GÜVENLİ VE KULLANIŞLI RESİM-ŞİFRE YÖNTEMLERİNİN TASARLANMASI VE GERÇEKLEŞTİRİLMESİ

MUSTAFA YÜCEEL

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EYLÜL 2010

ANKARA

Fen Bilimleri Enstitü onayı

_____

Prof. Dr. Ünver KAYNAK

Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

_____

Doç. Dr. Erdoğan DOĞDU

Anabilim Dalı Başkanı

Mustafa YÜCEEL tarafından hazırlanan GÜVENLİ VE KULLANIŞLI RESİM-ŞİFRE YÖNTEMLERİNİN TASARLANMASI VE GERÇEKLEŞTİRİLMESİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

_____	_____
Doç. Dr. Bülent TAVLI Tez Danışmanı	Doç. Dr. Kemal BIÇAKCI İkinci Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Doç. Dr. Kemal BIÇAKCI _____

Üye : Prof. Dr. Ünver KAYNAK _____

Üye : Doç. Dr. Bülent TAVLI _____

Üye : Yrd. Doç. Dr. Tansel ÖZYER _____

Üye : Yrd. Doç. Dr. Hakan GÜLTEKİN _____

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Mustafa YÜCEEL

Üniversitesi : **TOBB Ekonomi ve Teknoloji Üniversitesi**
Enstitüsü : **Fen Bilimleri**
Anabilim Dalı : **Bilgisayar Mühendisliği**
Tez Danışmanı : **Doç. Dr. Bülent TAVLI**
İkinci Tez Danışmanı : **Doç. Dr. Kemal BIÇAKCI**
Tez Türü ve Tarihi : **Yüksek Lisans – Eylül 2010**

Mustafa YÜCEEL

GÜVENLİ VE KULLANIŞLI RESİM-ŞİFRE YÖNTEMLERİNİN TASARLANMASI VE GERÇEKLEŞTİRİLMESİ

ÖZET

Kullanıcı adı ve şifre kullanarak kimlik kanıtlama güvenli erişim gerektiren uygulamalar için çok yaygın olarak kullanılan bir yöntemdir. Kullanıcıların çok kolay tahmin edilebilen şifreler seçmesi bir sistemin veri transferinde ve altyapısında aldığı diğer güvenlik önlemlerini etkisiz ve anlamsız kılmaktadır. Kullanıcılar için hem güvenli hem kullanışlı kimlik kanıtlama sistemleri gerekmektedir.
Bu tezde metin tabanlı şifrelere alternatif olarak sunulan grafik şifre yöntemlerini inceliyoruz. İnsanların grafiksel bilgileri sözel veya metinsel bilgilerden daha kolay hatırlamaları gerçeğinden yola çıkarak, grafik şifre sistemlerinin başarılı olabileceğini düşünüyoruz. Fakat geliştirilen sistemin kullanışlı olması kadar güvenli olması da değişmez bir gereksinimdir. Güvenlik ve kullanışlılığı bir arada temin etmek zordur ve kullanıcıyı yakından ilgilendiren birçok sistem gibi grafik şifre yöntemlerinin de temel problemlerinden bir tanesidir. Grafik şifre çalışmaları farklı bakış açıları ve yeni sistemlerle desteklenerek devam etmektedir ve göreceli olarak başarılı yöntemler geliştirilmiştir. Fakat bu yöntemlerin gerçek hayatta kullanıma geçmesi insanların alışkanlıklarından vazgeçmeleri ile mümkün olabilmektedir. Bu tezde başarılı grafik şifre yöntemlerinin pratikte kullanımına yönelik bir çalışma yaptık ve açık kaynak kodlu bir ağ tarayıcı eklentisi geliştirdik. Bu eklenti ile uygulamaların metin tabanlı şifre yöntemlerini değiştirmelerine gerek duymadan grafik şifrelerin kullanılabilmesini sağladık. Sunulmuş olan grafik şifre tasarılarından başarılı bir yöntem olan PassPoints yöntemini seçtik ve geliştirme ortamına uygunluğunu inceleyerek gerekli uyarlamaları yaptık. GPI, GPIS isimlerinde yeni tasarılar sunduk, güvenlik ve kullanışlılık deneylerini gerekli karşılaştırmaları yaparak gerçekleştirdik. Tüm bu çalışmaların amacı, kullanışlı, hatırlaması kolay şifreler oluşturulabilen güvenli grafik şifre tasarımları sunmak ve bu tasarımları pratik kullanıma elverişli bir uygulama olarak gerçekleştirmektir.

Anahtar Kelimeler: grafik şifre, kimlik kanıtlama, güvenlik, kullanışlı güvenlik

University : **TOBB Economics and Technology University**
Institute : **Institute of Natural and Applied Sciences**
Science Programme : **Computer Engineering**
Supervisor : **Associate Professor Dr. Bülent TAVLI**
Co-Supervisor : **Associate Professor Dr. Kemal BIÇAKCI**
Degree Awarded and Date : **M. Sc. – September 2010**

Mustafa YÜCEEL

DESIGNING AND IMPLEMENTING SECURE AND USABLE GRAPHICAL PASSWORD METHODS

ABSTRACT

Authentication with a username and password is very common in applications which require secure access. Security precautions on data transmission and on infrastructure are meaningless if users select weak passwords. This means that there is a great demand for both usable and secure authentication methods.

In this thesis we investigate graphical password schemes which were proposed as alternatives to text based authentication systems. The fact that people recall images better than verbal or textual knowledge suggests that graphical password systems could be successful. On the other hand, it is very difficult to provide security and usability simultaneously for systems that involve human interaction. The literature on graphical password schemes presents new and improved designs. However, changing habits of people is a challenging process. In this thesis we put forward a new idea to bring graphical passwords into practical usage. We developed an open-source browser extension which facilitates the use of graphical passwords on all web applications without a need to change the traditional text based password scheme implemented on servers. We have chosen one of the successful schemes i.e., PassPoints and investigated appropriateness of its usage in our development environment by conducting experiments. Then we proposed new schemes GPI and GPIS and conducted usability and security experiments. One of the main purposes of our studies is to find a usable and secure graphical password scheme so that users can use in practice and can generate passwords which are easy to remember but hard to guess.

Keywords: graphical password, security, usable security, authentication

TEŞEKKÜR

Çalışmalarımda çok büyük katkıları olan, bana karşı manevi desteklerini esirgemeden daima güven veren hocalarım ve kıymetli danışmanlarım Doç. Dr. Kemal BIÇAKCI, Doç. Dr. Bülent TAVLI' ya, beni kırmayarak tez savunmamı değerlendirmeyi kabul eden Yrd. Doç. Dr. Tansel ÖZYER' e en içten teşekkürlerimi sunarım. Akademik başarı, çalışkanlık ve yüksek insani değerlere sahip olan bu hocalarım benim için daima örnek olacaklardır.

Bu tez 107E227 proje numarasına sahip, TÜBİTAK (Türkiye Bilimsel ve Teknolojik Araştırma Kurumu) tarafından desteklenen ve Doç. Dr. Kemal BIÇAKCI koordinatörlüğünde yürütülen "Güvenli ve Kullanışlı Resim-Şifre Yöntemlerinin Tasarlanması ve Gerçekleştirilmesi" isimli proje kapsamında meydana gelmiştir. Bu projede çalışmış olan Dr. Nart Bedin ATALAY, Burak ERDENİZ ve Hakan GÜRBAŞLAR' a teşekkür ederim.

Hayatım boyunca benim için birçok fedakârlıklar yapan ve daima yanımda olduklarını hissettiren babam Mehmet Ali YÜCEEL, annem Aysel YÜCEEL ve ablam Elif Nur YEŞİLYURT' a en samimi teşekkürlerimi sunuyorum ve ayrıca eniştem İsa YEŞİLYURT' a manevi destekleri için teşekkür ediyorum.

İÇİNDEKİLER

ÖZET iii

ABSTRACT iv

TEŞEKKÜR v

İÇİNDEKİLER vi

ÇİZELGELERİN LİSTESİ viii

ŞEKİLLERİN LİSTESİ ix

KISALTMALAR xi

Kısaltmalar Açıklama xi

SEMBOL LİSTESİ xii

BÖLÜM 1 – GİRİŞ 1

1. 1 Teze Genel Bakış 2

1. 2 Bu Araştırmanın Temel Amaçları 3

BÖLÜM 2 – GENEL BİLGİLER 4

2. 1 Kimlik Kanıtlama 4

2. 2 Metin Tabanlı Şifreler 5

2. 3 Grafik Şifreler 7

2. 3. 1 Hatırlamaya Dayalı Grafik Şifre Yöntemleri 7

2. 3. 2 Tanımaya Dayalı Grafik Şifre Yöntemleri 9

2. 3. 3 İpucuyla Hatırlamaya Dayalı Grafik Şifre Tasarıları 10

2. 4 Şifre Alanı 11

2. 5 Sıcak-nokta Problemi 12

BÖLÜM 3 – EKLENTİ İÇİN UYGUN YÖNTEMİN SEÇİLMESİ 15

3. 1 PassPoints Tasarısının incelenmesi 15

3. 1. 1 Güvenlik ve Taşınabilme Problemleri 19

3. 2 PassPoints tasarısının eklenti için uyarlanması 21

3. 2. 1 Deney 21
3. 2. 2 Deney Sonuçları 23
3. 3 GPI-GPIS Tasarıları 28
3. 3. 1 Deney 28
3. 3. 2 Deney Sonuçları 32
BÖLÜM 4 – GPEX (GRAPHICAL PASSWORD AS BROWSER EXTENSION) 38
4. 1 GPEX in geliştirim esasları 39
4. 1. 1 Eklentiler Hakkında 39
4. 1. 2 Eklentilerde Kullanılan Teknolojiler 39
4. 1. 3 Temel Klasör - Dosya Yapısı ve Geliştirme Ortamı 45
4. 1. 4 Firefoxu Genişletmek 47
4. 1. 5 Krom Bildirisi 49
4. 1. 6 Eklentinin Çalışması 50
4. 1. 7 Şifre Kuralları 53
4. 1. 8 Seçenekler ve ara yüz ayarlamaları 55
4. 2 Bir Şifre Yöneticisi olarak GPEX 56
4. 2. 1 PwdHash 59
4. 2. 2 PwdHash ve GPEX Kullanışlılık karşılaştırması 60
4. 3 Grafik Şifre Sistemi olarak GPEX 65
4. 3. 1 Uzaktan Erişim 67
4. 4 İnternet güvenliği açısından GPEX 68
BÖLÜM 5 SONUÇ VE GELECEKTEKİ ÇALIŞMALAR 71
KAYNAKLAR 76
ÖZGEÇMİŞ 79

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 4. 1 İnternet Güvenliği Anketi 61

Çizelge 4. 2 GPEX ve PwdHash için görev tamamlama sonuçları 63

Çizelge 4. 3 GPEX – PwdHash Deney Sonu Anket Soruları 63

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. PassPoints Grafik Şifre Yöntemi 8

Şekil 2.2. PassFaces Grafik Şifre Yöntemi 9

Şekil 2.3 Cued Click Points Grafik Şifre Yöntemi 10

Şekil 2.4. Etkili Şifre Alanının İncelenmesi 12

Şekil 3.1 Kenar Problemi : Şekilde, ortada görünen mavi nokta, kullanıcının tıkladığı orijinal noktadır. Bu tasarıda tolerans aralığı x ve y düzleminde +10, -10 piksel olarak düşünülmüştür. Hatalı Ret: Sağdaki yeşil nokta kullanıcının şifresini girmek için tıkladığı fakat 5 piksel sapmayla seçtiği noktadır. +10 tolerans aralığındaki bir tasarının bu girişi doğru olarak kabul etmesi gerekirken, bu tasarıda tıklama başka bir kare içine kaydığı için giriş başarısız olarak kabul edilmektedir. Hatalı Kabul: Soldaki pembe nokta kullanıcının şifresini girmek için tıkladığı fakat 15 piksel sapmayla seçtiği noktadır. +10 tolerans aralığındaki bir tasarının bu girişi kabul etmemesi gerekirken, bu tasarıda tıklama aynı kare içine yapıldığı için giriş doğru kabul edilmektedir. 17

Şekil 3.2 Izgaralı ve ızgarasız durumlarda şifre üretmek için harcanan süre. 23

Şekil 3.3 Izgaralı ve ızgarasız durumlarda şifre onaylamak için harcanan süre. 23

Şekil 3.4. Izgaralı ve ızgarasız durumlarda doğru şifreyi hatırlamak için yapılan deneme sayısı 24

Şekil 3.6 Izgaralı ve ızgarasız durumlar için resim üzerindeki bölge sayıları ve tıklama sayıları. 26

Şekil 3.7. GPI ara yüzü (sol) ve GPIS ara yüzü (sağ). GPI ara yüzünde kullanıcı ikonları kendisi seçerken GPIS ara yüzünde ise sistem seçer ve gösterir. (şekiller sayfaya sığması için küçültülmüştür.) 31

Şekil 3.8. GPI, GPIS ve PassPoints ara yüzleri için şifre onaylamada geçen süre 33

Şekil 3.9. GPI, GPIS ve PassPoints ara yüzlerinde şifre hatırlamak için yapılan deneme sayıları 33

Şekil 3.10. GPI, GPIS and PassPoints ara yüzlerinde doğru hatırlanan şifreler girilirken geçen süre. 34

Şekil 3.11. PassPoints tasarısında her bölge için tıklama sayıları. 35

Şekil 3.12. GPI ara yüzünde her bölge için tıklama sayısı 36

Şekil 4.1. Basit Bir XUL Dosyası 42

Şekil 4.2. XUL ara yüzü 42

Şekil 4.3. Basit Bir XPI Dosyasının Yapısı 45

Şekil 4.4. Install.rdf Dosyasının Görünümü 46

Şekil 4.5. browser.xul dosyası üzerine bindirilen xul dosyası 48

Şekil 4.6 chrome. manifest dosyasının görünümü 49

Şekil 4.7. Eklenti penceresini oluşturan pwd.xul dosyasının görünümü 51

Şekil 4.8. Eklenti penceresinin ara yüz görünümü 51

Şekil 4.9. Şifre kurallarının belirlendiği rules. xml dosyası 53

Şekil 4.10. Tercihler penceresi ara yüz görünümü 56

Şekil 4.11. GPEX eklentisinde kullanılan grafik şifre tasarıları 59

Şekil 4.12. Anket Sonuçları 64

Şekil 4.13. Uzaktan Erişim Ara Yüz Görüntüsü 67

KISALTMALAR

Kısaltmalar	Açıklamalar
CCP	Cued Click-Points
GPEX	Graphical Password as Browser Extension
GPI	Graphical Password with Icons
GPIS	Graphical Password with Icons suggested by the System
ODTÜ	Ortadoğu Teknik Üniversitesi
PCCP	Persuasive Cued Click-Points
TOBB ETÜ	Türkiye Odalar ve Borsalar Birliği Ekonomi ve Teknoloji Üniversitesi

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
r	Tolerans aralığı
x	X düzleminde tıklanan ilk noktanın merkeze uzaklığı
X	X düzleminde tıklanan ikinci ve sonraki noktaların merkeze uzaklığı
d	Göreli konum değeri
i	x in bölge işaretçisi
I	X in bölge işaretçisi
h()	Özet fonksiyonu
P	Permutasyon
$\mathbf{E_a}$	a defa seçilen alan sayısı
a	Tıklanma sayısı

BÖLÜM 1 – GİRİŞ

İnternet uygulamalarının kullanımı bütün dünyada yaygın durumdadır ve birçok kullanıcı tarafından kullanımları sebebiyle kullanıcılara kimlik denetimi yapılmalıdır. Kimlik denetimi, güvenlik ve kullanışlılık açısından önem arz etmektedir ve bu sebeple kimlik denetim sistemi uygulamanın kullanımını zorlaştırmamak kaydıyla yeterli güvenliği sağlayacak şekilde tasarlanmış olmalıdır.

Kimlik denetimi mekanizmaları kullanıcıların ne bildiklerine (kullanıcı adı – şifre kimlik denetim sistemleri gibi), ne olduklarına (parmak izi veya retina testi ile kimlik doğrulayan sistemler gibi) veya ne taşıdıklarına (kimlik kartına göre doğrulama yapılan sistemler gibi) göre farklı şekillerde tasarlanabilir. Yaygın olan kimlik denetim mekanizması kullanıcıların önceden belirlenen kullanıcı adı ve metin tabanlı şifreleri girmesi esasına dayanır. Belirlenecek şifreyi kullanıcılar seçiyorlarsa seçilen şifrenin daha kolay hatırlanabilmesi için kolay tahmin edilebilir şifre seçilmesi bir güvenlik problemi olduğu gibi seçilen şifrenin güvenli fakat karmaşık dolayısıyla zor hatırlanabilir olarak seçilmesi de bir kullanışlılık problemidir. Güvenlik ve kullanışlılığın birlikte sağlanması zor bir problemdir ve bu konu ile ilgili çalışmalar devam etmektedir [1–3].

