyor olmaya devam ediliyor olması,
- Bakanlık bünyesinde "sahipsiz/geliştiricisi bulunamayan" uygulama yazılımlarının bulunması.

Fırsatlar:

- Tez çalışması, 27001 kurulumu ve organizasyonel yapılanmaların doğru zamanda gerçekleşmiş olması,
- ISO 27001 kurulumunun yapılması ve sürekli iyileştirme kapsamında Yazılım Geliştirme faaliyetlerine yönelik olarak tez konusu kapsamında da bulunan birçok husus ile ilgili birçok düzeltici faaliyetlere başlanmış olması,
- Diğer kamu kurumlarının deneyimleri,
- Teknolojinin gelişmesiyle konuyla ilgili kaynakların erişilebilirliğinin kolaylaşması.

Tehditler:

- Yönetimin sıklıkla değişmesi,
- Kullanıcı taleplerinin sürekli değişmesi,
- Kamu kurumu olması nedeniyle saldırılara hedef olma olasılığı,
- Bakanlık bazında kurumsal süreçlerin tanımlı olmaması,
- Bakanlık genelinde değişime açık olunmaması,

SWOT analizi ile elde edilen genel bakışa ek olarak kurumumuzda geliştirilen ve dışarıdan temin edilen uygulama yazılımları ile ilgili diğer tespitler aşağıdaki gibidir;

Kurum bünyesinde geliştirilen ve dışarıdan temin edilen uygulama yazılımları için ortak tespitler:

- ✓ Geliştirilme ve idamesi YTDB sorumluluğunda olan uygulama yazılımlarına ait uygun yazılı bilgiler mevcut olup arzu edilen seviyede değildir.
- ✓ Mevcut durumda Geliştirme, Test ve Uygulama ortamları ayrıştırılmış değildir. Ayrıştırılması için çalışmalar sürdürülmektedir.
- ✓ Sürüm kontrol sistemi kullanılmaktadır.
- ✓ Merkezi bir iz kayıt konsolidasyon sistemi mevcut bulunmakta ancak tüm bakanlık uygulamaları bu sisteme kayıt göndermemektedir.
- ✓ Yazılım geliştirme aracı olarak kurum bünyesinde bulunan .Net Framework ve Java kullanılmaktadır.
    - o Kurum politikası olarak geliştirilecek tüm yeni yazılımlarda .Net Framework kullanılması hedeflenmektedir.

Kurum bünyesinde geliştirilen uygulama yazılımları:

- ✓ Geliştirilecek uygulama yazılımı ile ilgili talep geldikten sonra uygulama sahibi ile yazılımcı bir araya gelerek analiz yapmaktadır.
- ✓ Yazılımcı, analiz bittikten sonra tasarım yaparak kodlama aşamasına geçmektedir.
- ✓ Geliştirilen yazılımlar için yazılım yaşam döngüsü modeli olarak çevik yöntem tercih edilmektedir.
- ✓ Güvenli uygulama yazılımı geliştirme kriterlerinden genel kabul görmüş kurallar dikkate alınmaktadır ancak uygulanan belirli bir metodoloji uygulanmamaktadır.
- ✓ Süreç bazlı yazılım geliştirme ile ilgili çalışmalar başlangıç aşamasındadır.

- ✓ Yazılım geliştirme aşamalarının dokümante edilmesi ile ilgili çalışmalara başlanmıştır.

Dışarıdan temin edilen uygulama yazılımları:

- ✓ Yayınlanan 2014-14 sayılı Bilişim ve Coğrafi Bilgi Sistemi Projeleri Yürütme ve Koordinasyon Esasları Genelgesi ile dışarıdan temin ile alınacak uygulama yazılımlarının teknik şartnamelerinin CBS Genel Müdürlüğü tarafından kontrol edilmesi zorunlu hale gelmiştir.
- ✓ 2015 yılından sonra teknik şartnamelere güvenlik esasları ile ilgili maddeler eklenmeye başlanmıştır.
- ✓ 2017 yılında Yazılım Geliştirme ve Birlikte Çalışma Esasları Prosedürü hazırlanmış ve bu prosedüre uyulmasıyla ilgili bir maddenin teknik şartnamelere eklenmesi planlanmaktadır.
    - o Bu prosedürde güvenlik konusu bir başlık olarak yer almaktadır. Bu başlık altında uygulama yazılımı geliştirildikten sonra yapılması beklenen güvenlik testleri, testi yapacak kişilerin nitelikleri, test sonuçlarının değerlendirilmesi gibi konular ele alınmaktadır.
- ✓ Dışarıdan edinilen uygulama yazılımlarında kamu ihale mevzuatına uygun olarak Şelale Modeli tercih edilmektedir.
- ✓ Yüklenicilerin yazılım tasarımını Bakanlık Bilgi Güvenliği Politikalarına (Kullanıcı Erişim Yönetimi, Parola, Şifreleme vb.) uygun şekilde gerçekleştirmesi beklenmektedir.
- ✓ Bakanlık Bilgi Güvenlik Politikaları gereği Yükleniciye Uzaktan Erişim (VPN) sağlanmamaktadır.

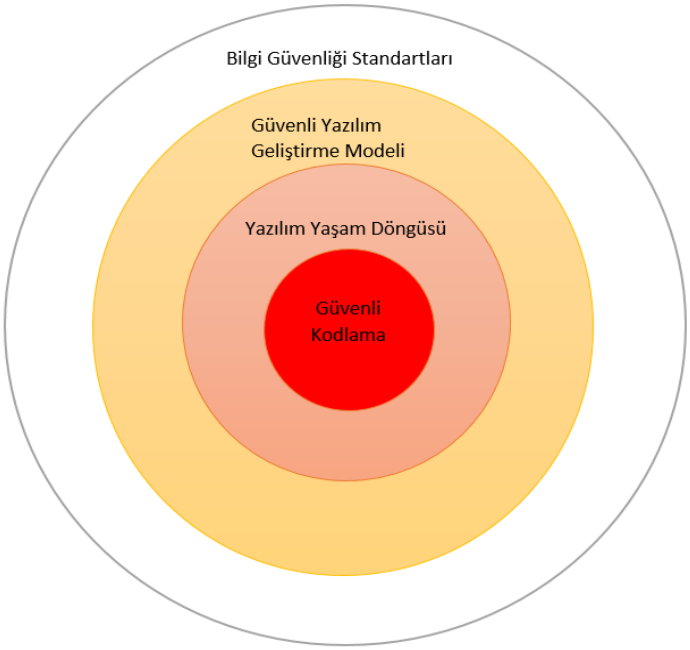
5.2. Bakanlık İçin Öneriler ve Yol Haritası

Hazırlanan bu tez kapsamında yapılan araştırmalar ve çalışmalar sonucunda güvenli uygulama yazılımı geliştirme esasları belirlenmeye çalışılmıştır. Elde edilen verilere güvenli uygulama yazılımı geliştirebilmenin temel şartları aşağıda verildiği gibidir;