Grafik şifre (Resim şifre), bilgisayarın veya şifre girilecek olan aygıtın grafiksel giriş/çıkış aygıtlarının kullanılmasıyla oluşturulan, kullanıcıya özgü, gizli bilgidir. İnsan beyninin imajları metinlerden daha kolay tanıması ve hatırlaması [4–7] esasından faydalanarak grafik şifrelerin kullanımı önerilmiştir. Bu sayede hatırlama kolaylığı, başka bir deyişle kullanışlılık açısından metin tabanlı şifrelere karşı bir üstünlük elde edilmesi mümkün olabilir. Her ne kadar insan beyni grafikleri kolay hatırlama eğiliminde olsa da, grafik şifre yöntemlerinde de gerekli güvenliği, yeterli kullanışlılık ile birlikte temin etmek bir problem olarak önümüze çıkabiliyor. İnternet ortamına taşınan birçok uygulamayla birlikte şifre ve kullanıcı adı gerektiren uygulamalar artmıştır. Bu durum, bir kullanıcının aynı şifreyi birçok internet sitesinde kullanma durumunu doğurmuştur. Saldırganların zayıf güvenlikli bir internet sitesinden ele geçirdikleri şifreyi, bu kullanıcının kullandığı diğer sitelere

giriş için kullanabilmeleri bir güvenlik zayıflığıdır ve kullanıcıya tahmininden daha büyük zararlar getirebilir.

Kullanışlılığı ve güvenliği arttırmak üzere hatırlama, tanıma ve ipucuyla hatırlama gibi türlerde yeni grafik şifre önerileri ortaya atılmaktadır. Bu tezde, geliştirilmiş olan grafik şifre yöntemleri üzerinde durulacaktır, hatırlama tabanlı ve tanıma tabanlı iki farklı türdeki grafik şifrelerin kısa dönemli deneylerinden bahsedilecektir. Önerdiğimiz grafik şifre yöntemlerinin tarayıcı eklentisi olarak gerçekleştirimine de yer verilmiştir ve bu gerçekleştirimde eklenti bir şifre yönetim aracı olarak da görev yapmak suretiyle bir şifrenin birçok site için güvenli bir şekilde kullanılabilmesine imkân tanımıştır.

Grafik şifreler üzerine hazırlanmış olan bir doktora tezinin[8] temel amaçları arasında güvenlik ve kullanışlılık açısından farklı grafik şifre yöntemlerini deneysel olarak inceleyerek arasından en umut vaat edeni seçmek de vardır ve bu sebeple incelenen diğer tasarılar arasından kullanışlılık ve güvenlik açısından üstün olan yöntemi belirlemişlerdir. Biz de gerçekleştirdiğimiz eklentide kullanmak üzere amacımıza en uygun, yani taşınabilme, yavaş yüklenme zamanı, karmaşıklık, düşük güvenlik gibi problemleri olmayan bir tasarı seçmeye çalıştık. Deneysel olarak değil fakat gözlemsel olarak uygun bir tasarı seçtik. Seçtiğimiz bu tasarının yukarıda bahsedilen tezde de kullanışlı bir tasarı olarak seçilmiş olması doğru bir tercih yaptığımızı göstermiştir. Seçtiğimiz bu tasarı, başka çalışmalar yapılarak geliştirilmiştir. Bu geliştirilmiş tasarıların eklentimizde kullanılabilme durumunu ilerideki bölümlerde inceleyeceğiz.

1. 1 Teze Genel Bakış

Bu tez içerisinde grafik şifre ve resim şifre terimleri aynı anlamda kullanılmıştır ve farklı yerlerde iki farklı kullanıma rastlamak mümkündür. Bu tezin gelecek kısımları şu şekilde organize edilmiştir; Bölüm 2 de kimlik kanıtlamanın tanımı ile birlikte metin tabanlı ve grafik tabanlı olmak üzere iki farklı kimlik denetimi yönteminden

bahsedilerek grafik tabanlı şifrelere örnekler verilecektir. Şifre alanları ve grafik şifrelerin problemlerinden de yine bu bölümde bahsedilecektir. Bölüm 3 de, gerçekleştireceğimiz bir tarayıcı eklenti uygulaması için en uygun grafik şifre tasarılarının seçimini konu alacağız ve önerdiğimiz yeni tasarıların incelemelerini yaparak deneylerini anlatacağız. Bölüm 4 de geliştirdiğimiz uygulamanın gerçekleştirim esaslarından, kullanılan teknolojilerden bahsettikten sonra uygulamayı farklı bakış açılarıyla inceleyeceğiz. Bölüm 5 i de sonuç ve gelecekte yapılabilecek çalışmalara ayıracağız.

1. 2 Bu Araştırmanın Temel Amaçları

Bu tez, yapılan grafik şifre çalışmalarına katkı sağlamak üzere temel olarak şu amaçları taşımaktadır;

- Kimlik denetimi ve yetkilendirme yöntemi olarak kullanılan metin tabanlı şifre yönteminden yeni geliştirilen grafik şifre yöntemlerine geçişi kolaylaştırmak ve bu sürece internet tarayıcı eklentisi geliştirerek katkı sağlamak.

- Grafik şifre yöntemlerinin bilinen problemlerini gidermek yolunda kullanışlılığı ve güvenliği düşürmeden iyileştirme sağlamak.

BÖLÜM 2 – GENEL BİLGİLER

2. 1 Kimlik Kanıtlama

Güvenlik sistemleri yetkili kişilerin girişine izin verecek ve yetkili olmayan kişilerin girişlerini engelleyecek şekilde tasarlanmışlardır. Bu işlem 3 farklı basamak içermektedir: kimlik belirleme, kimlik kanıtlama ve yetkilendirme.

Kimlik belirleme: Bu aşamada kullanıcıya kendisini tanımlaması için özgün bir bilgi sorulur ki bu bilgi genellikle e-posta adresi, kimlik numarası, hesap numarası veya kullanıcı adıdır.

Kimlik kanıtlama: Bu aşamada kullanıcı tanımladığı kimliğin kendisine ait olduğunu ispatlamak için bir kanıt sunmalıdır. Bu kanıt kullanıcının bildiği bir şifreyi girmesi, daha önceden belirlemiş olduğu bir varlığı tanıması veya daha önceden gösterdiği bir davranışı göstermesi gibi birçok değişik yolla sunulabilir.

Yetkilendirme: Kimliğini belirleyip, bunu kanıtlayan kişiye sistemde sahip olduğu yetkileri verme aşamasıdır.

Kişiler ile sistem arasında gizli bir bilgi tutulur ve sistem bu bilgiyi doğru olarak sağlayabilen kişinin doğru kişi olduğunu kabul ederek gerekli yetkilendirmeyi yapar. Burada bahsedilen gizli bilgi sisteme göre değişiklik gösterebilir. Kullanıcının ezberlediği ve biliyor olması gereken bir bilgi, kullanıcının ezberlemiş olması gerekmeyen ancak gördüğünde tanıması gereken bir varlığın bilgisi veya kullanıcının kişiliğine ait olan bir bilgi sistem ve kullanıcı arasında gizli bir bilgi olarak paylaşılabilir.

2. 2 Metin Tabanlı Şifreler

Metin tabanlı şifreler birçok kimlik kanıtlama yöntemi arasından en yaygın olarak kullanılanıdır. PricewaterhouseCoopers [9]tarafından yapılan bir araştırmanın sonuçlarına göre İngiltere'deki firmalar arasında kullanıcı adı ve şifre kullanılarak yetkilendirme yapılması %93 oranında çok yaygın bir uygulamadır [10] ve ortalama bir kullanıcı üç farklı kullanıcı adı-şifre çifti ezberlemek zorundadır. Firmaların diğer yetkilendirme yöntemlerini kullanmamalarının sebepleri arasında diğer sistemlerin pahalılığı ile birlikte kullanışlılık problemleri yer almaktadır. Kullanıcıların birçok kullanıcı adı ve şifre kullanmak zorunda kalmaları her bilgisayar kullanıcısının ve özellikle internet kullanıcılarının karşılaştığı bir durumdur ve bir araştırmaya göre[11] internet kullanıcılarından sadece %19 u her site için farklı şifre kullanmaktadır[12]. Şifrelerin karmaşık olarak seçilmesi tahmin edilebilirliğini azaltarak güvenliği arttırır ancak kişilerin şifrelerini hatırlamasını zorlaştırarak kullanışlılığı azaltır. Şifrelerin hatırlanması için oluşturulan şifrenin kullanıcıya bir anlam ifade etmesi önemlidir. Anlamsız karakter dizilerini hatırlamak ve uzun süre unutmamak, anlamlı karakter dizilerine göre daha zordur[13]. Birden çok şifre kullanma zorunluluğu da göz önüne alındığında kullanıcılar kullanışlılığı arttırmak adına güvenlikten taviz vererek karmaşık şifreler (örn. %rt6WgY!1) yerine kolay tahmin edilebilir (örn. password, baseball) şifreler seçmektedirler[14]. Bu durumlarda kullanıcılara karmaşık şifreler tavsiye etmek veya kullandıkları şifrenin güvensiz olduğu uyarısını vermek kullanıcıların karmaşık şifreleri ezberlemeleri için ikna edici olmamaktadır veya karmaşık şifreye sahip kullanıcılar ezberlemek yerine kolay ulaşılabilir bir yere not etmeyi tercih etmektedirler. Kullanıcılara güvenli ve kullanışlı şifre oluşturmaları için tavsiye edilen bir yöntem ise elit sözcükler ("leet speak") yöntemidir[15]. Bu yöntemde bir kelime veya cümle hatırlatıcı olarak kullanılmaktadır ve oluşturulacak şifre bu cümleye göre oluşturulmaktadır. Bu yöntemde cümledeki bazı harfleri çıkararak, kelimelerin sadece baş harflerini kullanarak veya harfleri benzer rakamlarla değiştirerek şifreler oluşturulabilir. Hatırlamada faydalı olabilecek bu yöntem şifrenin tahmin edilebilirliğini engellemede, popüler kelime ve deyimlerin kullanılmasıyla ve harf değişimlerinin (örn. A yerine 4, E yerine 3) tahmin edilebilir olması sebebiyle yetersiz kalabilir.

Bunun yanında bir şifrenin birden fazla sitede kullanılmasına da çözüm olamamaktadır çünkü kullanıcılar hangi sitenin şifresinin ne olduğunu karıştırabilecekleri için birden fazla siteye bir şifre kullanma eğiliminde olacaklardır.

Kullanıcılar yukarıda bahsedilen şifrenin karmaşıklığı, kimlik kanıtlama gerektiren birçok web uygulaması bulunması gibi sebeplerden dolayı şifrelerini yazma ihtiyacı duyarlar. Bu ihtiyacı güvenli bir şekilde karşılamak üzere şifre yöneticileri geliştirilmiştir. Şifre yöneticilerinin bir kısmı kullanıcı adı, şifre ve internet sitesi adresi gibi bilgileri saklar, kullanıcının ihtiyacı olduğunda kullanıcıya sunar. İnternet tarayıcılarına eklenti olarak kurulan ve kullanıcıdan aldığı şifreyi siteye özel güçlü bir şifre oluşturmak için kullanan şifre yöneticileri de bulunmaktadır [16, 17]. Tarayıcı eklentisi olarak gerçekleştirilen şifre yöneticileri metin şifrelere yönelik olarak çalışmaktadır. Bölüm 4'de de bahsedeceğimiz üzere bizim geliştirmiş olduğumuz grafik şifre eklentisi bir şifre yöneticisi görevi görmekle birlikte grafik şifre yöntemini kullanmaktadır ve bilgimize göre bunun daha önce benzer bir uygulaması yoktur.

Metin şifrelerin sahip olduğu bir diğer problem de şifre alanı ile ilgilidir. Şifre alanı, oluşturulması mümkün olan bütün şifreler kümesidir. 8 haneli bir şifre içerisinde kullanılabilecek 26 küçük harf, 26 büyük harf, 10 adet rakam ve 32 adet sembolden oluşan, yani 94 farklı ASCII karakteri için oluşturulabilecek şifre sayısı teorik olarak 94^8 dır. Hâlbuki kullanıcılar hatırlamak adına kendilerine anlamlı şifreler seçmek için tamamı harflerden oluşan anlamlı bir şifreyi harfler ve sembollerden oluşan bir şifreye tercih etmektedirler. Bu durum da teoride mümkün olan şifre alanının kullanılamaması demektir. Uygulamaların şifre içinde sembol, sayı, büyük harf ve küçük harf bulunma zorunluluğu koyması şifrenin güvenliği arttırmak için bir adım olsa da, bu yaklaşım da yine şifre alanını azaltmaktadır. Örneğin, yukarıda tanımlanan 94 karakter için oluşturulabilecek sembol içermesi zorunlu olan şifre sayısı, oluşturulabilecek tüm şifrelerin sayısı ile sembol içermeyen şifrelerin sayısının farkı olan $((94^8) - ((94-32)^8))$ dır. Buna rakam, küçük ve büyük harf ekleme zorunlulukları da eklendiğinde şifre alanında daha fazla bir azalma söz konusu olur. Zorunluluklar eklendiğinde meydana gelen şifre alanındaki bu azalma

çok kritik bir azalma sayılmayabilir ve bu sebeple karakter içerme zorunluluğunun getirdiği kullanışlılık problemi daha öncelikli bir öneme sahiptir.

2. 3 Grafik Şifreler

Metin tabanlı şifrelerin yanında alternatif kimlik kanıtlama yöntemleri de bulunmaktadır ve grafik şifreler bu yöntemlerden bazılarıdır. Grafik şifre fikri ve tasarımı ilk olarak Greg E. Blonder [18] tarafından ortaya atışmıştır. Blonder ın tasarımında daha önceden belirlenmiş bir resim üzerinde yine önceden belirlenmiş bölgeler bulunmaktadır ve kullanıcı sisteme erişebilmek için doğru bölgeleri doğru sırayla seçmelidir. Grafik şifreler üzerine yapılan çalışmalar daha önce insan beyni üzerinde yapılmış olan bilişsel araştırmalara dayanmaktadır. Bu bilişsel araştırmalar[19–22], insan beyninin metinsel ve sözel bilgilere nazaran görsel bilgileri daha kolay tanıdığını ve hatırladığını ortaya koymuştur. Bu gerçekten faydalanarak birbirinden farklı grafik şifre tasarıları [23–25] oluşturulmuş, güvenlik ve kullanışlıkları incelenmiştir ve bu alandaki çalışmalar devam etmektedir. Bu başlık altında grafik şifre kategorilerinin tanımlarıyla birlikte farklı kategorilerde geliştirilmiş olan grafik şifrelere örnekler verilerek grafik şifreler hakkında daha net bir fikir vermek amaçlanmıştır. Oluşturulan farklı grafik şifreler 3 kategoride incelenebilir:

2. 3. 1 Hatırlamaya Dayalı Grafik Şifre Yöntemleri

Hatırlamaya dayalı grafik şifre yöntemlerinde kullanıcının bir defa seçtiği veya oluşturduğu (ya da sistem tarafından belirlenen) bir yapıyı, nesneyi veya aktiviteyi yeniden oluşturması veya gerçekleştirmesi beklenir. Bu kategori içerisinde verilebilecek güzel bir örnek Passpoints [23] yöntemi Şekil 2.1.'de gösterilmiştir.

Şekil 2.1. PassPoints Grafik Şifre Yöntemi

Passpoints yöntemi daha önce bahsedildiği üzere ilk defa ortaya atılan Blonder [18]'ın grafik şifre yönteminden türetilmiştir. Passpoints'te Blonder'ın yönteminden farklı olarak önceden belirlenmiş alanlar bulunmamaktadır. Bu yöntemde kullanıcı resim üzerinde beş farklı noktayı fare ile tıklayarak seçer. Kullanıcının seçtiği bu noktalar o kişinin şifresi olarak belirlenmiş olur ve bundan sonraki girişlerde kullanıcı bu noktaları aynı sıra ile yeniden seçmelidir. Burada ilk defa seçilen nokta bir piksel olduğu için yeniden aynı tek pikseli seçmek çok zor olacaktır. Bu sebeple ilk başta tıklanılan pikselden belli bir uzaklıktaki piksellerin seçilmesini de kabul edilebilir görmek gereklidir. Bu yöntemde kullanılabilecek resim için bir sınır getirilmemekle birlikte bir resim üzerinde binlerce tıklanılabilecek alan bulunduğu için bu yöntemin şifre alanı oldukça geniştir. Bu yöntemde alfa nümerik şifrelere göre daha az denemede doğru şifrenin girilmesi gözlemlenmiştir ancak kullanıcıların şifrelerini öğrenmeleri daha zor olmuş ve şifre girerken geçen süre artmıştır[23]. Yine bu yöntemde karşılaşılan bir problem ise kullanılan resmin yapısından veya kullanıcıların yönelimlerinden kaynaklanan sebeplerle resim üzerindeki bazı noktaların diğer noktalara göre çok daha fazla seçilmesidir. Bu durum bir problem teşkil etmektedir çünkü bir saldırganın resim üzerinde yapacağı bir analizle belirlediği bazı noktalar üzerinde yoğunlaşıp dar bir şifre alanı üzerinde denemeler yapmasına imkân sağlamaktadır. Bu problem birçok grafik şifre yönteminde

bulunmaktadır ve bu problemin çözümüne yönelik yaptığımız çalışmalardan ilerideki bölümlerde bahsedeceğiz.

2. 3. 2 Tanımaya Dayalı Grafik Şifre Yöntemleri

Tanımaya dayalı şifre yöntemlerinde kullanıcıdan bir resim kümesi içerisinden seçilen (veya sistemin belirlediği) bazı resimleri tıklaması beklenmektedir. Bu yöntemde kullanıcı birçok resim arasından bazı resimleri tanıyarak tıklamalıdır.
Real User Corporation [2] tarafından geliştirilen Passfaces[2], tanımaya dayalı grafik şifre yöntemlerine örnek olarak verilebilir.

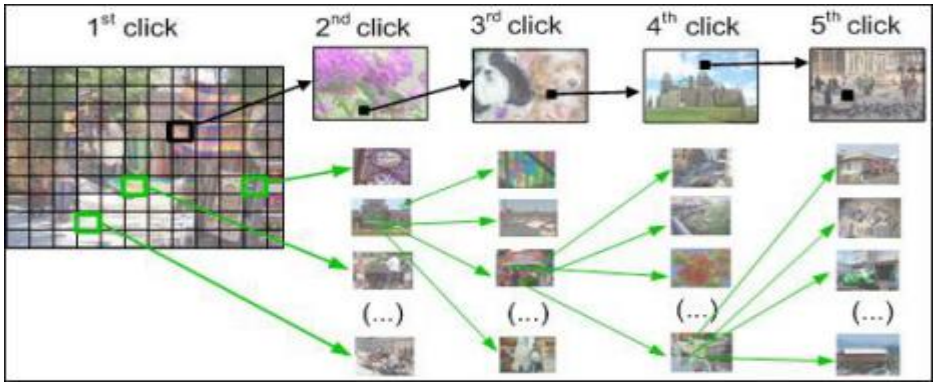
Passfaces de kullanıcıdan sunulan yüz resimleri arasından 4 tanesini seçmesi beklenmektedir ve seçilen bu resimler kullanıcının şifresi olarak belirlenmektedir. Sisteme giriş yapılacağı zaman Passfaces 8 adet çeldirici yüz resmi ve kullanıcının seçmiş olduğu yüzlerden 1 tanesini içerecek şekilde 9 adet yüz resmi gösterir[2](Şekil 2.2.). Kullanıcı sisteme girebilmek için bu resimlerden önceden seçmiş olduğu (ve dolayısıyla tanıyacağı) resmi seçmelidir. Bu işlem bu şekilde birkaç defa tekrar edilir ve eğer kullanıcı 4 resmi de doğru olarak seçmiş ise giriş işlemi gerçekleştirilir.

Şekil 2.2. PassFaces Grafik Şifre Yöntemi

Bu yöntemin uzun vadede hatırlanabilir ve hata oranı düşük olduğu yapılan çalışmalarda [24, 26, 27] görünse de güvenlik açısından eksikleri bulunmaktadır. Kullanıcıların bazı yüz resimlerine eğilimli olmaları sebebiyle (aynı ırktan veya güzel olduğu düşünülen yüz resimlerinin daha çok seçilmesi gibi) seçilen yüz resimlerinin tahmin edilebilir olması ve düşük şifre alanı gibi bazı güvenlik zayıflıkları bulunmaktadır. [28]

2. 3. 3 İpucuyla Hatırlamaya Dayalı Grafik Şifre Tasarıları

İpucuyla hatırlamaya dayanan yöntemler kullanıcının her bir tıklamasında tıklanılan şifrenin doğru olup olmadığı hakkında ipucu sağlayacak şekilde tasarlanmıştır. Cued Click Points (CCP)[29] bu yönteme güzel bir örnektir ve PassPoints yöntemine benzerlik göstermekle birlikte birden fazla resim kullanmak gibi bazı farklılıkları da bulunmaktadır. Bu yöntemde kullanıcı her bir noktayı farklı bir resim üzerinden seçer. İlk resim üzerinde seçilen bir nokta kullanıcıyı diğer bir resme yönlendirir[8] (Şekil 2.3.) ve kullanıcı seçeceği diğer noktayı bu resim üzerinden seçer, bu işlem bütün noktalar (5 adet nokta) seçilene kadar tekrarlanır. Her bir resim bir önceki resimde tıklanılan noktaya göre gösterilir ve her bir nokta farklı bir resim gösterilmesini sağlar, bu şekilde kullanıcı şifresini tıklarken karşılaştığı resimlerden çıkarım yaparak doğru yolda olup olmadığı hakkında ipucu elde eder.



Şekil 2.3 Cued Click Points Grafik Şifre Yöntemi

2. 4 Şifre Alanı

Şifre alanı daha önceden de bahsettiğimiz gibi sunulan sistemde oluşturulabilecek şifre sayısını ifade etmektedir. Fakat şifre alanı kavramının sistemin pratikteki kullanımı da göz önünde bulundurularak, teorik şifre alanı ve etkili şifre alanı olarak iki farklı yönden incelenmesi daha uygun olur. Bu ayrımın yapılmasının sebebi şifre elementlerinin (ikon, resim üzerindeki alan veya karakter gibi) kullanıcılar tarafından tamamen eşit algılanmaması ve eşit davranılmaması sebebiyle bazı elementlerin daha az veya hiç seçilmemesidir. Örneğin kullanıcılar Passpoints gibi bir grafik şifre modelinde kullanılan resim üzerindeki bütün noktalara karşı aynı tutum içerisinde olmazlar ve bazı noktalar (bir deniz manzarasında durgun bir deniz yüzeyi, bulutsuz gökyüzü veya desensiz yer, duvar dokusu) neredeyse hiç tıklanmaz. Bu durum teorik olarak hesap edilen şifre alanının pratikte kullanılamaması demektir. Passpoints sisteminde kullanılan 451 x 331 piksel boyutunda ve 20 x 20 piksellik kareler oluşturmak üzere ızgara ile bölünmüş bir resim üzerinde tıklanabilir 373 bölge oluşmaktadır. Bu sistemde 5 tıklama istendiği için 373^5 =7, 2 x 10^{12} büyüklüğünde bir şifre alanı oluşmaktadır. Fakat pratikte kullanılan şifre alanı ise kullanıcılara ve kullanılan resme göre farklılık göstermektedir. Passpoints yönteminde kullanılan havuz kenarında insanların ağaçların ve eşyaların da bulunduğu bir resim için etkili şifre alanının ne olabileceğini incelediğimizde bazı bölgelerin şifre içerisinde yer almak üzere uygun olmadığını görürüz (Şekil 2.4). Bu gibi bölgeler diğer bölgelere göre içerisinde ayırt edici özellik taşımayan alanlardır. Bu alanlar (yaklaşık 60 bölge) çıkartılarak diğer alanların şifre içerisinde yer almak için uygun adaylar olduğunu varsayarsak elde ettiğimiz şifre alanı 3, 0 x 10^{12} olmaktadır. Butonlar ve metinler için yer alanı ihtiyacından dolayı boyutu küçük tutulan ve deney amacıyla şifre alanının küçük tutulduğu [23] bu resimde 60 kadar alanın elenerek şifre alanını düşürdüğü düşünülürse, seçilecek olan resme göre de değişen şifre alanı hayli değişken ve çoğu durumda da düşük olacaktır.

Şekil 2.4. Etkili Şifre Alanının İncelenmesi

Geliştirilen şifre yöntemleri için etkili şifre alanını kullanışlılığı düşürmeden maksimize etmek güvenliği sağlamak adına önemli bir hedeftir.

2. 5 Sıcak-nokta Problemi

Metin tabanlı şifrelerde kullanıcıların hatırlamayı kolay kıldığı için kolay tahmin edilebilir şifreleri seçmeleri bilinen bir problemdir ve yukarıda da bahsedildiği üzere bu durum etkili şifre alanını düşürmektedir. Metin tabanlı şifrelerde bazı karakterlerin şifre elemanı olarak kullanılmamasının veya diğerlerine oranla çok az kullanılmasının sebebi bu karakterlerin uygun birer şifre elemanı olmamasıyla değil, tamamen kullanıcı tercihleriyle ilgilidir. Grafik şifrelerde ise iki farklı sebepten dolayı bazı bileşenler çok az kullanılmakta veya hiç kullanılmamaktadır. Bu sebepler, tasarımdaki zayıflık ve kullanıcı tercihleridir.

Tasarım zayıflığı: Bir önceki başlıkta da örneklerle ifade edildiği gibi şifre tasarımının içerdiği bileşenlerden bazıları grafik şifre elemanı olmaya uygun aday

değilse (örneğin düzlükler gibi ayırt edici bir özellik içermiyorsa) kullanıcılar tarafından seçilmezler. Bu durum şifrelere karşı yapılacak olan saldırıyı kolaylaştırır.

Kullanıcı tercihleri: Şifre tasarımındaki bileşenler homojen olarak dağıtılmış olsa ve her bir bileşen şifre elemanı olmaya uygun bile olsa kullanıcılar tarafından tercih edilmeyebilirler. Bunun sebeplerinin kullanıcılardaki eğilimler, bileşenlerin insanlar arasındaki popülerlik farkları, bileşenlerin görsel kalitesi gibi birçok sebep olması mümkündür.

Bir grafik şifre tasarımında kullanıcılar tarafından herhangi bir sebeple bazı bileşenlerin (noktaların, resimlerin, nesnelerin) diğerlerine göre seçilmesi daha muhtemel ise bu bileşenler sıcak noktalardır (Hot Spot).

Grafik şifreler arasında dikkat çeken bir çalışma olan PassPoints [30, 31, 23] yönteminde tek bir arka plan resmi kullanılır ve sisteme giriş için resim üzerindeki 5 farklı noktanın (veya nokta merkez olmak üzere 19 x 19 piksellik bir karenin) doğru sırayla tıklanması gerekmektedir. PassPoints de kullanılan havuz resmi (Şekil 2.4) üzerinde yapılan bir güvenlik araştırmasına göre[32], sadece 15 kullanıcının en çok tıkladığı noktalar (sıcak-noktalar) tespit edilerek, bu noktaların oluşturabileceği tüm şifreler tespit edilmiş ve bir sözlük oluşturulmuştur. Bu sözlük kullanılarak 114 şifre içinden 30 adet şifre yani kullanıcı şifrelerinin %27 si başarıyla tahmin edilebilmiştir. Bu araştırma sıcak nokta probleminin göz ardı edilemeyecek kadar önemli bir problem olduğunu ortaya koymaktadır.

Sıcak nokta problemine bir çözüm önerisi olarak şifre seçimini kullanıcılara bırakmamak ve her kullanıcıya sistemin bir şifre ataması düşünülebilir fakat bu durumun şifre hatırlamada zorluk çıkarıp kullanışlılığı düşüreceği açıktır. Sistem atamalı şifrelerde hatırlama oranını dolayısıyla kullanışlılığı arttıracak bir yöntem bulunabilirse sıcak-nokta problemi kullanışlılığı düşürmeden tamamen ortadan kaldırılmış olur.

Grafik şifreler üzerine yapılan çalışmalar ve grafik şifrelerin problemlerinin tespitiyle birlikte çözüm önerileri sunulmaya ve incelenmeye devam ediyor. Grafik şifrelerin yeni bir kimlik kanıtlama sistemi olarak metin tabanlı şifrelere alternatif olması yapılan çalışmalar ve geliştirmelerle birlikte çok uzak görülmemektedir. Fakat geliştirilen grafik şifre yöntemi ne kadar güvenli ve kullanışlı olsa da bu sistemlerin kabul görmesi ve yaygınlaşması için aşılması gereken bir engel daha vardır. Bu engel kullanıcı alışkanlıklarıdır. İnsanların alışkanlıklarını değiştirmeleri ve yeni bir sisteme alışmaları zordur. Bu sebeple geçiş aşamasında insanları yeni grafik şifre sistemlerine mecbur bırakmak yerine kullanımda olan metin tabanlı sistemlerle bütünleşik olarak çalışabilecek bir gerçekleştirimin daha uygun olacağını düşünmekteyiz. Bu sayede kullanıcılar yeni sistemi deneme amaçlı veya uzun süreli kullanım amaçlı olarak kolaylıkla kurup kullanabileceklerdir. İstedikleri takdirde çok fazla değişiklik gerektirmeden kurulumu geri alıp eski şifre girme yöntemlerine geri dönebileceklerdir. Böyle bir gerçekleştirim internet tarayıcılarına yönelik yazılacak bir eklenti ile mümkün olmaktadır ve bu sebeple biz de Mozilla Firefox internet tarayıcısıyla uyumlu çalışacak bir eklenti geliştirdik. Bu sayede yeni yöntemi denemek isteyen kullanıcılar kolaylıkla deneme imkânı bulacaklardır.

Eklenti olarak gerçekleştirdiğimiz grafik şifre yönteminde kullanılacak olan grafik şifre tasarısının seçilmesi ve yapılan uyarlamalar Bölüm 3'de anlatılacaktır.

BÖLÜM 3 – EKLENTİ İÇİN UYGUN YÖNTEMİN SEÇİLMESİ

Bölüm 2 de üç farklı kategori altında incelenen grafik şifre yöntemlerinden bazı örnekler gösterdik. Gerçekleştirdiğimiz uygulamada farklı grafik şifre kategorilerinden seçenekler sunmanın uygun olacağını düşündük ve hem hatırlamaya dayalı hem de tanımaya dayalı grafik şifre yöntemlerinden amacımıza uygun olan yöntemleri seçmek ve geliştirmek için çalışmalar yaptık.

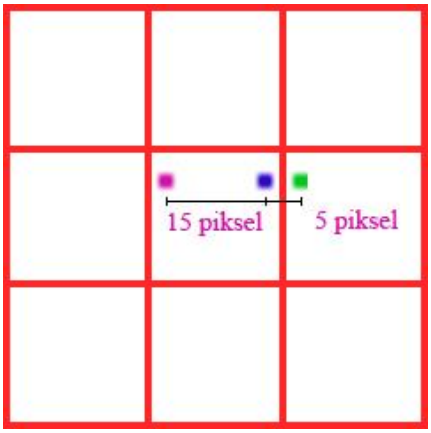
Önceki bölümlerde de bahsettiğimiz üzere PassPoints tasarısı başarısı kabul gören ve bu sebeple hakkında birçok çalışma yapılan bir yöntemdir. Biz de eklentimizde kullanmak üzere uygun bir aday olduğunu düşündüğümüz bu yöntemi inceledik. Grafik şifre yöntemlerinin tamamını güvenlik ve kullanışlılık açısından inceleyerek bir katalog oluşturmak ve yöntemler arasından en umut vaat edeni seçmek bu tezin amaçları arasında değildir fakat bunu amaçlayan bir çalışmanın[8] sonucuna göre de hatırlamaya dayalı PassPoints yöntemi güvenlik ve kullanışlılık açısından en umut vaat eden yöntemdir. Bu çalışmada [8] PassPoints yöntemi daha da geliştirilerek yeni grafik şifre tasarıları önerilmiştir. Bu tasarıların uygulamamızda kullanılabilme durumunu Bölüm 5'de inceleyeceğiz

3. 1 PassPoints Tasarısının incelenmesi

PassPoints yöntemi artalan resmi olarak seçilen bir resim üzerinde sırasıyla 5 nokta seçilerek şifre oluşturmaya imkân sağlayan bir yöntemdir. Sisteme giriş yapabilmek için kullanıcılar daha önceden belirledikleri noktaları yeniden aynı sırayla seçmelidirler fakat tıklanılan nokta 1 piksel olduğu için aynı noktanın tıklanmasını beklemek doğru bir yaklaşım değildir, bu sebeple uygun bir tolerans aralığı belirlemek gereklidir. Sisteme giriş zamanında tıklanılan noktanın kabul edilebilir değerde olup olmadığını (tolerans aralığında olup olmadığını) anlayabilmek için sunulabilecek en basit çözüm orijinal noktayı sistemde tutup sonradan tıklanılan

nokta ile karşılaştırmaktır. Şifreyi oluşturan orijinal noktanın sistemde açık olarak tutulması veya bu noktayı elde etmemize imkân verecek bir değer tutulması güvenlik açısından kabul edilemez. Şifreyi oluşturan noktaları açıkça sistemde tutmadan tıklanılan noktaların tolerans aralığında bulunup bulunmadığını şu şekilde anlayabiliriz:

2 boyutlu düzlemde tıklanılan bir noktanın etrafındaki r piksellik bir alana düşen noktalar kabul edilecekse tolerans aralığı r dir. Bu yöntemde resim üzerinde görünmez sabit bir ızgara vardır ve bu ızgara üzerindeki karelerin genişlik ve yükseklikleri 2r kadardır. Kullanıcıların ilk başta tıkladıkları nokta hangi karenin içine düşüyorsa, daha sonraki tıklamalarda da aynı kare içerisine düşmelidir. Bu sistemde bir kare içerisine düşen pikseller bu kare ile ilişkilendirilir ve hangi nokta tıklanırsa tıklansın şifre için kullanılacak değer içinde bulunulan karenin değeri olur. Akla gelebilecek basit bir yöntem olan bu yöntemde Şekil 3.1'de gösterilen "Kenar Problemi" vardır. Kenar problemi şu şekilde açıklanabilir: Resim üzerinde 20 x 20 piksellik kareler olsun ve seçilen nokta kare çizgisine yakın olarak seçilmiş olsun. Eğer kullanıcı sonraki girişlerde bu noktaya 5 piksel uzaklıkta fakat yandaki kareye düşen noktayı seçerse, doğru karenin değeri kullanılamayacağı için şifre kabul edilemez. Bu sistemde tolerans aralığı 10 pikseldir ve bu durum bir noktanın x ve y düzleminde 10 piksellik uzağına kadar görünen noktaların kabul edilmesini gerektirir. Fakat bu örnekte 5 piksellik bir fark yanlış şifre üretilmesine sebep olmuştur. Bu durum "hatalı ret" olarak adlandırılmaktadır. Örnekte bahsettiğimiz kenara yakın seçilen noktaya 10 pikselden daha uzak olan noktaların doğru bir tıklama olarak kabul edilmemesi gerekmektedir fakat bu noktanın içinde bulunduğu 20 x 20 piksellik kare içerisinde bu noktaya 10 pikselden daha uzak noktalar bulmak mümkündür. Bu noktalardan birisi tıklandığında sistem doğru karenin değerini kullanacağı için şifre doğru kabul edilir fakat tolerans aralığından daha uzaktaki bir nokta tıklanmıştır. Bu durum da "hatalı kabul" olarak bilinmektedir.