- ✓ Güvenlik açıklarını önlemek, tespit etmek ve düzeltmek için tüm kurum bünyesinde geliştirilen ve dışarıdan temin edinilen yazılımlarda yazılım yaşam döngüsü modeli seçilmeli ve benimsenmelidir.
    - Böylelikle yazılımın kalite güvencesi sağlanacaktır.
- ✓ Yazılım geliştirme aşamasında kullanılacak araç seti belirlenmiş olmalıdır. Bu araçların seçimi geliştirme süreçlerinde uygulanması gereken güvenlik faaliyetlerini etkilemektedir.
    - Araç seçimi yapılmadığı durumlarda proje yönetimi dolayısıyla güvenlik esaslarının sağlanması zorlaşmaktadır.
    - Kullanılacak olan programlama dili, veri tabanı türü, sürüm takip aracı, vb. belirlenmiş olmalıdır.
- ✓ Uygulama geliştirme güvenliği hususunda farkındalığı artıracak çalışmalar desteklenmelidir.
- ✓ Proje bazlı faaliyet yöntemine geçilerek yazılım projeleri merkezi olarak takip edilmelidir.
- ✓ Yazılım geliştirme sırasında kullanılmak üzere güvenlik hususlarını içeren bir kontrol listesi bulunmalıdır.
- ✓ Kurum yapısına uygun güvenli yazılım geliştirme modeli seçilmeli ve benimsenmelidir.
- ✓ Güvenli Yazılım Geliştirme Politikası oluşturulmalı ve uygulamaya geçirilmelidir.
- ✓ Olgunlaşma gereksinim aşamasından başlayarak tüm yazılım yaşam döngüsüne dâhil olmalıdır.
    - Kurumun yapısına en uygun olgunluk modeli seçilmeli ve benimsenmelidir.

Bakanlık için önerilen metodolojide kullanılması önerilen modelin güvenlik faaliyetleriyle ilgili rehberlik edebilecek, yazılım geliştiriciler için yeterli bilgiyi sağlayacak ve iyi tanımlanmış olması hedeflenmiştir. Oluşturulması hedeflenen yapı Şekil 25'te görselleştirilmiştir.

Şekil 24: Oluşturulması hedeflenen güvenli yazılım geliştirme metodolojisi



5.2.1. Çevre ve Şehircilik Bakanlığı için Önerilen Güvenli Yazılım Geliştirme Metodolojisi

Çevre ve Şehircilik Bakanlığının kurumsal altyapısı ve mevcut durumu incelendikten sonra kurumda Güvenli Uygulama Yazılımı Geliştirme metodolojisi için önerilen çözüm önerisinin aşağıdaki özelliklere sahip olması gerektiği sonucuna varılmıştır:

- ✓ Hızlı ve kolay kullanıma geçirilebilmelidir.
- ✓ Tam bir yazılım yaşam döngüsü entegrasyonu sağlamalıdır.
- ✓ Kurumsal güvenlik altyapısıyla uyumlu olmalıdır.
- ✓ Güvenlik açıklarını ve riskleri azaltmalıdır.
- ✓ Kimlik doğrulama, şifreleme, denetim vb. temel güvenlik konularını ele almalıdır.
- ✓ Bilgi güvenliği açısından öngörülen dayanıklılık seviyesine sahip olmalıdır.

Güvenli yazılım geliştirme metodolojisinin ilk adımı yazılım yaşam döngüsü modeli seçimi ve süreçlerinin entegrasyonudur. Bakanlık tarafından seçilmesi gereken yazılım yaşam döngüsü modeli ve uygulanması başka bir uzmanlık tezi konusu olduğu için bu çalışma kapsamında ayrıntıya girilmemiştir. Bakanlık tarafından belirlenmiş olduğu ve uygulandığı kabul

edilmiştir. Önerilen güvenli yazılım geliştirme metodolojisi yazılım yaşam döngüsü ve kullanılan platformlardan bağımsızdır.

Önerilen metodolojinin ikinci adımı ise organizasyonel güvenlik ihtiyaçlarını ve bilgi güvenliği kontrollerinin sağlanmış olmasıdır. Bu amaçla seçili yazılım yaşam döngüsü ile entegre edilmesi önerilen ISO 27001 standardı tüm dünyada kabul görmüş ve bilgi güvenliği için kurumsal çerçeve sağlayan bir standarttır. Standartta ele alınan önemli hususlardan biri de yazılım geliştirme süreçlerinde güvenliğinin sağlanması ve buna ilişkin olarak yazılım geliştirme politikasının oluşturulmasıdır. ISO 27001 tarafından önerilen başlıca kontroller ve bu kontroller kapsamında yazılım geliştirme süreçlerinde gerçekleştirilmesi gereken hususlar Tablo 16' da gösterilmiştir.

Tablo 16: ISO 27001 Bilgi Güvenliği Kontrolleri

A.14 Sistem Temini, Geliştirme Ve Bakımı		
A.14.1 Bilgi Sistemlerinin Güvenlik Gereksinimleri		
Amaç: Bilgi güvenliğinin, bilgi sistemlerinin tüm yaşam döngüsü boyunca dâhili bir parçası olmasını sağlamak. Bu aynı zamanda halka açık ağlar üzerinden hizmet sağlayan bilgi sistemleri gereksinimlerini de içerir.		
A.14.1.1	Bilgi güvenliği gereksinimleri analizi ve belirtimi	*Kontrol* Bilgi güvenliği ile ilgili gereksinimler, yeni bilgi sistemleri gereksinimlerine veya var olan bilgi sistemlerinin iyileştirmelerine dâhil edilmelidir.
A.14.1.2	Halka açık ağlardaki uygulama hizmetlerinin güvenliğinin sağlanması	*Kontrol* Halka açık ağlar üzerinden geçen uygulama hizmetlerindeki bilgi, hileli faaliyetlerden, sözleşme ihtilafından ve yetkisiz ifşadan ve değiştirmeden korunmalıdır.
A.14.1.3	Uygulama hizmet işlemlerinin korunması	*Kontrol* Uygulama hizmet işlemlerindeki bilgi eksik iletim, yanlış yönlendirme, yetkisiz mesaj değiştirme, yetkisiz ifşayı, yetkisiz mesaj çoğaltma ya da mesajı yeniden oluşturmayı önlemek için korunmalıdır.
A.14.2 Geliştirme Ve Destek Süreçlerinde Güvenlik		
Amaç: Bilgi güvenliğinin bilgi sistemleri geliştirme yaşam döngüsü içerisinde tasarlanıyor ve uygulanıyor olmasını sağlamak		
A.14.2.1	Güvenli geliştirme politikası	*Kontrol* Yazılım ve sistemlerin geliştirme kuralları belirlenmeli ve kuruluş içerisindeki geliştirmelere uygulanmalıdır.