Şekil 3.1 Kenar Problemi : Şekilde, ortada görünen mavi nokta, kullanıcının tıkladığı orijinal noktadır. Bu tasarıda tolerans aralığı x ve y düzleminde +10, -10 piksel olarak düşünülmüştür. Hatalı Ret: Sağdaki yeşil nokta kullanıcının şifresini girmek için tıkladığı fakat 5 piksel sapmayla seçtiği noktadır. +10 tolerans aralığındaki bir tasarının bu girişi doğru olarak kabul etmesi gerekirken, bu tasarıda tıklama başka bir kare içine kaydığı için giriş başarısız olarak kabul edilmektedir. Hatalı Kabul: Soldaki pembe nokta kullanıcının şifresini girmek için tıkladığı fakat 15 piksel sapmayla seçtiği noktadır. +10 tolerans aralığındaki bir tasarının bu girişi kabul etmemesi gerekirken, bu tasarıda tıklama aynı kare içine yapıldığı için giriş doğru kabul edilmektedir.

Merkezî Ayrıklaştırma (Centered Discretization): Hatalı kabul ve hatalı ret durumları tıklanılan noktanın karenin tam merkezinde bulunmamasından kaynaklanmaktadır. Buna bir çözüm olarak sunulan merkezî ayrıklaştırma tıklanılan her noktayı bir karenin merkezine alacak şekilde her nokta için görünmez ızgara oluşturmaktadır. Bunun için sunulan çözüm temel olarak şu şekildedir:

Resim üzerinde yapılan tıklamalar için anlatılacak olan bu yöntemde sadece x düzlemi göz önüne alınmıştır, y düzlemi için de aynı şekilde uyarlanabilir. Resim üzerinde bir nokta tıklamış olalım ve tıklanılan bu noktanın x düzlemindeki konumunu x olarak ifade edelim. Amacımız bir r tolerans aralığı için x i her seferinde merkezde bırakacak şekilde bu resmi bölgelere ayırmaktır. Bu sayede x noktasının her iki tarafından eşit bir tolerans aralığı bırakmış oluruz ve bu durum hatalı kabul ve hatalı ret durumlarının oluşmasına engel olur. Tıklanılan x noktasının merkeze geleceğini ve tolerans aralığının r olduğunu düşündüğümüzde her bir bölgenin genişliğinin 2r olması gerektiğini görebiliriz. Tıklanılan x noktasının tam ortada bulunduğuna emin olmak için 0 ile 2r noktaları arasındaki bölgeyi merkeze olan göreli uzaklık bölgesi olarak belirlemeliyiz ve göreli konumu da d ile ifade ederiz.

Bir x noktasının yerini doğru olarak kaydedebilmek için önce bu nokta için bir göreli konum değeri d, ($0 \leq d < 2r$), ardından x noktası hangi bölgede yer alıyorsa o bölgenin işaretçisini hesaplamalıyız (bölge işaretçisi $i \geq -1$, Eğer x merkezden r uzaklığına kadar bir mesafede ise $i = -1$ olur). Göreli konum değeri olan d olduğu gibi saklanabilir fakat bölge işaretçisi olan i değeri d değeri ile özet fonksiyonundan geçirildikten sonra saklanmalıdır. Bölgeyi benzersiz bir değerle tanımlamak için özet fonksiyonunda i ve d değerleri birlikte bulunmaktadır. Şifre ikinci defa girileceğinde ne kadar bir tolerans ile girilen değerlerin kabul edileceğini belirlemek için tolerans değeri olan r de sistemde tutulmalıdır. Bölge işretçisi olan ve x in içinde bulunduğu bölgeyi işaret eden i değeri denklem 3.1 ile hesaplanır.

$$i = \lfloor (x - r) / 2r \rfloor \tag{3. 1}$$

Göreli konum değeri d ise denklem 3.2 ile hesaplanır ve merkezden (resmin başlangıcı olan en solundaki sınırından) 0. bölgenin sol sınırına kadar olan uzaklığı ifade eder.

$$d = (x - r) \bmod 2r \tag{3.2}$$

Orijinal tıklama noktası olan noktayı x ile ifade ediyoruz, buradan sonra ikinci ve sonraki girişlerde tıklanılan noktaları X ile ve X in tıklandığı bölgeyi gösteren işaretçiyi de I ile ifade edeceğiz. Tıklanılmış olan X noktasının kabul edilebilir olup olmadığını anlamak için sistem denklem 3.3 ü kullanarak I değerini bulmalıdır.

$$I = \lfloor (X - d) / 2r \rfloor \tag{3.3}$$

Bu değer X noktasının hangi bölge içerisinde olduğunu gösterecektir(kullanılan göreli konum değeri olan d ve tolerans değeri olan r orijinal x değeri için kullanılan değerlerle aynı olduğundan dolayı, bölgelerde bir sapma olmadan doğru bölgeyi bulabiliyoruz). Eğer X, orijinal x noktasının en fazla r kadar uzağında ise I = i

olacaktır ve h(I, d) değeri de önceden hesaplanarak saklanmış olan h(i, d) değerine eşit olacaktır. Bu durumda sistem girişi doğru olarak kabul edecektir. Eğer X, x den kabul edilebilir tolerans değeri olan r değerinden daha uzak bir noktada ise başka bir bölge içerisinde yer alıyordur ve bu durumda I ≠ i olacaktır. Bunun sonucu olarak h(I, d) ≠ h(i, d) olduğu için sistem girişi doğru olarak kabul etmeyecektir.

Örneğin, x=83 ve r=5 olsun. Bölge işaretçisi olan i = (83-5)/10 = 7 olacaktır ve göreli konum değeri olan d= (83-5) mod 10 = 8 olacaktır. Sistemde saklanacak değerler, göreli konum değeri d=8 ile özet fonksiyonuyla korunmuş olan h(i, d) = h(7, 8) dir. Kullanıcı diğer girişlerinin birinde orijinal noktadan 4 piksel uzaklıkta olan X=79 noktasını tıklarsa sistem bölge işaretçisini I= |(79-8)/10|=7 olarak bulacaktır. Daha sonra sistemde saklanmış olan h(i, d) ile h(I, d) değerlerini karşılaştıracaktır ve I=i olduğu için, ayrıca kullanılan göreli konum değeri de aynı olduğu için eşitlik olduğundan dolayı girişi kabul edecektir. Sistemde sadece bir nokta değil birden fazla nokta tıklanılacağı için bölge işaretçilerini ayrı ayrı özet fonksiyonundan geçirmek yerine tüm bölge işaretçilerini art arda ekleyerek birlikte özet fonksiyonundan geçirmek, noktaların tek tek ele alınıp onlara karşı yapılacak bir atağı önlemede etkili olur[36].

Yukarıdaki anlatımda da görülebileceği gibi tıklanılan noktaların karşılaştırılabilmesi için sistemde göreli konum değeri ve bölge işaretçisi ile birlikte göreli konum değerinin özet fonksiyon sonucu tutulmalıdır.

Merkezî ayrıklaştırma yöntemi avantajlı bir yöntem olarak görünse de gerçekleştireceğimiz uygulama için yeterince uygun bir yöntem değildir. Merkezî ayrıklaştırma yöntemi uygulamamız için kullanışlılık ve güvenlik ile ilgili olmak üzere iki farklı problem teşkil etmektedir.

3. 1. 1 Güvenlik ve Taşınabilme Problemleri

Güvenlik Zayıflığı: Göreli konum değerlerini açık olarak tutmak, tıklanılan noktaların değerlerini açıktan tutmak kadar tehlikeli bir güvenlik açığı değildir fakat

tutulan bu değerlerin saldırganlara sistem hakkında analizler yapma imkânı verebileceği ve saldırılarını kolaylaştırabileceği ihtimalleri göz ardı edilmemelidir. Sistem hakkında bilinenler düşünülürse elimizde göreli konum değeri (d), tolerans değeri (r) ve 15 kadar kullanıcıyla kısa bir deney sonucunda elde edilebilecek Sıcak-Noktalar olduğu gözlemlenebilir. Bu durum sadece Sıcak-Noktaları değerlendirerek saldırı yapmayı amaçlayan bir saldırganın işini fazlasıyla kolaylaştıracaktır. Çünkü artık saldırgan bütün Sıcak-Noktaları denemek yerine göreli konum değerini kullanarak çizeceği ızgara üzerinde kare merkezlerine denk gelen Sıcak-Noktalar üzerine yoğunlaşacaktır.

Taşınabilme Problemi: Uygulamamız için merkezî ayrıklaştırma yöntemini tercih etmemize engel olan diğer bir konu da kolay taşınabilir bir uygulama gerçekleştirmek istememizdir. Gerçekleştireceğimiz uygulama sadece bir internet sitesinde değil, şifre ile kimlik kanıtlama yöntemini kullanan tüm internet sitelerinde kullanılmak üzere tasarlanmıştır. Bu sebeple internet sitesiyle bütünleşik olarak sunucu tarafında saklanacak bir uygulamanın sahip olduğu, göreli konum değeri gibi kullanıcıya özgü bilgileri saklama imkânından mahrum durumdayız. Uygulamamızı bir internet tarayıcısı eklentisi olarak gerçekleştireceğiz ve bir kullanıcı sürekli aynı bilgisayarı kullanmadığı için kullanmak istediği diğer bilgisayarlara da kendisine özgü bilgilerini taşımak zorunda kalacak. Bu ise kullanımı zorlaştıran büyük bir kullanışlılık problemidir.

Herhangi bir kişisel bilgi taşıma ihtiyacı olmadığı durumlarda da uygulamanın diğer bilgisayarlara kurulması yine bir gerekliliktir fakat kullanıcıya özgü bilgileri taşımak kadar zor olmamakla birlikte eklenti kurulması standart ve kolay bir işlemdir. Eklenti kurulamadığı durumlarda kullanılabilecek alternatif yöntemler Bölüm 4'de ayrıntılı olarak anlatılacaktır.

Burada bahsedilen taşınabilirlik ve güvenlik problemleri sebebiyle resmimizin üzerine sabit bir ızgara bulundurarak kullanmayı uygun gördük. Fakat resim üzerinde bulunan sabit ızgaranın görünür olması kullanıcılara bir rehber olabileceği için ve karelerin içini tıklamaları gerektiğini bilen kullanıcılar uygulamayı daha az

hata ile kullanabileceklerinden dolayı resim üzerine görünür bir ızgara çizmeyi uygun gördük. Izgara çizgilerinin etkililik, verimlilik, kullanıcı memnuniyeti ve güvenliği nasıl etkilediğini görmek için bu konuda bir deney yürüttük.

3. 2 PassPoints tasarısının eklenti için uyarlanması

3. 2. 1 Deney

Yürüttüğümüz bu deneyin amacı eklentimizde kullanmak üzere seçtiğimiz PassPoint yönteminin üzerinde yapılan uyarlamaları ve etkilerini gözlemlemektir. Bu deneyde PassPoints yönteminde kullanılan resimler arasından bir otopark sahnesini gösteren resim kullanılmıştır[23]. Bu resim üzerine ızgara çizilmeden ve ızgara çizilerek iki farklı deney grubu üzerinde denenmiştir ve deneklerin hatırlama aktiviteleri, veri giriş süreleri karşılaştırılmıştır.

İki farklı deney grubu için hazırlanan deney uygulamasında 451 x 331 piksel büyüklüğündeki otopark resmi kullanılmıştır. Bir uygulamada resmin üzerine 20 x 20 piksel boyutunda kareler oluşturacak şekilde bir ızgara çizilmiştir ve kullanıcılardan gördükleri karelerin içine tıklamaları istenmiştir. Tam çizgi üzerine tıklanması durumu olabileceği için karelerin sağındaki ve aşağısındaki çizgiler karelerin içi gibi kabul edilmiştir. Diğer resimde ise bu ızgara yoktur. Kullanıcılardan şifre olarak 5 adet nokta seçmeleri beklenmektedir.

Etkililik faktörü şifresini unutan kullanıcıların sayısı, şifreyi hatırlamak için yaptıkları deneme sayıları ve şifreyi doğru girme sayıları ile belirlenir. Verimlilik ise, kullanıcıların şifre üretme ve onaylamadaki harcadıkları süre ile ölçülmüştür. Deney sonunda yapılan anketle kullanıcı memnuniyeti değerlendirilerek, uygulamanın güvenliğini test etmek için ise grafik şifreler için büyük önem taşıyan Sıcak-Nokta analizi yapılmıştır.

TOBB Ekonomi ve Teknoloji Üniversitesinde(TOBB ETÜ) gerçekleştirilen deneye TOBB ETÜ öğrencileri ve çalışanları gönüllü olarak katılmışlardır ve yaş ortalaması 22, 5 olan katılımcılardan 24 ü erkek, 22 si bayandır. İki adet deney yapılacağı için kullanıcılar dengeli bir şekilde dağıtılmaya çalışılmıştır. Izgaralı resim için yapılan deneye 11 erkek, 12 bayan katılırken, ızgarasız resim için yapılan deneye 13 erkek, 10 bayan katılmıştır.

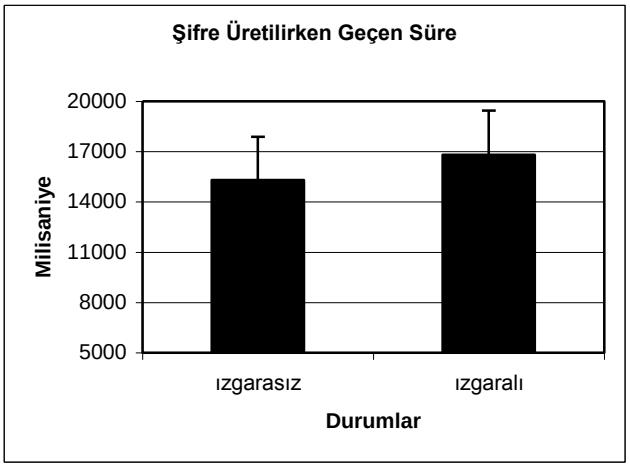
Kullanıcılar deneye başlamadan önce deney hakkında bilgilendirildiler ve ilk defa deneyecekleri bir sisteme alışmaları gerektiğini düşündüğümüz için küçük bir alışma uygulaması yaptılar. Bu uygulama gerçek deney uygulamasına benzerlik gösteren fakat kullanıcıların deney resmini erkenden görüp etkilenmemeleri için farklı resim kullanılan bir uygulamadır. Bir masaüstü bilgisayar karşısına oturan kullanıcılar uygulama üzerindeki uygun düğmeleri de kullanarak ilk başta farklı 5 noktadan oluşan bir şifre belirlediler. Kullanıcılar resim üzerine tıkladıkça tıklama sayısı resmin altında gösterildi ve kullanıcılar buradan kaç tıklama yaptıklarını ve yapacakları tıklama sayısını görerek giriş yaptılar. Her iki grup için de uyarlanan ve birisinde ızgaralı, diğerinde ızgarasız resim gösterilen alışma uygulamasında şifresini oluşturup onaylayamayanlar başarılı olana kadar yeniden deneyerek sisteme alıştılar.

Gerçek deneye hazır olan kullanıcılar için yeni bir oturum açıldı ve kullanıcılar daha sonra hatırlamaları gereken şifrelerini belirleyerek aynı noktaları ikinci bir defa daha tıklamak suretiyle onayladılar. Her bir kullanıcı en az 4 gün sonra deney laboratuarına davet edildiler. İkinci oturumu yapılan deneyde kullanıcılar önceden belirledikleri 5 noktadan oluşan şifrelerini hatırlayarak giriş yapmaya çalıştılar. Kullanıcılardan şifrelerini başarılı olarak giremeyenler doğru şifreyi girene kadar deneme veya denemekten kendi istekleriyle vazgeçme imkânına sahiptiler. Deney sonunda her kullanıcıya bir anket yapılarak kullanıcıların düşünceleri öğrenildi.

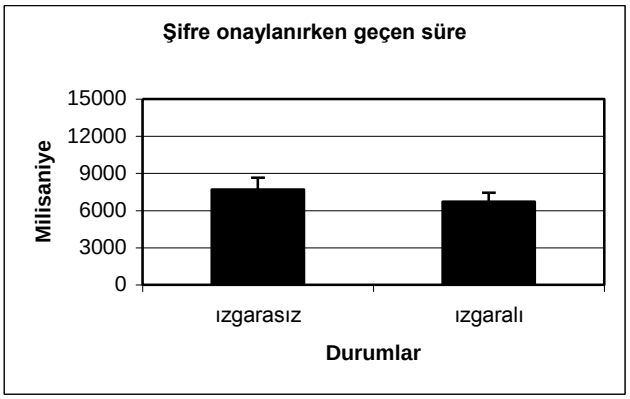
3. 2. 2 Deney Sonuçları

Bu deneyin sonucu olarak önerilen sistemin etkililiğini, verimliliğini ve kullanıcı memnuniyetiyle birlikte güvenliğini inceleyeceğiz.