A.14.2.2	Sistem değişiklik kontrolü prosedürleri	*Kontrol* Geliştirme yaşam döngüsü içerisindeki sistem değişiklikleri resmi değişiklik kontrol prosedürlerinin kullanımı ile kontrol edilmelidir.
A.14.2.3	İşletim platformu değişikliklerden sonra uygulamaların teknik gözden geçirmesi	*Kontrol* İşletim platformları değiştirildiğinde, kurumsal işlemlere ya da güvenliğe hiçbir kötü etkisi olmamasını sağlamak amacıyla iş için kritik uygulamalar gözden geçirilmeli ve test edilmelidir.
A.14.2.4	Yazılım paketlerindeki değişikliklerdeki kısıtlamalar	*Kontrol* Yazılım paketlerine yapılacak değişiklikler, gerek duyulanlar hariç önlenmeli ve tüm değişiklikler sıkı bir biçimde kontrol edilmelidir.
A.14.2.5	Güvenli sistem mühendisliği prensipleri	*Kontrol* Güvenli sistem mühendisliği prensipleri belirlenmeli, yazılı hale getirilmeli ve tüm bilgi sistemi uygulama çalışmalarına uygulanmalıdır.
A.14.2.6	Güvenli geliştirme ortamı	*Kontrol* Kuruluşlar tüm sistem geliştirme yaşam döngüsünü kapsayan sistem geliştirme ve bütünleştirme girişimleri için güvenli geliştirme ortamları kurmalı ve uygun bir şekilde korumalıdır.
A.14.2.7	Dışarıdan sağlanan geliştirme	*Kontrol* Kuruluş dışarıdan sağlanan sistem geliştirme faaliyetini denetlemeli ve izlemelidir.
A.14.2.8	Sistem güvenlik testi	*Kontrol* Güvenlik işlevselliğinin test edilmesi, geliştirme süresince gerçekleştirilmelidir.
A.14.2.9	Sistem kabul testi	*Kontrol* Kabul test programları ve ilgili kriterler, yeni bilgi sistemleri, yükseltmeleri ve yeni versiyonları için belirlenmelidir.
A.14.3 Test Verisi		
Amaç: Test için kullanılan verinin korunmasını sağlamak.		
A.14.3.1	Test verisinin korunması	*Kontrol* Test verisi dikkatli bir şekilde seçilmeli, korunmalı ve kontrol edilmelidir.

(ISO 2013a)

Bu kontrollere ek olarak ISO 27002 de belirtilmiş olan “6.1.5 Proje yönetiminde bilgi güvenliği” kontrolünün de sağlanmış olmasının faydalı olacağı düşünülmektedir. İlgili kontrol Tablo 17’de verilmiştir.

Tablo 17: ISO 27002 Kontrolü

6.1.5 Proje yönetiminde bilgi güvenliği
Kontrol Proje yönetiminde, proje türüne bakılmaksızın bilgi güvenliği ele alınmalıdır.
Uygulama kılavuzu Bilgi güvenliği, bilgi güvenliği risklerinin tanımlanması ve bir projenin parçası olarak ele alınmasını sağlamak için kuruluşun proje yönetimi yöntemine/yöntemlerine entegre edilmelidir. a) Bilgi güvenliği amaçlarının proje amaçlarına dâhil olması, b) Bilgi güvenliği risk değerlendirmesinin gerekli kontrollerin tanımlanması için projenin erken bir aşamasında yapılması, c) Bilgi güvenliğinin uygulanan proje metodolojisinin her aşamasının bir parçası olması.

(ISO 2013b)

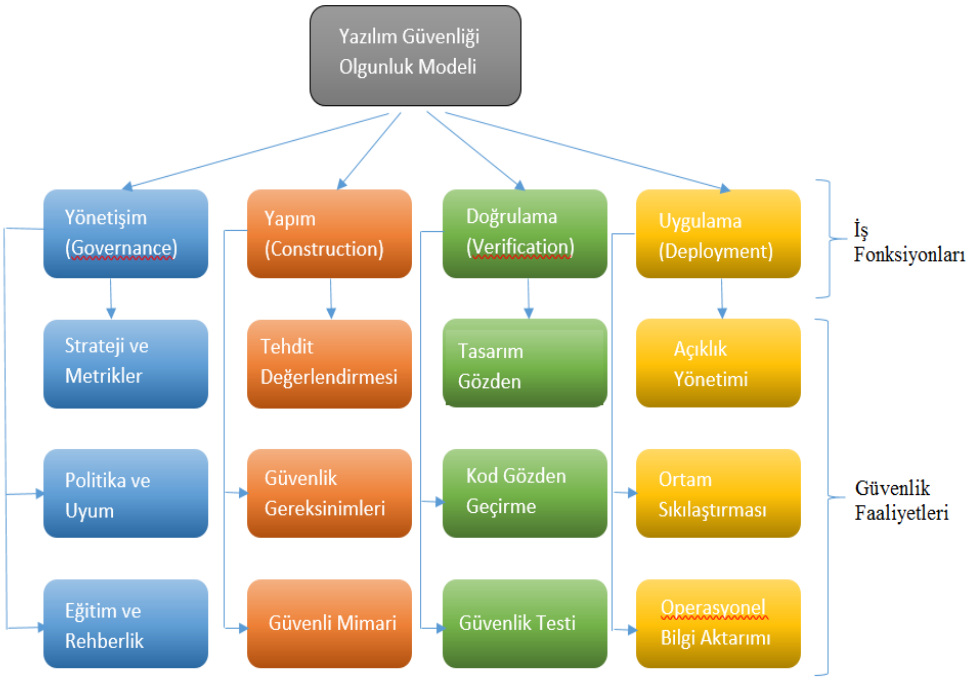
Güvenli yazılım geliştirme metodolojisinin üçüncü adımında kuruma özel Güvenli Yazılım Geliştirme Politikası oluşturulmalı ve tüm uygulama yazılımlarında benimsenmelidir. Ayrıca bu politika kapsamında oluşturulacak Güvenli Yazılım Geliştirme Kontrol Listesinde belirtilen hususlar geliştirme süreçlerinde dikkate alınmalıdır. Çevre ve Şehircilik Bakanlığı için önerilen Güvenli Yazılım Geliştirme Politikası ve Güvenli Yazılım Geliştirme Kontrol Listesi ekte yer almaktadır. (EK -1, EK-2)

Önerilen metodolojinin son adımı SAMM modelinin kullanıma geçirilmesidir. Bu sayede yazılım yaşam döngüsü süreçlerine güvenlik bakış açısı entegre edilmiş olacaktır. Örneğin yazılım yaşam döngülerinin doğal adımlarından olan gereksinim ve kullanım durumu (use-case) analizlerine yazılım projesine özgü güvenlik ihtiyaçları da dâhil olacaktır. Kalite olgusu geliştirilecek, güvenlik unsuru tüm sistemlerle bütünleştirilmiş olacaktır. SAMM kullanımı kolay ve anlaşılabilir bir modeldir. Katı kuralları olmadığı için var olan sistemlere uyumluluğu yüksektir. Önerdiği güvenlik

faaliyetlerinden birkaçı veya hepsi uygulamaya geçirilerek uygulanabilmektedir. Ayrıca Risk yönetimi konusundaki sadeliği kullanıcılara kolaylık sağlamaktadır.

SAMM dört iş fonksiyonu ve on iki güvenlik faaliyetinden oluşmaktadır. SAMM genel yapısı Şekil-26'de gösterildiği gibidir.

Şekil 25: SAMM Genel Yapısı



(Deleersnyder vd. 2009)

1. **Yönetim (Governance):**
    - Strateji&Metrikler:
        - Kurum içinde bir yazılım güvenliği yönetim süreci oluşturulmalıdır.
        - Uygulamaların ve bilgilerin risk sınıflandırmasını yapılandırılmalıdır.
        - Oluşturulan risk sınıflarının güvenlik hedefleri belirlenmelidir.

- Politika&Uyumluluk:
    - Yasal gereksinimler temel alınarak kurum için güvenlik ve uyumluluk kontrolleri oluşturulmalıdır.
- Eğitim&Destek:
    - Yazılım geliştirme sürecinde yer alan personellerin (proje yöneticileri, yazılım mimarları, geliştiriciler vb.) güvenlik konusundaki yetkinlikleri eğitimler ve teknik destek yoluyla artırılmalıdır.

2. Yapım:

- Tehdit Değerlendirme:
    - Yazılımlar etkin risk yönetimi gerçekleştirilmelidir.
    - Yazılımların karşı karşıya kalabilecekleri saldırılar analiz edilmeli ve güvenlik öncelikleri belirlenmelidir.
- Güvenlik Gereksinimleri:
    - En iyi uygulamalar dikkate alınarak iş akışı esnasında uygulanması gereken güvenlik gereksinimleri belirlenmelidir.
- Güvenli Mimari:
    - Yazılım tasarımı esnasında güvenlik tasarım modellerinden (security design patterns) ve de güvenli mimari ilkelerinden (secure architecture principles) faydalanılarak mimari tasarım esnasında güvenlik hususları sistem ile bütünleştirilmelidir.

3. Doğrulama:

- Tasarım Denetimi:
    - Yapım iş fonksiyonu esnasında oluşturulan yazılım mimarisi güvenlik açısından denetlenmelidir.
    - Yapım iş fonksiyonu esnasında güvenlik gereksinimlerinin tasarlanan mimari tarafından gerçekleştirilip gerçekleştirilmediğini kontrol edilmelidir.

- Kod Denetimi:
    - Yazılım kodunda olabilecek güvenlik açıklarını tespit edebilmek için kod gözden geçirilmelidir.
    - Kod denetimi, bir kontrol listesi oluşturup manuel olarak ya da otomatik kod denetim araçları ile gerçekleştirilebilir.
- Güvenlik Testi:
    - Yazılımdaki güvenlik açıklarını tespit edebilmek için güvenlik testleri yapılmalıdır.
    - Otomatik test araçları kullanılabilir.

4. Kurulum:

- Güvenlik Açığı Yönetimi:
    - Yazılımın kullanımı esnasında ortaya çıkan güvenlik açıklarına karşı gerekli (açığı inceleme, yama çıkarma gibi) adımları içeren bir açık yönetim süreci oluşturulmalıdır.
- Ortam Sıkılaştırma:
    - Yazılımların kurulu ya da etkileşimde olduğu altyapı bileşenlerinin (işletim sistemi, uygulama sunucusu, veri tabanı sunucusu gibi) güvenlik ayarları arttırılmalıdır.
    - Bu ortamlar dayanıklı hale getirilmelidir.
- Operasyonel Bilgi Aktarımı:
    - Güvenlikle ilgili kritik ayarların kurulum ve kullanım esnasında dikkate alınmasını sağlamak için yazılım geliştiriciler ile kullanıcılar arasında iletişim sağlanmalıdır.

Yukarıda verilmiş olan on iki güvenlik fonksiyonunda belirtilen önerilerin gerçekleştirilmesi ile belirli bir seviyede güvenlik sağlanmış olmaktadır. Yazılımlarda güvenlik açısından olgunluğu ölçebilmek ve değerlendirme yapabilmek için değerlendirme çizelgeleri kullanılmalıdır. Elde edilen skorlar analiz edilmeli ve elde edilen sonuçlar izlenmesi gereken yol haritasını belirlemek için kullanılmalıdır. Değerlendirme çizelgesi, skor tablosu ve eylem planı ile ayrıntılı bilgi aşağıda verilmiştir.

Değerlendirme Çizelgesi (Assessment Worksheet)

Uygulama yazılımı geliştirilmesi esnasında gerçekleştirilen güvenlik faaliyetlerini değerlendirip, SAMM ile uygulama yazılımının güvenlik seviyesini belirleyip güvenlik açısından durumun görülebilmesi için değerlendirme çizelgeleri kullanılmaktadır. Değerlendirme Çizelgesinde, her iş fonksiyonu (toplam 4 adet) ve güvenlik faaliyeti (toplam 12 adet) için belirlenmiş sorular bulunmaktadır. Bu sorular da 3 farklı olgunluk seviyesini belirlemek için gruplanmıştır. 1. olgunluk seviyesini belirlemek için sorulan tüm sorulara "Evet" yanıtı alınıyorsa o aktivite için birinci seviyeye ulaşılmış demektir. Aynı şekilde 2. ve 3. seviye sorularına verilen "Evet" seviye tespiti için kullanılmaktadır. Bu seviyelere ek olarak ara seviyeler de (0+, 1+,2+ ve 3+) mevcuttur. Bir seviyedeki soruların tamamı değil de bazıları "Evet" şeklinde cevaplanmışsa bu ara seviyelere ulaşıldığının göstergesidir.

Skor Tablosu (Scorecards)

Her bir güvenlik faaliyeti için değerlendirme çizelgesi kullanılarak ölçülen güvenlik seviyelerini kayıt altına almak, analiz etmek ve karşılaştırma yapabilmek için skor tabloları kullanılmaktadır.

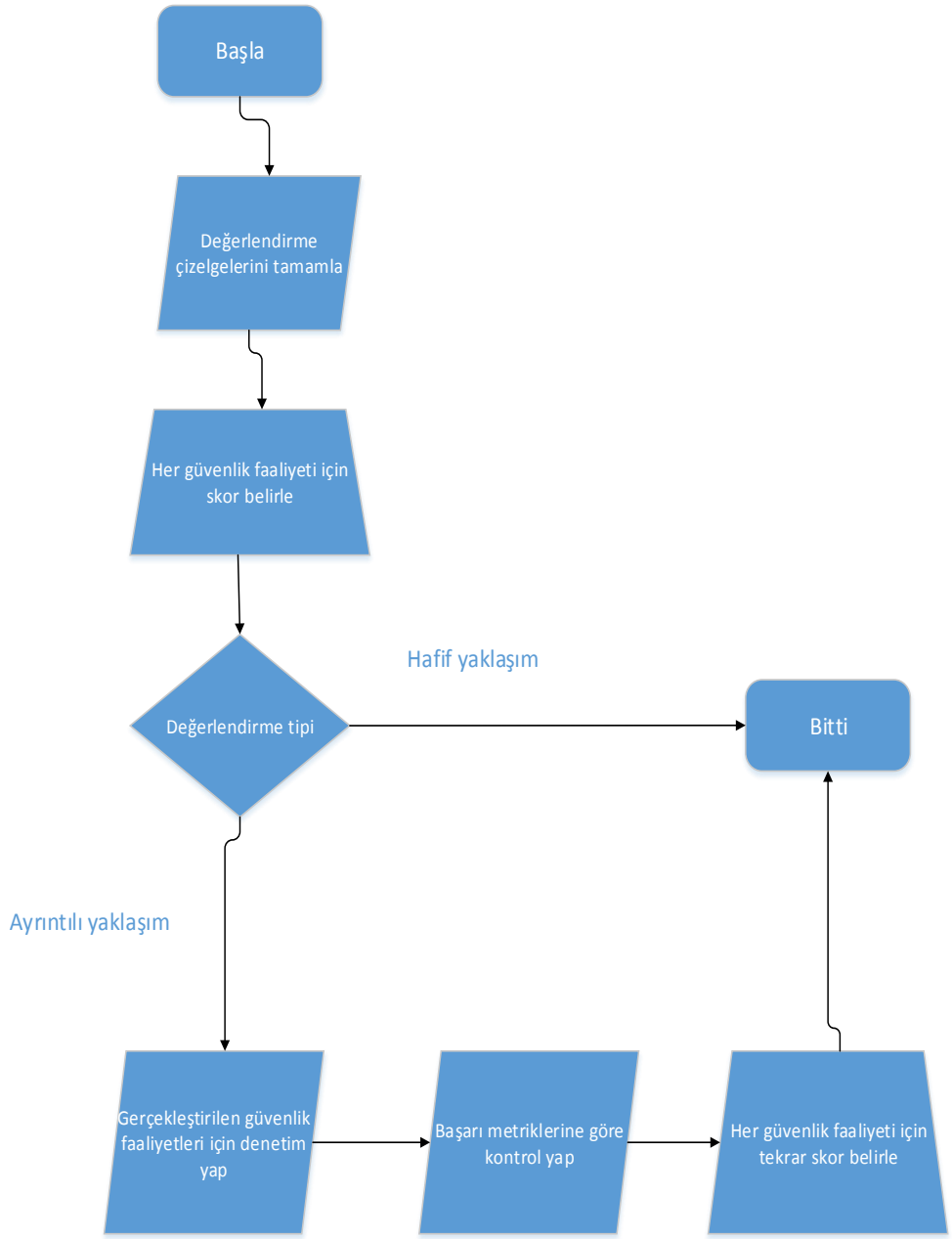
Eylem Planı Şablonu (Roadmap Template)