Şifre üretme (Şekil 3.2) ve onaylama (Şekil 3.3) sırasında harcanılan süre bize verimlilik hakkında bilgi vermektedir.



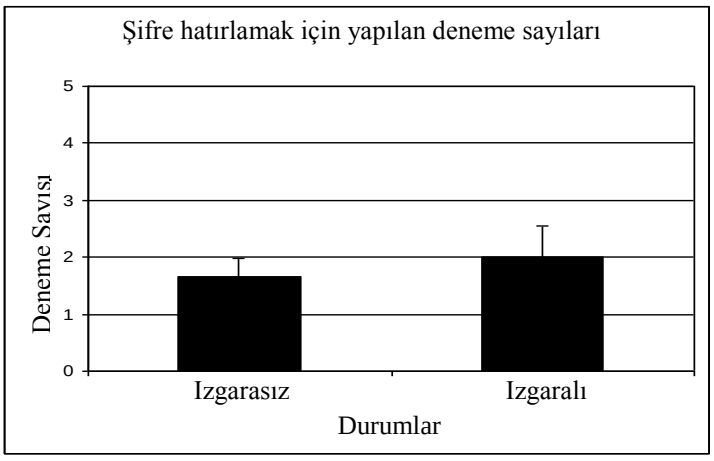
Şekil 3.2 Izgaralı ve ızgarasız durumlarda şifre üretmek için harcanan süre.



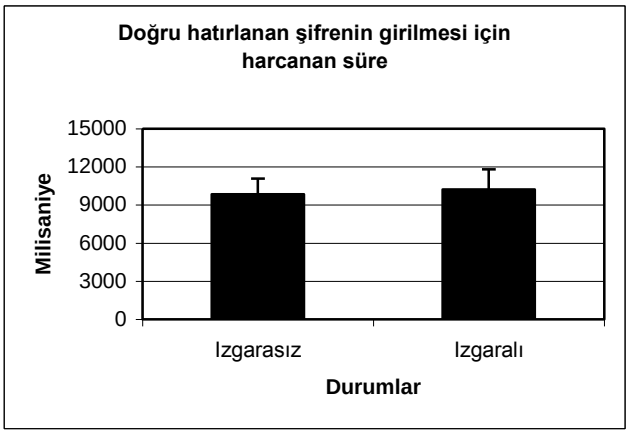
Şekil 3.3 Izgaralı ve ızgarasız durumlarda şifre onaylamak için harcanan süre.

Şifre onaylama ve üretme arasında ızgaralı ve ızgarasız deney grupları arasında önemli bir fark gözlemlenmemektedir (t(44) = -0. 41, p>0. 05 şifre üretme; t(44) = 0. 83, p>0. 05 şifre onaylama).

Etkililik ise şifresini unutan kullanıcı sayısıyla ve şifre hatırlamak için yaptıkları deneme sayısıyla ve şifreyi doğru girme sayısıyla belirlenir. Izgaralı grupta % 17lik bir dilimi oluşturan 4 kullanıcı ve ızgarasız grupta ise % 22lik bir dilimi oluşturan 5 kullanıcı şifrelerini unutmuştur. Gruplar arasında önemli bir fark gözlemlenmemektedir (X^2(1) = 0. 138, p>0. 05). Doğru şifreyi hatırlamak için yapılan deneme sayıları Şekil 3.4'de gösterilmiştir ve grafikten de görülebileceği gibi gruplar arasında önemli bir fark yoktur(t(35) = -0. 52, p>0. 05). Doğru şifreyi hatırlamak için harcanan süreyi gösteren Şekil 3.5 ye bakılırsa yine gruplar arasında önemli bir fark olmadığı görülebilir(t(35) = -0. 19, p>0. 05).



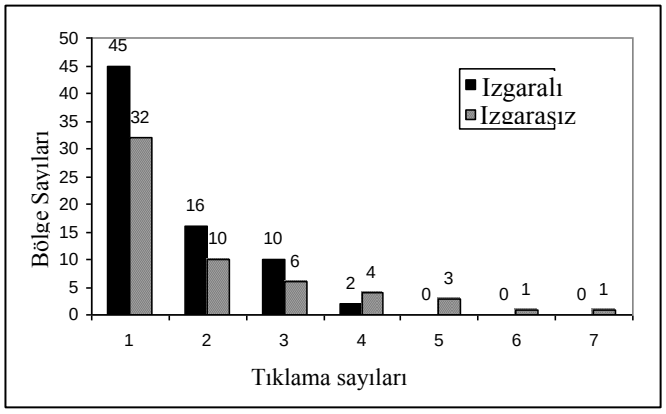
Şekil 3.4. Izgaralı ve ızgarasız durumlarda doğru şifreyi hatırlamak için yapılan deneme sayısı



Şekil 3.5 Izgaralı ve ızgarasız durumlarda doğru hatırlanan şifrenin girilmesi için geçen süre

Kullanıcılara memnuniyetlerini ölçebilmek için bir anket yapılmıştır, ızgaralı ve ızgarasız grupların cevapları Mann-Whitney U test kullanılarak karşılaştırılmıştır. Karşılaştırma sonucu kullanıcı memnuniyeti açısından önemli bir fark bulunamamıştır. Sonuç olarak resim üzerine ızgara çizmenin kullanıcı memnuniyeti ve kullanışlılığı değiştirmediği anlaşılmıştır.

Yapılan değişikliğin güvenlik açısından getirebileceği değişiklikleri ölçmek için ise sıcak-nokta analizi ve karşılaştırması yaptık. Her iki farklı grupta 23 kişi yer almıştır ve 23 kişinin toplam tıklama sayısı 115 dir. Şekil 3.6 tıklanma sayılarına göre bölge sayılarını göstermektedir. Örneğin, ızgaralı resim üzerinde sadece 1 defa tıklanılmış olan 45 bölge vardır ve sadece 2 defa tıklanılan 16 bölge görülmektedir. Izgarasız resim üzerinde eğer bir nokta diğer bir noktanın tolerans aralığında ise bu iki noktanın aynı bölgede olduğunu kabul ettik.



Şekil 3.6 Izgaralı ve ızgarasız durumlar için resim üzerindeki bölge sayıları ve tıklama sayıları.

Burada gösterilen ızgaralı ve ızgarasız grupların grafiğini şu şekilde karşılaştırdık: 3.4 deki formül her bir bölgenin beklenen tıklanma değerini verir.

$$E_r = \binom{115}{r} \times \left(\frac{390}{391}\right)^{115-r} \times \left(\frac{1}{391}\right)^{r} \times 391 \qquad (3.4)$$

E_r "r" defa seçilen bölgelerin sayısını belirtmektedir. Toplamda 115 tıklama sayısı bulunduğu için, bir bölgenin "r" defa seçilme ihtimali 3.5 deki gibidir.

$$\left(\frac{1}{391}\right)^{r} \times \left(\frac{390}{391}\right)^{115-r} \qquad (3.5)$$

Seçim sırası da önemli olduğu için $\binom{115}{r}$ i de hesaba kattık ve beklenen değeri bulmak için bunu toplam bölge sayısı olan 391 ile çarptık.

Eğer her bir kullanıcı tıklaması bağımsız bir rastgele eylem ise bu formülü kullanarak 86 bölgenin 1 defa, 13 bölgenin 2 defa ve 1 bölgenin 3 defa tıklanmış olması gerektiğini buluruz. Buradan çıkarak ideal bir durumda 100 farklı bölgenin tıklanmış olması gerektiğini buluruz.

Hesapladığımız beklenen değer ile Şekil 3.6. de gösterilen değerleri karşılaştırarak hangi resmin daha fazla sıcak-nokta içererek güvenlik açısından daha zayıf olduğunu tespit ettik. Şekil 3.6. yi kullanarak ızgaralı resimde 73 ve ızgarasız resimde 57 farklı bölgenin tıklanmış olduğunu görüyoruz. Izgarasız resimde tıklamalar daha çok belli bölgelerde yoğunlaştığı için resmin üzerine ızgara çizmiş olmanın sıcak-nokta probleminde fayda sağladığı yorumunu yapabiliriz. Bu faydanın bir sebebi şu şekilde açıklanabilir: resim üzerine ızgara çizildiğinde resimdeki araçlar, bitkiler ve renkler gibi ayırt edici özelliklerin yanı sıra bir de ızgara çizgileri kullanıcılara yön göstererek şifre seçme ve hatırlamada etkili olmuştur. Diğer bir deyişle, görünür ızgara çizgileri birçok seçenek arasından noktaları seçmede yardımcı olmak üzere resmi zenginleştirmiştir.

Yaptığımız deney sonucunda uygulamamızda kullanmayı düşündüğümüz resim üzerine görünür ızgara çizgilerinin çizilmesi kullanışlılık ve kullanıcı memnuniyetini olumsuz yönde etkilememekle birlikte sıcak nokta sayısını da azaltarak güvenliğe katkı sağlamıştır ve uygulamamızda kullanmak için uygundur.

Uygulamamızda yukarıda bahsedilen taşınabilirlik problemini resim üzerine ızgara eklemek suretiyle çözerek kullandık fakat önemli bir problem olarak görülen sıcak nokta problemine karşı (resme ızgara çizmenin getirdiği bir fayda olan sıcak-noktaların azalması dışında) özel bir çözüm uygulamadık. Bu sebeple bu yöntemin sıcak-nokta problemi hala önemli bir problemdir. PassPoints yönteminin sıcak-nokta problemine karşı önerilen Cued Click-Points[29] ve Persuasive Cued Click-Points[25] yöntemlerini ve bu yöntemlerin uygulamamızda kullanılabilirliğini Bölüm 5'de inceleyeceğiz.

Uygulamamızda kullanacağımız hatırlamaya dayalı yöntemimizi bu şekilde belirledikten sonra bu yöntemde bulunan sıcak-nokta probleminin en az ihtimalle bulunduğu veya tamamen giderildiği tanımaya dayalı iki yöntem önererek bu yöntemlerin kullanışlılık ve güvenlik deneylerini yaptık.

3. 3 GPI-GPIS Tasarıları

Tek bir artalan resmi kullanılan ve resim üzerinde belli sayıda noktalara tıklanılan yöntemlerde sıcak-nokta problemi kullanılan resme göre değişiklik göstermektedir ancak soyut şekiller veya arabalar, ataçlar gibi aynı türde nesneler kullanılmış bile olsa bu problem hala önemli ölçüde devam etmektedir[32]. Bu sebeple bir resim üzerinden tıklanılarak seçim yapılan bu yöntemden farklı olarak ve sistemi kullanışlı tutarak sıcak-nokta problemini çözmede başarılı olabilme potansiyeli taşıyan iki farklı tasarı (**G**raphical **P**assword with **I**cons, **G**raphical **P**assword with **I**cons suggested by the **S**ystem) önerdik.

Bu tasarılar tıklanılacak nokta olarak ikonlar içermektedir ve önceki sistemlerle (örneğin PassPoints[23] de olduğu gibi 2^{43}) aynı şifre alanını sağlamaktadır. Açıklayacağımız ilk grafik şifre yöntemi GPI (**G**raphical **P**assword with **I**cons) yöntemidir. Bu yöntemde kullanıcı kendisine sunulan ikonlar arasından bir altkümeyi kendi şifresi olarak seçmektedir. Deney sonuçları gösteriyor ki, artalan resmi yerine ikonların kullanılması mümkün olan tıklama noktalarının eşit olarak dağılmasını sağlamaktadır, bu sayede Bölüm 2'de bahsettiğimiz etkili şifre alanının teorik şifre alanına yakınlaşması mümkün olmuştur, gösterilen ikonların da mümkün olduğunca birbirine eşit olarak seçilmesiyle sıcak-nokta probleminin oldukça azaltılması hedeflenmiştir. Sıcak-nokta problemini tamamen ortadan kaldırmak için ise GPIS (**G**raphical **P**assword with **I**cons suggested by the **S**ystem) tasarısını önermekteyiz. GPIS tasarısında şifre olarak seçilecek olan ikon altkümesini sistem rastgele olarak seçerek kullanıcıya önermektedir. Eğer kullanıcı önerilen şifreden memnun değilse yeniden bir şifre üretilmesini isteyebilir.

3. 3. 1 Deney

Yaptığımız bir laboratuar çalışması ile GPI, GPIS ve PassPoints[23] yöntemlerinin kullanılabilirliğini karşılaştırdık. Deney sonuçlarımız gösterdi ki, şifresini unutan kişilerin oranı bakımından üç farklı tasarı arasında önemli bir fark yoktur fakat

PassPoints tasarısının şifre giriş süresi diğerlerinden daha kısadır. Deney sonuçlarımıza göre önerdiğimiz tasarıların güvenli ve kullanışlı grafik şifre çözümleri yolunda bir adım olduğunu söyleyebiliriz.

Tanımaya dayalı tasarıların internet uygulamaları için çok uygun olmadığı, daha çok birkaç denemeden sonra sisteme girişi engelleyen sistemler için uygun olabileceği düşünülmüştür. Örneğin, bir çalışmada [31] *"resim tanımaya dayalı olan bütün tasarıların bir dezavantajı, sadece sınırlı sayıda resim gösterebilmeleridir, örn. Bir tanesi seçilen resim olacak şekilde 9 resim"* denilmiştir. Biz bu çalışmamızda hatırlamaya dayalı tasarılar gibi geniş şifre alanına sahip ve daha üstün kullanışlılık ve güvenlik özellikleri bulunan tanımaya dayalı bir grafik şifre yönteminin nasıl tasarlanabileceğini araştırdık.

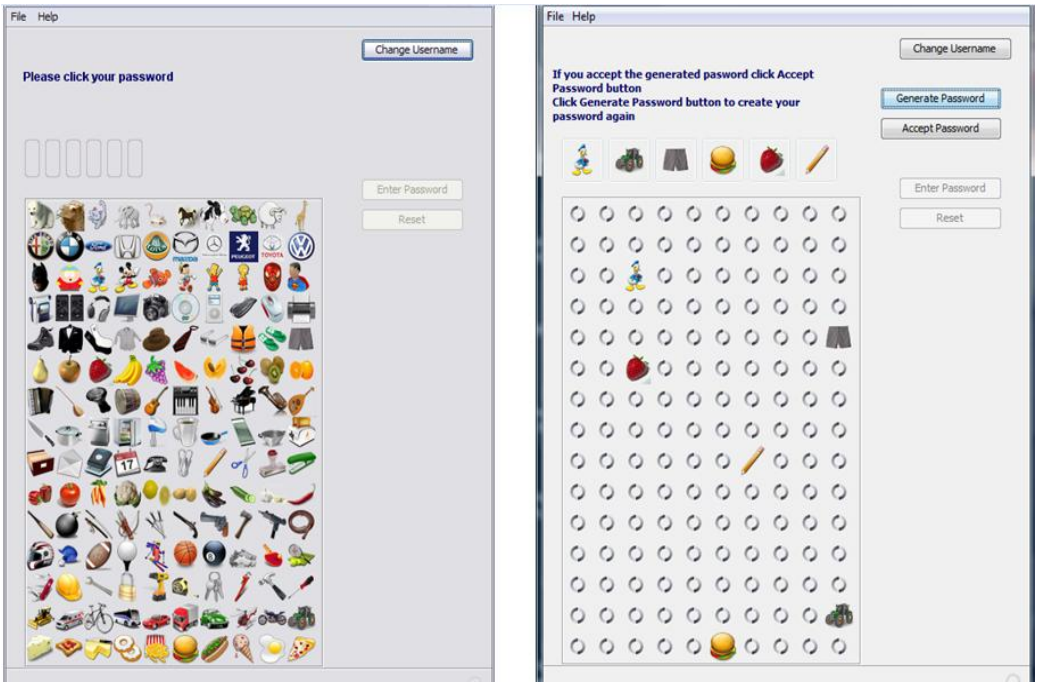
GPI ve GPIS için internet üzerindeki açık kaynak resim veritabanlarından faydalandık[44]. Sunduğumuz tasarıda ve PassPoints tasarısında şifre alanı yeterince büyük (2^{43}) olsa bile çevrimdışı ataklar [32] için hala yetersiz kalabilir, fakat bazı şifre genişletme teknikleriyle [33, 17], yeterli seviyede şifre güvenliği sağlanabilir.

Sunduğumuz ikinci tasarı olan GPIS de şifre olarak belirlenen ikonlar sistem tarafından rastgele atanarak kullanıcıya sunulmuştur. Kullanıcı istediği takdirde üretilen bu şifreyi kabul eder veya sisteme yeni bir rastgele şifre ürettirebilir. Kullanıcı üretilen şifreyi kendi şifresi olarak seçmeye karar verene kadar yeni şifre üretilmesine devam edilebilir. Literatürde sistemin kullanıcıya şifre seçmede yardımcı olduğu ipucu ile hatırlama yöntemleri vardır. Örneğin bu tasarıda [25] sistem kullanıcının sıcak-noktalardan kaçınabilmesi için yardımcı olmak amacıyla resmin sadece sınırlı bir kısmından seçim imkânı sağlamaktadır. Sistemde bulunan bir "Karıştır" ("Shuffle") düğmesi şifre seçilebilecek sınırlı alanın yerini rastgele değiştirmektedir ve kullanıcı sadece bu alandan seçim yapabilmektedir. Bizim GPIS sistemimizde bulunan "Şifre Üret" ("Generate Password") düğmesi bu sistemdeki "Karıştır" ("Shuffle") düğmesine benzer bir mantıkla çalışmaktadır.

Şekil 2.1. de gösterildiği gibi artalan resim üzerine tıklama yapılan tasarılarda artalan resmi üzerinde bulunan fakat bir şifre elemanı olmak için uygun olmayan bazı dokular bulunması mümkündür. Bu sebeple teorik şifre alanı etkili olarak kullanılamamaktadır. Biz de GPI ve GPIS de artalan resmi yerine ikon kullanımını önererek etkili şifre alanını maksimum seviyeye çıkarmayı hedefledik. Çünkü farklı nesneleri gösteren ikonların her biri anlamlı ve ayırt edici özelliği bulunan bileşenler olduğu için sistemde görünen bütün ikonlar kullanılabilir durumdadır.

İkon kullanımının sebep olabileceği farklı türde bir sıcak-nokta durumuyla karşılaşılabilir. Bazı ikonların diğer ikonlara göre daha fazla dikkat çekmesi sonucu bu ikonlar kullanıcıların daha çok seçtiği ikonlar durumuna gelebilir. Kullanıcıların bazı ikonları diğer ikonlara tercih etmelerinin sebebi bazı ikonların daha ilgi çekici olarak çizilmiş olması veya ikonun simgelediği nesnenin kullanıcılara daha tanıdık gelmesi olabilir. Bunlar gibi sebepler sonucu farklı sıcak-ikon olarak adlandırılabilecek farklı bir tür sıcak-nokta durumuyla karşılaşılabilir. Bu sebeple öncelikle kullandığımız ikonların benzer stilde çizilmiş ikonlar olmasına özen gösterdik. Toplam 150 ikon kullandık ve bu ikonların her 10 tanesi bir kategori içinde yer alacak şekilde 15 farklı kategoriden ikonlar seçtik. İkonların hangi nesneleri gösteren ikonlar olacağını belirlemek için bir kategori norm çalışmasından faydalandık[34]. Bu çalışmada çalışmaya katılanlar tarafından oluşturulmuş farklı kategorilerden kelime listeleri bulunmaktadır. Çalışmada yüzlerce katılımcıya bazı kategorilerden (örneğin “mobilya” kategorisi) nesneler sorulmuştur ve katılımcıların akıllarına gelen nesneleri yazmaları istenmiştir. Daha sonra yazılan kelimelerin o kategori içinde bulunabilme olasılığını hesaplamışlardır.

Bu çalışmada 50 adet kategori listelenmiştir. Biz de bu kategoriler arasından 15 nesne kategorisini (hayvanlar, araba amblemleri, çizgi karakterler, elektronik eşyalar, elbiseler, meyveler, müzik enstrümanları, mutfak eşyaları, ofis eşyaları, meyveler, silahlar, spor türleri, tamir aletleri, araçlar ve yiyecekler), her bir kategoride 10 nesne olacak şekilde seçtik. Her bir kategoride akla kolayca gelebilecek olan fakat diğer nesnelere göre çok fazla veya çok az seçilme oranına sahip olmayan nesneleri seçtik, bu sayede nesnelerin tercih edilme ihtimallerini birbirlerine yakın bir seviyede tuttuk (Şekil 3.7.).



Şekil 3.7. GPI ara yüzü (sol) ve GPIS ara yüzü (sağ). GPI ara yüzünde kullanıcı ikonları kendisi seçerken GPIS ara yüzünde ise sistem seçer ve gösterir. (şekiller sayfaya sığması için küçültülmüştür.)

Tasarladığımız ara yüzde aynı kategoriye dâhil olan ikonlar aynı satırda gösterildiler. Passpoints ara yüzünde artalan resmi 451 x 331 piksel boyutundadır ve kullanıcılardan 5 noktayı tıklamaları istenir. Daha sonra tıkladıkları noktaları orijinal nokta merkez olmak üzere 19 piksellik bir tolerans aralığında onaylamaları beklenir. Bu tasarıda toplam 391 tıklanılabilecek bölge vardır ve bu P(391, 5) yani yaklaşık 2^{43} lük bir şifre alanı oluşturur. (P = Permutasyon).

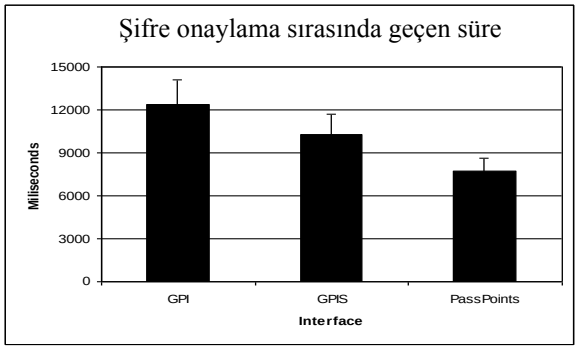
Şifre alanını Passpoints ile aynı seviyede tutabilmek için GPI tasarısında kullanıcıların 6 ikon seçmeleri ve bu ikonları aynı sıra ile tekrar seçerek onaylamaları istenmiştir ($P(150, 6) \approx 2^{43}$). GPI ve GPIS sistemlerinde ikon büyüklükleri 32 x 32 boyutunda tutulmuştur ve bu da ikonların kapladığı alanın Passpoints sistemindeki artalan resmine yakın olmasını sağlamıştır.

Sıcak-nokta problemini tamamen ortadan kaldırabilmek için GPIS sisteminde sistem tarafından seçilen ikonların şifre olarak atanması uygun görülmüştür. Bu tasarıda sistem 6 adet ikon belirler ve kullanıcıya bir animasyonla ikonların aralarına çizgi çizerek gösterir. GPI sistemi GPIS ile aynı ara yüzü kullanır fakat sadece şifre üretme aşaması yoktur. GPI, GPIS ve Passpoints arasında kullanışlılık ve güvenlilik karşılaştırması yapmak üzere bir deney gerçekleştirdik. Artalan resmi yerine ikonların kullanılmasının etkililiği, verimliliği, kullanıcı memnuniyetini ve güvenliği nasıl etkilediğini değerlendirdik. Bu deneyin yapılması için Ortadoğu Teknik Üniversitesi Etik Kurulundan gerekli onay alınmıştır.

Her bir kullanıcı GPI, GPIS veya Passpoints ara yüzlerinden birisini kullanmıştır ve hepsinin de onaylama safhasında tıklanılan sırayla onaylamaları istenmiştir. Bir hafta sonra tekrar çağırılan kullanıcılardan şifrelerini yeniden girmeleri istenmiştir. Deneye TOBB ETÜ ve ODTÜ deki öğrenci ve çalışanlardan toplam 69 kişi katılmıştır. Katılımcılar üç farklı tasarıdan herhangi birisine rastgele olarak atanmıştır ve GPI için 23, GPIS için 23 Passpoints için 23 katılımcı vardır. Deney masaüstü bilgisayarında yapılmıştır ve kullanıcılar deneye başlamadan önce deney hakkında bilgilendirilmişlerdir.

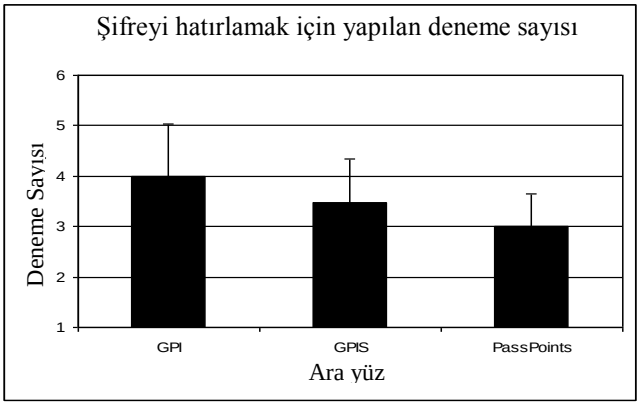
3. 3. 2 Deney Sonuçları

Her üç tasarı için de şifre onaylamak için geçirilen süre Şekil 3.8.'de gösterilmiştir ve bu süre tasarılar için verimliliği belirlemektedir. Onaylama süresi en uzun süren tasarı GPI iken en kısa onaylama süresine sahip olan tasarı ise Passpoints tasarısıdır. Tasarılar arasındaki süre farkı neredeyse önemli bir fark sayılabilir [$F(2, 65)=2.828$, $p<0.06$]. GPI ile Passpoints arasında ise önemli bir fark görülmektedir($p<0.05$).



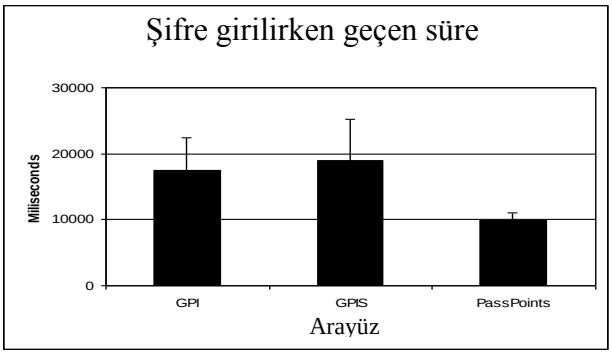
Şekil 3.8. GPI, GPIS ve PassPoints ara yüzleri için şifre onaylamada geçen süre

Etkilik, şifresini unutan katılımcıların sayısı, şifre hatırlamak için yapılan deneme sayısı ve doğru girilen şifreler için geçen süre ile belirlenir. GPI grubundan 4 katılımcı, GPIS grubundan 6 katılımcı ve Passpoints grubundan ise 5 katılımcı şifrelerini unutmuştur. Gruplar arasında önemli bir fark yoktur($X^2(2) = 0.5$, n. s.). Şifre hatırlamak için yapılan deneme sayıları Şekil 3.9. de gösterilmiştir ve gruplar arasında önemli bir fark gözlemlenmemektedir[$F(2, 65)=0.34$, n. s.].



Şekil 3.9. GPI, GPIS ve PassPoints ara yüzlerinde şifre hatırlamak için yapılan deneme sayıları

Doğru hatırlanan şifreler girilirken harcanan süre Şekil 3.10. da gösterilmiştir. Buna göre, Passpoints tasarısı en hızlı giriş süresine sahiptir fakat gruplar arasında önemli bir fark yoktur[F(2, 65)=0. 34, n. s.].



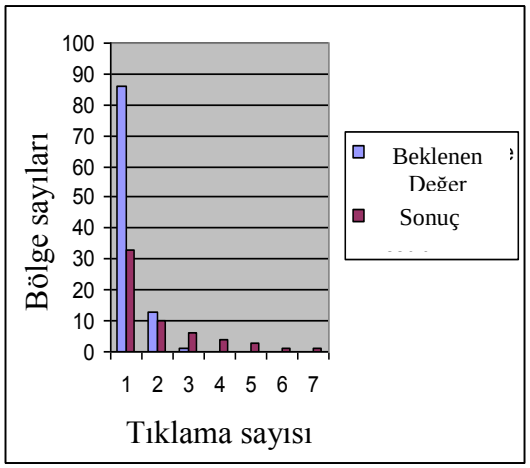
Şekil 3.10. GPI, GPIS and PassPoints ara yüzlerinde doğru hatırlanan şifreler girilirken geçen süre.

Sonuç olarak, Passpoints yöntemine doğru bir avantaj görünmekteyse de GPI ve GPIS in sonuçlarına bakıldığında bu tasarılarda kabul edilebilir limitler içindedirler. GPI ve Passpoints tasarılarını güvenlik açısından karşılaştırmak için sıcak-nokta durumları incelenmelidir. GPIS tasarısında ise, üretilen şifreler sistem tarafından atandığı için bu tasarı sıcak-nokta içermez. (23 kullanıcıdan sadece 4 tanesi "Şifre Üret" ("Generate Password") butonuna tıkladığı için, kullanıcıların defalarca şifre üreterek popüler ikonlara yönelmediğini kabul ediyoruz.)

Passpoints tasarısı için her kullanıcı 5 tıklama yaptığından dolayı toplam 115 tıklama yapılmıştır ve GPI için ise her kullanıcı 6 tıklama yaparak toplamda 138 tıklama yapılmıştır. Şekil 3.11. Passpoints tasarısı için tıklama sayısı ile tıklanabilecek bölge sayısını göstermektedir. Bu grafik için örneğin 32 bölge sadece 1 defa, 10 bölge sadece 2 defa, tıklanmıştır. Her bölge için ideal olarak beklenilen tıklama ise şu 3.6'daki formülle bulunabilmektedir:

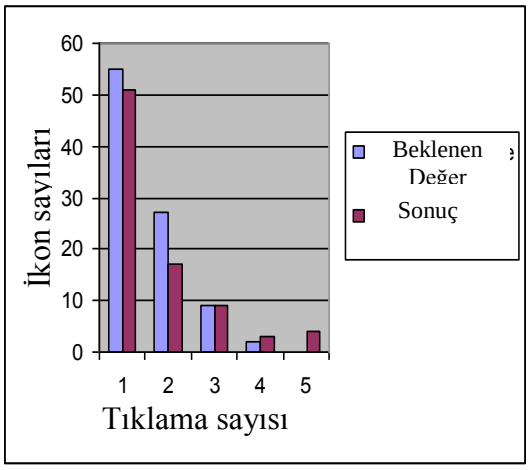
$$E_a = \binom{115}{r} \times \left(\frac{390}{391}\right)^{115-a} \times \left(\frac{1}{391}\right)^{a} \times 391 \qquad (3.6)$$

Burada E_a a defa seçilen alan sayısını göstermektedir. Toplamda 115 tıklama yapıldığı için bir alanın a defa tıklanma ihtimali $\left(\frac{1}{391}\right)^a \times \left(\frac{390}{391}\right)^{115-a}$ dır.



Şekil 3.11. PassPoints tasarısında her bölge için tıklama sayıları.

Aynı formül GPI tasarısı için de beklenen tıklama değerini vermektedir. Şekil 3.12. ikonlar için yapılan tıklama sayısını göstermektedir. Şekil 3.11. ve Şekil 3.12. yi karşılaştırırsak, GPI tasarısının Passpoints tasarısına göre sıcak-nokta açısından daha üstün olduğunu görebiliriz. Şekil 3.12. ye göre GPI ın sonuçları beklenen değere yakın çıkmıştır fakat hala sıcak-nokta probleminin bir ölçüde devam etmekte olduğu görülebilir. Diğer yandan, GPIS sıcak-nokta probleminden tamamen uzaktır.



Şekil 3.12. GPI ara yüzünde her bölge için tıklama sayısı

Bu çalışmada iki farklı grafik şifre tasarısı önerildi ve bu tasarılar ilk defa ipucuyla hatırlama tasarılarının ölçüsünde şifre alanı sunan tanımaya yönelik grafik şifre tasarılarıdır. GPI ve GPIS tasarılarının deney sonuçları, grafik şifrelerin sıcak-nokta probleminin giderilebileceğini doğrular niteliktedir. GPI ve GPIS için bazı kullanılabilirlik kaygıları vardır. Deneye katılan kişilerle yapılan anket sonucu, kullanılabilirlik problemlerinin sebebinin küçük boyuttaki ikon büyüklükleri olduğu düşünülebilir. İkonların bu boyutlarda seçilmesinin sebebi önceki tasarılarla karşılaştırabilmek için yaklaşık boyutlarda tutmak istememizdir. Bugünün teknolojisi daha geniş ekranlarda görüntü imkânı sunduğu için GPI ve GPIS tasarıları daha büyük ikonlarla denenerek daha iyi sonuçlar alınabilir.

Önerilen GPI tasarısı, tıklanacak bileşenlerin eşit dağılımı, etkili şifre alanını daraltacak olan bileşen bulundurmaması ve ikonların dikkatli seçiminden dolayı kullanıcıları mümkün oldukça belli ikonlara eğilimden uzaklaştıracak bir yapıdadır. Bu sebeple gerçekleştireceğimiz uygulamada bu tasarıyı kullanmayı uygun gördük. GPIS tasarısı da sıcak-nokta problemi bulundurmayan ve kullanışlılığı kabul edilebilir olan bir tasarı olsa da diğer bütün zorunlu atamalı sistemler gibi bu tasarı da gerçekleştireceğimiz uygulama için uygun değildir. Bu durumun sebepleri Bölüm 4'de ayrıntıları ile anlatılacaktır.

Geliştireceğimiz uygulamada kullanmak üzere belirlediğimiz ve uygun hale getirdiğimiz tasarıların uygulama geliştiriminde nasıl kullanıldığını ve uygulama detaylarını bölüm 4 de anlatıyoruz.