Eylem planı şablonu ile yazılım geliştirme sürecinde güvenlikle ilgili gerçekleştirilmesi gereken kontrollerin ve ulaşılması hedeflenen güvenlik olgunluk seviyelerinin yönetilmesi sağlanmaktadır. Bu şablonu kullanmak için değerlendirme çizelgeleri kullanılarak olgunluk seviyesi belirlenir. Daha sonra eylem planı şablonu işlenir ve her bir güvenlik faaliyeti için zamanla ulaşılmak istenen seviyeler ve hedefler belirlenerek şablona işlenmelidir. Oluşturulan bu şablon sayesinde hedeflenen güvenlik faaliyetleri ve planları takip edebilir ve yönetebilir. Bu şablonda her bir güvenlik faaliyeti için hedeflenen olgunluk seviyeleri grafiksel olarak da gösterilmektedir.

Bu süreçler için takip edilmesi gereken akış diyagramı Şekil 27'de gösterildiği gibidir. SAMM yaklaşımı işletilirken iki türlü yaklaşım benimsenebilmektedir. Bunlar hafif ve ayrıntılı yaklaşımlardır. Bu model kullanılmaya başlandığı ilk zamanlarda hafif yaklaşımla değerlendirmeler yapılması uygulamada kolaylık sağlayacaktır. Modelin kullanım alanı genişledikten sonra ise ikinci yaklaşım olan ayrıntılı yaklaşım kullanılarak

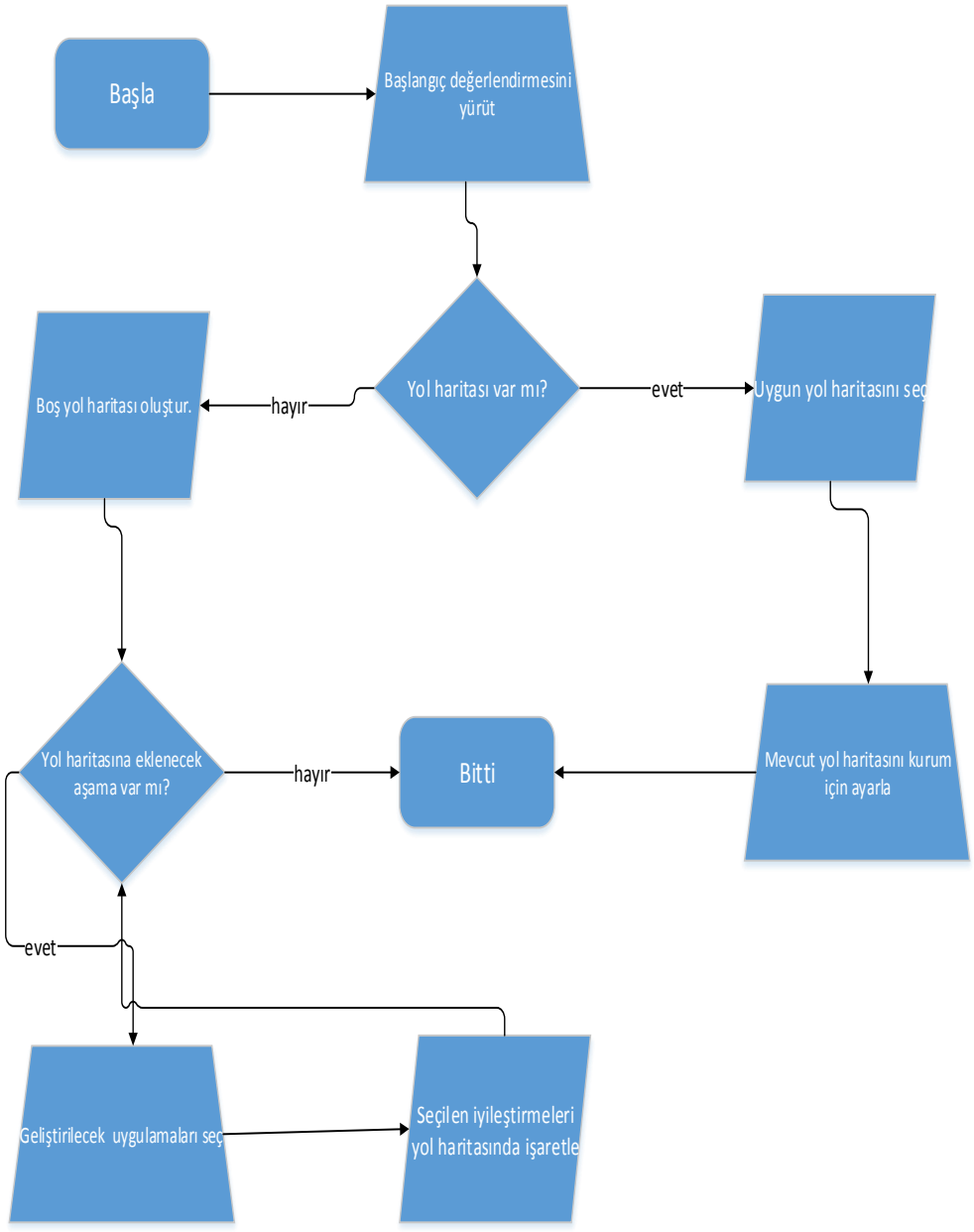
başarı metrikleri de sisteme entegre edilmelidir. Başarı Metrikleri, bir organizasyonun verilen seviyede performans gösterdiğini kontrol etmek için kullanılabilecek örnek ölçümleri belirtir. Veri toplama ve yönetimi, her organizasyonun seçimine bırakılır, ancak önerilen veri kaynakları ve eşikleri SAMM tarafından sağlanmaktadır. Başarı metrikleri sistem ile bütünleştirildikten sonra ideal durum ile mevcut durum kıyaslaması yapılmalı ve fark analizi yapılmalıdır.

Şekil 26 : Değerlendirme çizelgeleri için akış diyagramı



Fark analizi sonucunda elde edilen bulgular kullanılarak geliştirilmesi gereken konular belirlenmelidir. Geliştirmeler sonucunda tekrar durum değerlendirmesi yapılmalı ve skorlar oluşturulmalıdır. Elde edilen skorlara göre belirlenen hedeflere ulaşmak için izlenmesi gereken yol haritası belirlenmelidir. SAMM'i işletmek için izlenmesi gereken yol Şekil 28'de verilen akış diyagramı ile gösterilmiştir. SAMM modelinde bulunan güvenlik faaliyetleri yinelemeli olarak değerlendirilmelidir. Böylece hedeflenen güvenlik olgunluk seviyesine ulaşmak mümkün olacaktır.

Şekil 27: SAMM işletimi akış diyagramı



SONUÇLAR VE TARTIŞMA

Ülkemizde ve dünyada teknolojinin ilerlemesiyle birlikte kamu tarafından sunulan hizmetler dönüşüm geçirmiş ve E-Devlet uygulamaları örneğinde olduğu gibi elektronik ortamda vatandaşlara uygulama yazılımları aracılığıyla sunulmaktadır. Bu hizmetler kişi veya kurumlara ait özel verileri içermektedir. Dolayısıyla bu verilerin gizliliğinin, bütünlüğünün korunması ve erişilebilirliğinin sağlanması yadsınamayacak bir ihtiyaç halini almıştır. Bu ihtiyaç 2016-2019 Ulusal Siber Güvenlik Eylem Planında da yer almış ve gerekli önlemlerin alınması için çalışmalar yapılması hedef olarak belirlenmiştir.

Bu kapsamda hazırlanan bu tez çalışmasında Bilgi Güvenliği ve Siber Güvenlik konuları açıklanmış, Güvenli Uygulama Yazılımı Geliştirme metodolojileri araştırılmıştır. Ayrıca "Güvenli Uygulama Yazılımı Geliştirme" konusunda yapılan anket çalışmasıyla yazılım geliştiricilerin konu ile ilgili düşünceleri öğrenilmeye çalışılmıştır. Bunlara ek olarak çalışma konusunun Türkiye'deki durumuyla ilgili bilgi sahibi olabilmek için saha araştırması yapılmıştır. Saha araştırmasında bazı kamu kurumları ve sektörden firmalar ziyaret edilmiş, güvenli uygulama yazılımı konusunu ele alış biçimleri araştırılmıştır. Saha çalışması kapsamında Türkiye Atom enerjisi Kurumu, T.C. Sosyal Güvenlik Kurumu, T.C. Başbakanlık Afet ve Acil Durum Yönetimi Başkanlığı, T.C. Posta ve Telgraf Teşkilatı Genel Müdürlüğü ve Türksat Uydu Haberleşme Kablo TV ve İşletme A.Ş. ile güvenli yazılım geliştirme konusunda yaptıkları çalışmalar ile ilgili ayrıntılı bilgi alabilmek için yerlerinde görüşmeler yapılmıştır. Ayrıca Gümrük ve Ticaret Bakanlığı ve Türkiye Bilimsel Ve Teknolojik Araştırma Kurumu Ulusal Akademik Ağ ve Bilgi Merkezinden konu ile ilgili çalışmaları hakkında bilgi alınmıştır.

Yapılan saha çalışmaları sonucunda kamu kurumlarında güvenli yazılım geliştirme konusunda sistematik bir yaklaşımın henüz oluşturulamadığı ancak bu konuda çalışmaların yürütülmeye başlandığı gözlemlenmiştir. Bu çalışmalar kapsamında ISO 27001 sertifikası alınmış veya alınma aşamasında olduğu görülmüştür. Ayrıca kurumların çoğunda yazılım geliştirme politikası bulunmasına rağmen güvenlik odaklı bir politika bulunmadığı gözlemlenmiştir. Kamu kurumlarında görülen bir diğer durumda

güvenli uygulama yazılımı geliştirme konusunda yetkin personelin bulunmadığıdır. Görüşülen kurumlarda bu durumu çözebilmek için personele konu ile ilgili hizmet içi eğitimler verilmektedir.

Özel sektörde büyük firmalarda kalite odaklı ürün geliştirme yaklaşımı bulunduğu için yazılım geliştirme konusu daha sistematik olarak ele alınmaktadır. Güncel yaklaşımlar takip edilmekte gerekli değişiklikler daha kolaylıkla uygulanabilmektedir. Özel sektörde güvenli yazılım geliştirme politikalarının oluşturulduğu ve uygulandığı gözlemlenmiştir. Ayrıca personele düzenli aralıklarla eğitimler verilerek yetkinliklerini geliştirmeleri sağlanmaktadır. Küçük çaplı yazılım firmalarında ise daha çok kamuya benzer bir yapı bulunmaktadır.

Ülkemizde ve dünyada siber güvenlik konusu nispeten yeni bir konu olduğu için güvenli uygulama yazılımı geliştirme konusunda olgunluğa ulaşabilmek üzerinde durulması gereken birçok nokta bulunduğu sonucuna varılmıştır. Elde edilen bulguların değerlendirilmesiyle ortaya çıkan sonuçlar aşağıdaki gibidir;

- ✓ Siber güvenliğin sağlanabilmesi için atılması gereken önemli adımlardan biri olan "Güvenli uygulama yazılımı geliştirme" konusu ciddi bir şekilde ele alınmalıdır.
- ✓ Güvenli uygulama yazılımı geliştirmek için konuya sistematik olarak yaklaşılmalı ve dünya çapında kabul görmüş modeller benimsenmelidir.
- ✓ Alınacak yeni teknolojilerin özellikleri araştırılırken güvenlik konusuna önem verilmeli ve alımlar bu yönde gerçekleştirilmelidir.
- ✓ Bakanlık için Güvenli Uygulama Yazılımı Geliştirme Politikası hazırlanmalı ve kullanıma geçirilmelidir.
- ✓ Konu ile ilgili yetkin insan kaynağı yetiştirilmelidir.
- ✓ Konu ile ilgili gelişmeler takip edilerek güncellik sağlanmalıdır.

Bu çalışma kurumun mevcut durumu göz önünde bulundurularak yapılmıştır. Kurumun yazılım geliştirme ve güvenli yazılım geliştirme konularında olgunluk seviyelerini zaman içinde arttıracağı öngörülmektedir.

Olgunluk seviyelerinde gelişmeyi sağlamak amacıyla konuyu daha geniş çaplı olarak ele alan Ortak Kriterler yaklaşımının benimsenmesinin iyi bir uygulama olacağı düşünülmektedir.

KAYNAKÇA

"Açık anahtarlı şifreleme". 2015. https://tr.wikipedia.org/wiki/Açık_anahtarlı_şifreleme.

Alkan, Mustafa; Akkaya Mariye Umay; İnceefe Mehmet Ali; Kesen Mustafa; vd. 2013. *Siber Güvenlik ve Standardizasyon Kitapçığı*.