BÖLÜM 4 – GPEX (GRAPHICAL PASSWORD AS BROWSER EXTENSION)

İnsanların alışkanlıklarını değiştirmeleri ve yeni bir sisteme alışmaları zordur. Bu sebeple insanları yeni grafik şifre sistemlerine mecbur bırakmak yerine kullanımda olan metin tabanlı sistemlerle bütünleşik olarak çalışabilecek bir gerçekleştirimin daha uygun olacağını düşünmekteyiz. Bu sayede kullanıcılar yeni sistemi deneme amaçlı veya uzun süreli kullanım amaçlı olarak kolaylıkla kurup kullanabileceklerdir. İstedikleri takdirde çok fazla değişiklik gerektirmeden kurulumu geri alıp eski şifre girme yöntemlerine geri dönebileceklerdir. Günümüzün popüler internet tarayıcıları bize bu imkânı sağlamaktadır.

Mozilla Firefox internet tarayıcısı açık kaynak olmanın verdiği güven ve avantajlarla birlikte, sürekli kendisini yenileyen, günden güne artan kullanıcı kitlesine sahip olduğu için Gpex geliştirimine uygun bir tarayıcıdır. Tarayıcılara oldukça geniş bir alanda esneklik sağlayan eklentiler, tarayıcıları istenilen şekilde değiştirme, geliştirme ve kişiselleştirme imkânları sunmaktadır. Tarayıcılara yeni özellikler eklemek eklentiler sayesinde mümkündür ve bu eklentiler de açık kaynak olup tarayıcının resmi sitesinde yayınlanabileceği için, güvenlik konusunda da kullanıcılarına güven sağlamaktadır. Tarayıcı ve eklentilerin açık kaynak olması, kod hatalarının hızla bulunmasını, rapor edilmesini ve giderilmesini de sağlamaktadır. Gpex de bir Firefox eklentisidir ve grafik şifre kullanarak sitelere giriş yapabilme imkânı sunmakla tarayıcıya yeni bir özellik eklemektedir. Bu bölümde Gpex i inceleyeceğiz.

Gpex en başta bir eklentidir ve biz de Gpex i bir eklenti olma özelliğini dikkate alarak inceleyeceğiz. Gpex geliştiriminde kullanılan teknolojilerle birlikte, geliştirim detayları, üzerinde duracağız. Bu bölümün tarayıcı eklentisi geliştirecek kişilere de yardımcı olabileceğini düşünüyoruz. Gpex aynı zamanda bir şifreyi birçok web sitesinde kullanmaya imkân tanıyan ve her bir site için uygun kurallarda şifre üretebilen bir şifre yöneticisidir ve bir grafik şifre yönteminin eklenti olarak ilk gerçekleştirimidir. Bu bölümde Gpex i bir şifre yöneticisi olması özelliğini dikkate

alarak ve ayrıca ilk defa tarayıcı eklentisi olarak geliştirilmiş bir grafik şifre gerçekleştirimi olarak da ele alacağız.

4. 1 GPEX in geliştirim esasları

4. 1. 1 Eklentiler Hakkında

Firefox da kurulu eklentileri yönetebilmek için Eklenti Yönetimi vardır. Eklenti Yönetiminin görevi temel olarak, eklentileri güvenli olarak kurup kaldırmak, etkisiz hale getirmek, eklentilerin Firefox un kurulu sürümüyle uyumlu olduğuna emin olmak, eklenti yüklemenin güvenilir olduğu web sitelerinin listesini tutmak, güncellemeleri kontrol etmek ve çalıştırmak, kurulu eklentilerin ayarlarını değiştirebilmek için tercihler ara yüzüne ulaşımı sağlamaktır.

4. 1. 2 Eklentilerde Kullanılan Teknolojiler

Eklenti geliştiriminde temel olarak 4 farklı teknoloji kullanılmaktadır. Bu teknolojiler şu şekilde sıralanabilir;

- XML tabanlı bir ara yüz dili olan XUL (XML User Interface Language - "zul" olarak okunur).
- XHTML veya XML diliyle yazılmış olan içeriğe biçim vermek için kullanılan CSS (Cascading Style Sheet).
- Neredeyse bütün işlevselliği üstlenmiş olan nesne tabanlı betik dili JavaScript.
- Bir çerçeve (framework) olan ve eklentilere javascript ile yapılamayan birçok işlemi yapabilme imkânını sağlayan XPCOM.

CSS ve JavaScript teknolojileri internet sayfası tasarlama gibi birçok alanda kullanılan teknolojiler olduğu için bunlardan bahsetmeyeceğiz fakat XUL ve XPCOM hakkında kısaca bilgi vereceğiz.

4. 1. 2. 1 XUL

XUL, Mozilla ürünleri olan Firefox ve Thunderbird gibi uygulamalar üzerinde kullanıcı ara yüzleri tasarlamak için kullanılır. Firefox un kendisi de XUL un sunduğu menüler, düğmeler ve çerçevelerden oluşmaktadır.

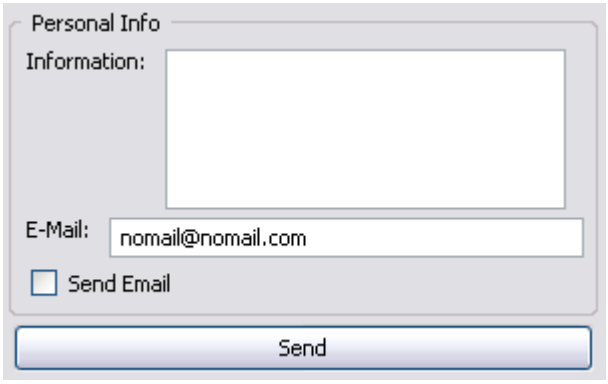
XUL dosyaları aslen birer XML dosyasıdır ve XML dosya yapısının tüm özelliklerini göstermektedir. Bu sebeple tüm XML dosyalarında olduğu gibi ilk satır olarak dosya türünü niteleyen <?xml version="1. 0"?> satırını içerir. XML dosyalarının temel bir özelliği açılış ve kapanış etiketleriyle çok iyi tanımlanmış olmalarıdır. Düzgün bir şekilde açılan ve kapatılan etiketler tuş, metin kutusu, paragraf gibi bir eleman belirtirler. Her eleman kendi içerisinde başka elemanlar bulundurabilir (bir tuşun üzerinde etiket bulundurması gibi) ve bu elemanlar en dıştaki elemanın çocuklarıdırlar. Doküman Nesne Modeli (DOM - Document Object Model), betik kodlarının doküman içeriğine, yapısına ve biçim özelliklerine erişimini sağlayan bir arabirimdir. XUL dokümanlarının içerisindeki tuş, etiket, metin kutusu gibi elemanlara bu model sayesinde JavaScript kullanılarak ulaşılabilir. Her bir eleman "height"(yükseklik), "width"(genişlik), "id"(kimlik) gibi birçok özelliğe sahip olabilir. XUL da dahil olmak üzere tüm XML türündeki dokümanlar genişleyebilir ve farklı türlerde elemanları içerebilecek bir yapıda olmalıdır. Örneğin, XHTML dokumanı "body", "head" gibi HTML isim uzayından(namespace) elemanlar gösterir, fakat bu dokuman içerisinde yine "body", "head" gibi etiketleri bulunan ve önceden tanımlanmış BIYOLOJI isim uzayından elemanlar da gösterilmek istenebilir. Bu durumda karışıklık olmaması için XHTML dokümanının içerisinde hangi "body" veya "head" etiketine referans verdiğimizi belirtmeliyiz. Hangi elemanın hangi isim uzayına ait olduğunu "xmlns" özelliğinde belirtiriz. (İsim uzayları içerisindeki her bir elemanın sadece o elemana özgü tanımlayıcı bir ismi vardır.) Eğer tasarlanan ara yüz tarayıcı içerisinde bir pencere olarak gösterilecek ise kök etiketi "<window>" olmalıdır ve pencere içerisinde XUL elemanları kullanılacağı için xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul" şeklinde belirtilmelidir. Belge içerisinde HTML elemanları da kullanılacaksa HTML isim

uzayı ayrı bir “xmlns” özelliği olarak “xmlns:html="http://www.w3.org/1999/xhtml"” şeklinde belirtilmelidir. Burada öncekinden farklı olarak “xmlns:” ifadesinden sonra “html” yazarak bir ön ek belirttik, bu ön ek tanımlanan isim uzayına ait elemanlarla birlikte kullanılacaktır. Artık belge içerisinde herhangi bir html elemanı kullanılacağı zaman “html” ön eki ile birlikte <html:div> </html:div> şeklinde kullanılabilir, XUL elemanları ise ön eksiz kullanılabilir.

Ara yüz tasarlamasında kullanılabilecek birçok XUL elemanı vardır ve bütün elemanlardan burada bahsetmek mümkün değildir. XUL içerisinde “window”(pencere), “page”(sayfa), “dialog”(dialog kutusu), “wizard”(sihirbaz) gibi üst seviye elemanlar, “label”(etiket), “button”(tuş), “list box”(liste kutusu), “text box”(metin kutusu) gibi nesneleri belirten elemanlar, “box”(kutu) , “grid”(ızgara), “stack”(yığın), “deck”(deste) gibi diğer elemanları bir grup olarak tutmaya yarayan elemanlar bulunabilir. Burada XUL elemanlarından ayrıntılı olarak bahsetmeyeceğiz fakat uygulamamızı geliştirme kısmında bazılarını kullanacağız. Şekil 4.1. de gösterilen eksiksiz basit bir XUL dosyasının oluşturduğu ara yüz Şekil 4.2. de gösterildiği gibidir.

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin/" type="text/css" ?>
<window id="theWindow" title="The Window" orient="horizontal"
        width = "300" height = "150"
     xmlns="http://www.mozilla.org/keymaster
     /gatekeeper/there.is.only.xul"
     xmlns:html="http://www.w3.org/1999/xhtml">
 <vbox width="300">
<vbox>
<hbox>
<groupbox  flex="1">
  <caption>
   <label value="Personal Info"/>
  </caption>
 <hbox>
    <label control="ctrlInfo" value="Information:"/>
    <textbox id="txtInfo" width="200" height="80"/>
 </hbox>
 <hbox>
 <label control="ctrlEmail" value="E-Mail:"/>
 <textbox id="txtEmail" flex="1" value="nomail@nomail.com"/>
  </hbox>
  <checkbox label="Send Email"/>
 </groupbox>
</hbox>
     <button label="Send"/>
 </vbox>
</vbox>
</window>
```

Şekil 4.1. Basit Bir XUL Dosyası



Şekil 4.2. XUL ara yüzü

4. 1. 2. 2 XPCOM(Cross Platform Component Object Model)

Java, Python ve C++ gibi programlama dilleri kullanılarak dosya okuma yazma işlemleri, bellek yönetimi, veri yapılarının kullanımı gibi işlemler yapılabilir fakat JavaScript dili kullanılarak bu işlemler çok sınırlı bir düzeyde yapılabilirler veya hiç yapılamazlar. XPCOM bileşenleri yukarıdaki programlama dilleri ile yazılmış kodları tarayıcının hizmetine sunar ve bu sayede dosya okuma yazma gibi işlemler yapılabilir. XPCOM sayesinde uygulamalar küçük parçalar (bileşenler) halinde geliştirilebilirler ve birbiriyle alakalı bileşenler bir kütüphanede toplanarak modüleri oluşturur. Modüler yapı kodların tekrar kullanımında, kod güncellemede, performans artırımında ve kodların yönetiminde kolaylıklar sağlar. Geliştirilen bileşenler değişkenlere metotlara sahip sınıflardır. Bir bileşen bir defa yazıldıktan sonra bu bileşen sınıftan birçok örnek üretilerek farklı uygulamalarda kullanılabilirler(kod tekrar kullanımı). Bileşenler güncellemede de kolaylıklar sağlar çünkü güncelleme gerektiğinde sadece bileşen sınıfının güncellenmesi yeterlidir. Tüm kodun bir defada yüklenmesi yerine bileşenler sayesinde kodun gerekli kısmının gerektiğinde yüklenmesi de performans açısından faydalar sağlar. Geliştiricilerin milyonlarca satır kod arasından ilgilendikleri kısmı kolaylıkla bulmaları da yine bu modüler yapı sayesindedir.

XPCOM bileşenleri Mozilla üzerinde birçok servise ve nesneye ulaşmayı sağlar ve bu bileşenlere JavaScript ile kolaylıkla erişilebilir. Örneğin, XPCOM bileşenleri kullanılarak dosya sistemine ulaşılabilir. JavaScript ile yazılan şu kod, bize bir XPCOM bileşeni döndürür;

```
var dosya = Components.classes["@mozilla.org/file/local;1"].
createInstance(Components.interfaces.nsILocalFile);
```

Burada “file” isimli modül içerisinde bulunan “local” isimli bileşen sınıfından bir nesne oluşturulması istenmektedir ve oluşturulan nesne dosya isimli değişkene atanmaktadır. XPCOM nesnesi “dosya” değişkenine atanmıştır ve artık bu nesnenin sağladığı dosya işlemleri ile ilgili bütün metotlara ve alanlara erişebiliriz.

Örnekte de görülebileceği gibi her bir bileşenin "@alan_adı/modül_ismi/bileşen_ismi; versiyon_numarası" şeklinde bir kontrat kimliği vardır ve bu kimlik sayesinde hangi sınıftan bir nesne oluşturulacağı belirlenir. Bileşenler bir veya daha fazla ara yüzü gerçekleştirirler ve ara yüzler genellikle "nsILocalFile" da olduğu gibi "nsI---" şeklindedir. Nesne oluştururken bu nesnenin gerçekleştirdiği hangi ara yüze ait metotları kullanacağımızı belirtmek için createInstance() metoduna ara yüz ismini (örnekte Components. interfaces. nsILocalFile) parametre olarak veririz.

Bazı XPCOM bileşenlerinden birçok nesne üretilip kullanılabilirken(örneğin birçok dosya oluşturulabilir) bazı bileşenler ise bir servistir ve bu bileşenlerden sadece bir nesne oluşturulabilir, her ihtiyaç olduğunda bu tek nesne kullanılır. Örneğin tarayıcı ile ilgili birçok ayar ve kullanıcı tercihleri tarayıcıda saklanır. Firefox un adres satırına about:config yazılarak bu tercihler görüntülenebilir. Bu ayarlara JavaScript ile ulaşmak için aşağıdaki kod kullanılır;

```
var tercihler = Components.classes["@mozilla.org/preferences-
service;1"].getService(Components.interfaces.nsIPrefService);
```

Burada "nsIPrefService" ara yüzünü gerçekleştirmiş olan "preferences-service" sınıfından nesne oluşturuyoruz (eğer bu nesne daha önceden oluşturulmuş ise yeni nesne oluşturulmaz, var olan nesneye ulaşılır). Burada elde edilen "tercihler" değişkeni kullanılarak "nsIPrefService" ara yüzünün sağladığı tercih ekleme, silme, değiştirme gibi metotlara erişilebilir.

Servis olan bileşenlere . getService(), diğer bileşenlere ise . createInstance() metoduyla ulaşılır, bu sebeple ulaşılacak olan bileşene hangi metotla ulaşılacağını belirlemek önemlidir.

4. 1. 3 Temel Klasör - Dosya Yapısı ve Geliştirme Ortamı

Bir XUL uygulamasında kullanıcı arayüzünü oluşturan elemanların tümü “Chrome” dur. Firefox tarayıcısının tüm ara yüzü bir “chrome” dur. Chrome u üç farklı paket oluşturur. Bunlar temel XUL dosyalarını ve javascript dosyalarını içeren “content”, farklı dil destekleri sunmak için tanım dosyalarını içeren “locale” ve görünümü yönetmek için kullanılan görsel elementleri içeren “skin” klasörleridir.

Eklentiler belirli bir klasör ve dosya yapısı içerisinde oluşturulduktan sonra basit bir zip dosyası haline dönüştürülür fakat uzantısı eklenti olarak tanınmak üzere değiştirilir. Bir eklenti dosyasının uzantısı XPI dır ve “zippy” olarak telaffuz edilir. Basit bir xpi dosyasının yapısı Şekil 4.3. de gösterilmiştir;

Şekil 4.3. Basit Bir XPI Dosyasının Yapısı

4. 1. 3. 1 Tanımlama Dosyası

Klasör yapısı içinde en üst seviyede bulunan install. rdf (Şekil 4.4.) dosyası eklentimiz hakkında bazı tanımlama bilgileri içermektedir.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:RDF="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:em="http://www.mozilla.org/2004/em-rdf#">
    <RDF:Description about="urn:mozilla:install-manifest">
        <em:aboutURL>chrome://gpex/content/about.xul</em:aboutURL>
        <em:creator>Mustafa Yuceel</em:creator>
        <em:description/>
        <em:homepageURL>http://myuceel.etu.edu.tr/gpex/</em:homepageURL>
        <em:iconURL/>
        <em:id>myuceel@etu.edu.tr</em:id>
        <em:name>GPEX</em:name>
        <em:version>1.0A</em:version>
        <em:updateURL/>
        <em:type>2</em:type>
        <em:targetApplication>
            <RDF:Description>
                <em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>
                <em:maxVersion>4.5.0</em:maxVersion>
                <em:minVersion>3.0</em:minVersion>
            </RDF:Description>
        </em:targetApplication>
    <em:optionsURL>chrome://gpex/content/options.xul</em:optionsURL>
    <em:updateKey/><em:hidden/>
    </RDF:Description>
</RDF>
```

Şekil 4.4. Install.rdf Dosyasının Görünümü

Install. rdf dosyası Şekil 4.4. de görülebileceği gibi basit bir XML dosyasıdır ve geliştirilecek olan eklenti hakkında gerekli bilgileri tanımlar. Bu bilgilerden önemli olan bazıları şöyledir;

- "chrome://gpex/content/about. xul" eklenti yöneticisinden eklenti hakkında bilgi alınmak istendiğinde, burada yolu belirtilen XUL dosyası gösterilir.
- myuceel@etu. edu. tr, bu metin eklentinin kimliğidir. Bir tarayıcıya kurulmuş olan her bir eklentinin tek ve eşsiz bir kimliği olmalıdır, bu sebeple e-posta adresi gibi tek olduğu garanti edilebilen kimlik isimleri kullanılması uygundur.
- Kurulacak olanın tipi hakkında bilgi vermek için <em:type>2</em:type> etiketleri kullanılır ve buradaki 2 eklenti kurulacağını simgeler, eklenti yerine tema kurulacak olsaydı burada belirtmeliydik.