"Atik yazılım geliştirme - Vikipedi". 2016. https://tr.wikipedia.org/wiki/Atik_yazılım_geliştirme (02 Nisan 2017).

Bakır, Emre. 2012. "5. Boyutta Savaş: Siber Savaşlar - I - Ulusal Bilgi Güvenliği Kapısı". *Tübitak Bilgem*. https://www.bilgiguvenligi.gov.tr/siber-savunma/5.-boyutta-savas-siber-savaslar-i.html (02 Nisan 2017).

Barış Can Kaşıkçı. 2009. "Yazılım Sektörünün Geleceği ve Ülkemizin Konumu | e-bergi". *Odtü Bilgisayar Topluluğu Elektronik Dergisi*. http://e-bergi.com/y/Yazilim-Sektorunun-Gelecegi-ve-Ülkemizin-Konumu (02 Nisan 2017).

Beydağlı, Erkut Kara, Mehmet Bahşi, Hayrettin Alparslan, Erdem. 2009. "Güvenli Yazılım Geliştirme Modelleri ve Ortak Kriterler Standardı". Içinde *4. Ulusal Yazılım Mühendisliği Sempozyumu, ,* 11–17.

"Bilgi Sistemleri Güvenliği Tatbikatı BOME 2008 - Ulusal Bilgi Güvenliği Kapısı". https://www.bilgiguvenligi.gov.tr/raporlar-kategorisi/bilgi-sistemleri-guvenligi-tatbikati-bome-2008.html (02 Nisan 2017).

Bozer, Recep. 2013. "TİCARET SiCiLi MÜDÜRLÜĞÜ ÇALIŞANLARININ MERSİS UYGULAMASI HAKKINDAKİ GÖRÜŞLERİNİN BELİRLENMESİ". Gümrük ve Ticaret Bakanlığı.

Büyüköztürk, ş. Kılıç Çakmak, E. Akgün, Ö. 2010. *Bilimsel Araştırma Yöntemleri*. 6. baskı. Pegem Akademi.

C. C., Micheal, van Wyk, Ken, Radosevich. 2005. "Risk-Based and Functional Security Testing". *Cigital*.

Canbek, Gürol. 2016. "Türkiye'nin Küresel Siber Güvenlik Göstergesi". *HAVELSAN Aylık Siber Güvenlik Bülteni* (7): 12–13.

Canbek, Gürol, ve Şeref Sağıroğlu. 2006. "Bilgi, Bilgi Güvenliği ve Süreçleri Üzerine Bir İnceleme {A Review on Information, Information Security and Security Processes}". *Politeknik Dergisi* 9(3): 165–74. http://www.politeknik.gazi.edu.tr/index.php/PLT/article/download/299/295.

"CMMI - Vikipedi". https://tr.wikipedia.org/wiki/CMMI (03 Nisan 2017).

"CMMI İlkeleri ve Değerleri". 2013. https://msdn.microsoft.com/tr-tr/library/hh765978(v=vs.120).aspx (03 Nisan 2017).

"COBIT 4.1 Executive Summary". http://www.isaca.org/knowledge-center/cobit/documents/cobit4.pdf (03 Nisan 2017).

"Cryptographic Storage Cheat Sheet - OWASP". 2017. https://www.owasp.org/index.php/Cryptographic_Storage_Cheat_Sheet (02 Nisan 2017).

Deleersnyder, Seba, Pravir Chandra, Kuai Hinojosa, ve Bart De Win. 2009. "Software Assurance Maturity Model - Version 1.0". : 96. http://www.opensamm.org/downloads/SAMM-1.0.pdf.

Demir, Bünyamin. 2015. *Yazılım Güvenliği, Saldırı ve Savunma*. 2. Baskı. İstanbul: Dikeyeksen Yayın.

Eminağaoğlu, Mete, ve Yılmaz Gökşen. 2009. "full-text". *BİLGİ GÜVENLİĞİ NEDİR, NE DEĞİLDİR, TÜRKİYE' DE BİLGİ GÜVENLİĞİ SORUNLARI VE ÇÖZÜM ÖNERİLER* 11(4): 01–15.

Emiral, Fatih. 2009. "Yazılım Güvenliği, Güvenliği Yazılımla Bütünleştirmek". : 8–10.

Enstitüsü, TÜBİTAK BİLGEM Siber Güvenlik. 2014. *Güvenli˙yazilim geliştirme temel kurallari dokümani*.

Franks,j, Hallem-Baker,P. , Hostetler, J. 1999. "RFC 2617, HTTP Authentication: Basic and Digest Access Authentication". https://www.ietf.org/rfc/rfc2617.txt (02 Nisan 2017).

Gary McGraw. 2006. *Software Security: Building Security in*. Addison-Wesley. https://books.google.com.tr/books?hl=en&lr=&id=HCQdypbpZXgC&oi=fnd&pg=PA1&dq=software+security&ots=npS2Ro_KHT&sig=ehGTWOVcgD3L3sV_NRbDax5fS9g&redir_esc=y#v=onepage&q=software security&f=false (02 Ağustos 2016).

"Get Started | CMMI Institute". 2017. http://cmmiinstitute.com/get-started (03 Nisan 2017).

GÜRLER, Korhan. 2007. "Güvenli Web Uygulamalarının Geliştirilmesi". : 6.

Hamilton, Booz Allen. 2014. "Cyber Power Index". : 1–36. papers3://publication/uuid/567637EE-E478-4634-908E-72E08A1EF0B8.

"Hash Fonksiyonu". https://tr.wikipedia.org/wiki/Hash_fonksiyonu.

Hewlett Packard Enterprise. 2016. Febrero 2016 *HPE Security Research Cyber Risk Report 2016*. http://techbeacon.com/sites/default/files/gated_asset/hpe-cyber-risk-report-2016.pdf.

Howard, Michael, ve Steve Lipner. 2006. "Development Lifecycle". (May).

"Information Systems Security Association". 2017. http://www.issa.org/ (03 Nisan 2017).

"Internet Live Stats - Internet Usage & Social Media Statistics".

http://www.internetlivestats.com/ (02 Nisan 2017).

“Internet Security Threat Report 2016 | Symantec”. https://www.symantec.com/security-center/threat-report?inid=globalnav_scflyout_istr (02 Nisan 2017).

ISO. 2012. “Iso 27032 Information technology — Security techniques — Guidelines for cybersecurity”. 25021: 11.

———. 2013a. “TS ISO/IEC 27001 Bilgi Teknolojisi – Güvenlik Teknikleri – Bilgi Güvenliği Yönetim Sistemleri – Gereksinimler”. 2012(112).

———. 2013b. “TS ISO/IEC 27002 Bilgi Teknolojisi - Güvenlik Teknikleri - Bilgi Güvenliği Kontrolleri İçin Uygulama Prensipleri”. (112).

“ISO 15504 (SPICE)”. 2014. https://en.wikipedia.org/wiki/ISO/IEC_15504.

ITU. 2008. “ITU-T-Rec.X.500”.

———. 2015. *CYBERSECURITY*.

Kabay,M.E. Whyne,Eric, Bosworth Seymour. 2009. *Computer Security Handbook*. fifth. https://www.safaribooksonline.com/library/view/computer-security-handbook/9780471716525/12_chap03.html (02 Nisan 2017).

Kara, Mehmet. 2004. “Türkiye’de Yazılım/Donanım Güvenliği Değerlendirme Çalışmaları”. *Tübitak-UEKAE*: 149–54.

Karasar, Niyazi. 2012. *Bilimsel araştırma yöntemi: Kavramlar, İlkeler, Teknikler.* Nobel Dağıtım Yayın.

“Kaspersky Cyberthreat real-time map”. https://cybermap.kaspersky.com/# (02 Nisan 2017).

“Kriptografi - Vikipedi”. https://tr.wikipedia.org/wiki/Kriptografi (02 Nisan 2017).

“List of cyber attacks”. https://en.wikipedia.org/wiki/List_of_cyber-attacks.

Mcgraw, Gary. 2009. Addison-Wesley Software Series *Software Security Touchpoint : Architectural Risk Analysis Cigital*.

“Number of Internet Users (2016) - Internet Live Stats”. http://www.internetlivestats.com/internet-users/ (02 Nisan 2017).

Owasp. 2013. “OWASP Top 10 - 2013”. *OWASP Top 10*: 22. http://owasptop10.googlecode.com/files/OWASP Top 10 - 2013.pdf.

OWASP. 2017. *Error Handling Cheat Sheet - OWASP*. https://www.owasp.org/index.php/Error_Handling (02 Nisan 2017).

Özbilgin, Gökhan Özlü, Mustafa. 2010. “Yazılım Geliştirme Süreçleri ve ISO 27001 Bilgi Güvenliği Yönetim Sistemi”. *Ulusal Bilgi Güvenliği Kapısı*: 1–9.

“Parkerian Hexad”. https://en.wikipedia.org/wiki/Parkerian_Hexad.

Pender-Bey, Georgie. 2012. “The Parkerian Hexad: The CIA Expanded”. : 4–20.

Richardson, Robert. 2013. “Executive viewpoint: Mixed messages on software security”. http://searchsecurity.techtarget.com/feature/Executive-viewpoint-Mixed-messages-on-software-security (02 Nisan 2017).

“Secure SDLC”. : 4.

Seker, Sadi Evren. 2015. “Yazılım Geliştirme Modelleri ve Sistem / Yazılım Yaşam Döngüsü”. *YBS Ansiklopedi* 2(3).

“Siber Bülten – Dikkat: 22 Türk bankasının müşterileri siber saldırıya hedef oldu”. https://siberbulten.com/strateji-guvenlik/dikkat-22-turk-bankasi-siber-saldiriya-hedef-oldu/ (02 Nisan 2017).

“Siber saldırıların tarihçesi”. http://www.nato.int/docu/review/2013/Cyber/timeline/TR/index.htm (02 Nisan 2017).

“Siber Savaş”. https://tr.wikipedia.org/wiki/Siber_savaş.

Singh, Aniruddha, Abhishek Vaish, ve Pankaj Kumar Keserwani. 2014. “Information Security: Components and Techniques”. *International Journal of Advanced Research in Computer Science and Software Engineering* 4(1): 2277–128.

“Software Development Spiral.svg - Wikipedia”. https://en.wikipedia.org/wiki/File:Software_Development_Spiral.svg (02 Nisan 2017).

“SSE-CMM (Systems Security Engineering Capability Maturity Model)”. https://www.symantec.com/security_response/glossary/define.jsp?letter=s&word=sse-cmm-systems-security-engineering-capability-maturity-model (03 Nisan 2017).

Stallings, William. 2011. 139 Network *Cryptography and Network Security Principles and Practice, 5th Edition*.

“SWOT analizi - Vikipedi”. 2016. https://tr.wikipedia.org/wiki/SWOT_analizi (03 Nisan 2017).

Temur, Büşra. 2013. “Yazılım Geliştirme Süreç Modelleri Nelerdir ?” https://salyangoz.com.tr/blog/yazilim-gelistirme-surec-modelleri-nelerdir/ (02 Nisan 2017).

“The three biggest cyber-attacks of 2016”. https://www.rheagroup.com/news/three-biggest-cyber-attacks-2016 (02 Nisan 2017).

Tiirik, Karl. 2004. “Comparison of SDL and Touchpoints”. : 1–6.

“Top 10 most notorious cyber attacks in history”. https://www.arnnet.com.au/slideshow/341113/top-10-most-notorious-cyber-attacks-history/?image=1 (02 Nisan 2017).

TSE. 2007. “TS ISO/IEC 12207”. 15189(112): ararlar yerel şartlar dikkate alınarak

ihtiyatla v.

“TSE- Ortak Kriter Nedir”. https://www.tse.org.tr/tr/icerikdetay/950/3296/ortak-kriter-nedir.aspx (03 Nisan 2017).

Ulastirma Bakanligi. “2016-2019 Ulusal Siber Guvenlik Stratejisi”.

Ulaştırma Denizcilik ve Haberleşme Bakanlığı. 2016. *2016-2019 Ulusal Siber Guvenlik Stratejisi*.

Ünver, Mustafa. 2015. “Türkiye ’ de Siber Güvenlik”. (August 2015).

“Wikileaks”. https://tr.wikipedia.org/wiki/WikiLeaks.

“Yazılım geliştirme süreci - Vikipedi”. 2016. https://tr.wikipedia.org/wiki/Yazılım_geliştirme_süreci (02 Nisan 2017).

YILMAZ, Yrd.Doç.Dr. Güray. 2007. “Yazılım Geliştirme Yaşam Döngüsü”. : 33.

EK-1 : Önerilen Güvenli Yazılım Geliştirme Politikası

EK-2 : Önerilen Güvenli Yazılım Geliştirme Kontrol Listesi

ÖZGEÇMİŞ

1985 yılında Eskişehir'de doğdu. Lise öğrenimini Eskişehir Kılıçoğlu Anadolu Lisesi'nde tamamladı. 2008 yılında Eskişehir Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümü'nden mezun oldu. 2011 yılında TOBB Ekonomi ve Teknoloji Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliğinde yüksek lisans eğitimini tamamladı. Halen ESOGÜ Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Bölümünde doktora eğitimine devam etmektedir. 2009 -2013 yılları arasında ESOGÜ Fen Bilimleri Enstitüsünde Araştırma Görevlisi olarak görev yaptı. 2013-2014 yılları arasında T.C. Başbakanlık Türkiye Atom Enerjisi Kurumu Bilgi İşlem Şubesinde mühendis olarak çalıştı. 2014 yılında Çevre ve Şehircilik Bakanlığı'nda Çevre ve Şehircilik Uzman Yardımcısı olarak göreve başladı. Kendisi evlidir.

ETİK KURALLARA UYGUNLUK BEYANI

Uzmanlık tezi olarak sunduğum bu çalışmayı, bilimsel ahlak ve geleneklere aykırı düşecek bir yol ve yardıma başvurmaksızın yazdığımı, yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu, bunlardan her seferinde değinme yaparak yararlandığımı ve Çevre ve Şehircilik Uzmanlığı Yönetmeliğine uygun olarak hazırladığımı belirtir, bunu onurumla doğrularım. Çevre ve Şehircilik Bakanlığı tarafından belli bir zamana bağlı olmaksızın, tezimle ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara katlanacağımı bildiririm.

06.04.2017

Gülizar Duygu KURT KAYA