- Eklentinin birlikte çalışacağı Firefox sürümü maxVersion ve minVersion etiketleriyle belirtilir.
- optionsURL etiketi eklenti yöneticisinden eklenti ile ilgili ayarlara ulaşılmak istendiğinde gösterilecek olan XUL dosyasının yolunu belirtir.

4. 1. 4 Firefoxu Genişletmek

Yazdığımız XUL dosyaları metin kutuları, etiketler, düğmeler gibi ara yüz elemanları içermektedirler ve bu elemanlara ait işlevsellikler de JavaScript vasıtasıyla yerine getirilmektedir. Bu dosyaların tarayıcı ile birlikte çalışabilmesi için tarayıcıya eklenmelidirler. Firefox un kendisi de XUL ve JavaScript kullanılarak yazılmıştır, bu sebeple bizim yazdığımız ara yüzle bütünleşebilir bir yapıya sahiptir. Eklentimizi Firefox ile bütünleştirmek için Firefox un sahip olduğu bir bileşeni değiştirebilir veya Firefox a yeni bir bileşen ekleyebiliriz.

Tarayıcının tanımlandığı XUL dosyası /chrome/browser. jar içerisinde bulunan content/browser/browser. xul dosyasıdır. Tarayıcı üzerine eklemek istediğimiz ara yüzü browser. xul dosyası ile birleştirmeliyiz, bu sayede tarayıcının sahip olduğu arayüz görüntülenirken, bizim arayüzümüz de görüntülenebilir. Burada bahsedilen birleştirme işlemi, aslında yazdığımız XUL dosyasını browser. xul üzerine bindirme işlemidir. Bindireceğimiz XUL dosyası Şekil 4.5. de gösterilmektedir.

```
<?xml version="1.0"?>
<overlay id="gpex"
    xmlns="http://www.mozilla.org/keymaster
    /gatekeeper/there.is.only.xul">

    <script src="chrome://gpex/content/pwd.js" />
    <script src="chrome://gpex/content/listenerEkle.js" />

    <statusbar id="status-bar">
    <statusbarpanel id="my-panel" label="P" />
    </statusbar>
</overlay>
```

Şekil 4.5. browser.xul dosyası üzerine bindirilen xul dosyası

Şekil 4.5. de gördüğünüz gibi dosya içeriği yeni bir pencere olmadığı için <window> veya <dialog> gibi bir etiketle değil, bu tanımlamanın bir bindirme tanımlaması olduğunu belirtmek üzere <overlay> etiketi kullanılmıştır. Browser. xul dosyasına ekleyeceğimiz bu kodlar, aynı browser. xul içerisindeki kodlar gibi tarayıcı açılır açılmaz çalıştırılacaktır. Bu sebeple <script> etiketlerinde yolunu belirttiğimiz JavaScript dosyalarındaki javascript kodları tarayıcının açılmasıyla birlikte yorumlanacaktır. Kodların alt satırlarında görünen <statusbar> etiketine verilen "status-bar" kimliği zaten browser. xul içerisinde tanımlandığı için, burada yaptığımız tanımlama tarayıcının tanımlamasının üzerine yazılacaktır. Bu durumda tarayıcının durum çubuğunu değiştirerek "my-panel" kimliğinde bir <statusbarpanel> eklemiş oluyoruz.

Bu kodlarda açıklamadığımız fakat dikkat çekebilen bir ayrıntı da <script> etiketi içerisinde yazdığımız URI dır. Bu bir "chrom" URI dır ve chrome://gpex/content/pwd. js şekilnde yazılmıştır. Bu yazılışı inceleyecek olursak; "chrome://" ifadesi "http://" gibi bir ifadedir ve tarayıcıya bu ifadenin bir chrome URI olduğunu bildirir. Tarayıcı bu sayede gelecek olan dosyaya nasıl davranacağını bilir ve bu dosyaya bir "chrome" olarak davranacaktır. Burada görünen "gpex" ise paket ismidir ve ara yüz bileşenlerinin bulunduğu paketi ifade eder. Bu ismin diğer eklentilerle karışmaması için özenle seçilmiş olması gereklidir. Bu paketin nasıl

tanımlandığı bu bölümde bahsedilecektir. “content” olarak belirtilen kısım istenilecek olan verinin türünü bildirir. Burada istenilebilecek 3 farklı tür vardır; birinci tür “content” dir. Uygulamanın yapısını ve fonksiyonlarını belirleyen xul, javascript gibi dosyalar burada bulunur. İkinci tür düğme, etiket isimleri gibi ara yüz elemanlarının dilini değiştirmek için kullanılan dil tanımlama dosyalarının bulunduğu klasör olan “locale” dir. Diğer bir tür ise CSS ve resim dosyaları gibi tema ile ilgili dosyalardır ki “skin” ismiyle belirtilmelidir. “pwd. js” kısmında istenilen dosyanın ismi belirlenir. Biz burada bir JavaScript dosyasına ulaşmak istiyoruz.

4. 1. 5 Krom Bildirisi

Krom paket ismi ile krom klasör ve dosyalarının fiziksel adreslerinin ilişkilendirilerek tanımlanmaları gereklidir. Örneğin yukarıda bahsettiğimiz “gpex” bir paket ismidir ve bir fiziksel yola karşılık gelir. Dosyalar içerisinde bu paket isimleri kullanılırlar. Krom kayıt işlemi içeriği Şekil 4.6. da görülen chrome. manifest dosyaları içerisinde yapılır. Her bir satırda bir kayıt işlemi yapılmaktadır. Şekil 4.6. daki gösterimde buraya rahatlıkla sığması için “overlay” satırı iki satır olarak yazılmıştır fakat bu yazım chrome. manifest dosyası için hatalıdır. İlk satırda “gpex” paket ismiyle “content” klasörü tanımlanmıştır. İkinci satırda, browser. xul dosyası üzerine graphpwd. xul dosyası bindirilmiştir. Diğer satırlarda ise görünüm ve dil seçenekleri için “gpex” paketi kayıt edilmiştir.

```
content gpex chrome/gpex/content/
overlay  chrome://browser/content/browser.xul
         chrome://gpex/content/graphpwd.xul
skin gpex global chrome/gpex/skin/global/
locale gpex en-US chrome/gpex/locale/en-US/
```

Şekil 4.6 chrome. manifest dosyasının görünümü

4. 1. 6 Eklentinin Çalışması

Buraya kadar gerekli temel işlemler yapıldı ve Şekil 4.5. de görülen kodlarımız tarayıcı açıldığında çalıştırılmak üzere hazırlandı. Bu dosya içerisinde henüz hangi fonksiyonları içerdikleri açıklanmayan iki adet javascript dosyası bulunuyor. Bunlardan bir tanesi listenerEkle. js dosyasıdır. Bu dosya tarayıcıya bir olay dinleyici ekliyor ve bu dinleyici sayfa yüklenmelerini dinliyor. Tarayıcıya yeni bir sayfa yüklendiğinde önceden tanımlanmış bir fonksiyon çalışıyor. Çalışan bu fonksiyon, yüklenen sayfa içerisindeki (ve iç içe bulunabilecek bütün çerçevelerdeki) bütün şifre alanlarını dolaşarak bulduğu şifre alanlarının her birine birer olay dinleyici ekliyor. Eklenen bu olay dinleyiciler şifre alanına çift tıklandığında "userEventListener" fonksiyonunu çalıştırıyorlar. Bu fonksiyon "chrome://gpex/content/pwd. xul" dosyasının yeni bir pencere olarak açılmasını sağlıyor.

Buraya kadar anlatılan kısımda sayfadaki bir şifre alanına tıklandığında eklenti penceremizin açılmasını sağlamış olduk. "graphpwd. xul" dosyasında bulunan diğer bir javascript dosyası da "pwd. js" dir. Bu dosya içerisinde daha sonra kullanılacak olan fonksiyonlar tanımlanmıştır. İlerideki kısımlarda yeri geldikçe burada tanımlanan fonksiyonlardan bahsedilecektir.

Şifre alanına çift tıklanarak açılan "pwd. xul"(Şekil 4.7.) penceresi kullanıcıların şifrelerini tıklamaları için gerekli olan ara yüzü (Şekil 4.8.) sağlar.

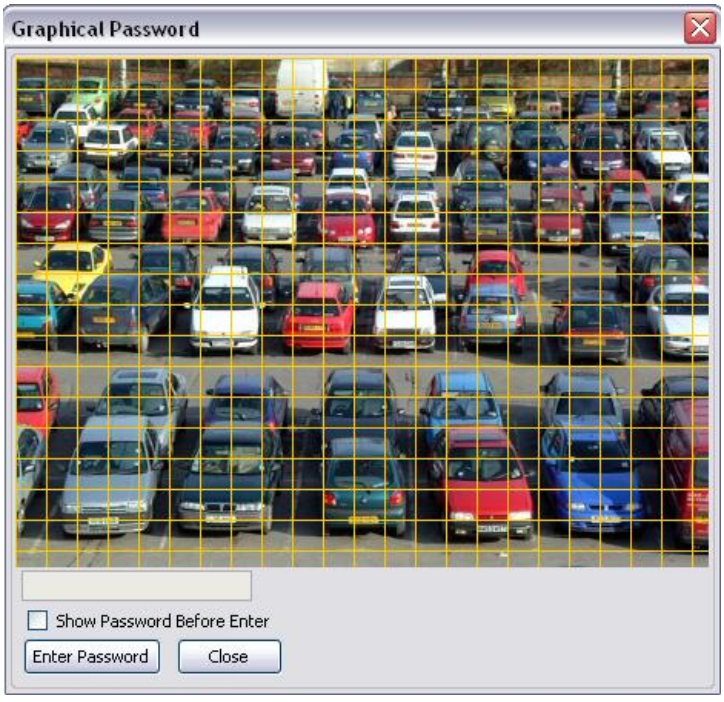
```
  <groupbox align="start">

      <vbox align="left">
          <image id="resim" style="width: 451px; height: 331px" src="resim.jpg" onload="getPicture()"
          onmousedown="updateMouseCoordinates(event);"/>
      </vbox>
      <vbox align="left"  >
          <textbox id="kalan" size="25" readonly="true" value=""/>
      </vbox>

      <checkbox id="showpwd" label="Show Password Before Enter" checked="false"/>
      <hbox align="left"  >
          <!-- <button label="Renew" oncommand="yenile()"/> -->
          <button label="Enter Password" oncommand="sifreyiGir()"/>
          <button label="Close" oncommand="pencereyiKapat()"/>
      </hbox>
      <box id="aBox" width="400"> <!-- buraya applet.js den frame eklenecek.
      document.getElementById("aBox") -->
      </box>
      <box id="imageAdjustBox" width="1" height="1"></box>
  </groupbox>
<script src="chrome://gpex/content/ArrangeInterface.js" />

'window>
```

Şekil 4.7. Eklenti penceresini oluşturan pwd.xul dosyasının görünümü



Şekil 4.8. Eklenti penceresinin ara yüz görünümü

“pwd.xul” penceresinde “resim” kimliği ile tanımlanan eleman yüklendiği zaman “getPicture()” fonksiyonu, resim üzerine tıklandığında ise “updateMouseCoordinates()” fonksiyonu çalışacak şekilde ayarlanıyor. Ara yüzde iki tane düğme görünüyor ve bu düğmelerden birisi pencereyi kapatmak için kullanılıyor. Diğer düğme ise “sifreyiGir()” fonksiyonunu çalıştırıyor. Şekil 4.7. de gösterilmeyen bazı javascript dosyaları da bu dosya içerisinde <script> etiketi ile eklenmiştir.

“getPicture()” fonksiyonu “nsIPrefService” den bir nesne örneği kullanmaktadır. “Preferences service” tarayıcının tercihler servisine erişimi sağlar bu servis kullanılarak kullanıcının kayıtlı tercihlerine erişilebilir veya değiştirilebilir. Tercihlerin ayarlanması ve değiştirilmesi daha sonra anlatılacaktır. Bu metot tercihlerde kayıtlı olan resmi bilgisayardan okuyarak “resim” kimliğiyle tanımlanan elemana yükler.

“updateMouseCoordinates()” fonksiyonu kullanıcı resim üzerine tıkladıkça çalışır. Bu fonksiyon her tıklamada farenin X ve Y koordinatlarını kaydeder ve ekranda tıklama sayısını gösterir.

“sifreyiGir()” fonksiyonu kullanıcı “Enter Password” düğmesine tıkladığında çalışır ve kaydedilen koordinatları kullanarak üretilen şifreyi ilgili metin kutusuna yazar. Bu fonksiyon içerisinde başka fonksiyonları da kullanarak her farklı site için site kurallarına uygun olarak şifre üretir. Bu fonksiyon kullanıcı tercihlerine ulaşarak kullanıcının hangi şifre yöntemini kullandığını tespit eder. (Hatırlamaya dayalı veya Tanımaya dayalı iki farklı sistem mevcuttur.) Şimdiki durumda hatırlamaya dayalı grafik şifre tasarısı seçilidir ve anlatım bu tasarı için devam edecektir. Tıklanılan koordinatların resim üzerindeki karelerden hangisine düştüğü bulunur ve bununla beraber şifre girilecek olan web sitesinin alan adı ve alt alan adı tespit edilir [42]. Tıklanılan karelerden oluşan bir dizi ve sitenin alan adı md5 özet algoritması ile şifre üretiminde kullanılır [43]. Üretilen şifre henüz site kurallarına uydurulmamıştır, bu sebeple elimizdeki şifre “getPassword()” fonksiyonuna parametre olarak gönderilerek site kurallarına uygun bir şifre elde edilir. “getPassword()” fonksiyonu şifreyi, alan adını ve alt alan adını parametre olarak alır, sonuç olarak ilgili site kurallarına uygun bir şifre verir.

"getPassword()" fonksiyonu "matchSiteRules()" fonksiyonuna tüm parametreleri gönderir. "matchSiteRules()" fonksiyonu alan adı ve alt alan adına göre bir "Rules" nesnesi oluşturur. Bu nesne "rules. xml" (Şekil 4.9.) isimli kurallar dosyasındaki ilgili alan adı ve alt alan adına göre oluşturulur. Oluşturulan kurallar nesnesi ile şifre "applyRules()" fonksiyonuna gönderilir ve ilgili kurallar şifre üzerinde uygulanır. Üretilen son şifre "matchSiteRules()" metodundan "getPassword()" metoduna oradan da "sifreyiGir()" metoduna aktarılarak ilgili şifre alanına girilir ve pencere kapatılır.

4. 1. 7 Şifre Kuralları

Domain/Subdomain	Minimum Password Length	Maximum Password Length	Symbols	Numeric	Start With Letter	Forbidden Characters
» etu.edu.tr	12	16	Not Allowed	Allowed	Allowed	
» mail.etu.edu.tr	12	16	Not Allowed	Allowed	Allowed	
» google.com	12	16	Allowed	Must	Allowed	
» abc.google.com	12	16	Allowed	Allowed	Allowed	
» kariyer.net	12	16	Not Allowed	Allowed	Allowed	
» mynet.com	12	16	Not Allowed	Allowed	Allowed	
Not in this list	12	16	Allowed	Allowed	Allowed	

```
<domain domainName="etu.edu.tr">
    <subDomain subDomainName="noSubDomain">
        <minLength obligationLevel="must" value="12"></minLength>
        <maxLength obligationLevel="must" value="16"></maxLength>
        <symbols obligationLevel="optional"></symbols>
        <numeric obligationLevel="optional"></numeric>
        <initWithLetter obligationLevel="optional"> </initWithLetter>
        <excludeChars value=""></excludeChars>
    </subDomain>
    <subDomain subDomainName="mail.etu.edu.tr">
        <minLength obligationLevel="must" value="12"></minLength>
        <maxLength obligationLevel="must" value="16"></maxLength>
        <symbols obligationLevel="optional"></symbols>
        <numeric obligationLevel="optional"></numeric>
        <initWithLetter obligationLevel="optional"> </initWithLetter>
        <excludeChars value=""></excludeChars>
    </subDomain>
</domain>
```

Şekil 4.9. Şifre kurallarının belirlendiği rules. xml dosyası

“rules. xml” dosyası eklenti her açıldığında güncelleme için kontrol edilir ve http://myuceel. etu. edu. tr adresinden indirilerek güncellenir. Bu dosya farklı siteler için farklı kurallar tanımlama imkânı sunar. Yeni siteler için kurallar eklemek veya dosyada var olan sitelerin kurallarını güncellemek için bu dosyanın sunucudaki kopyasını güncellemek yeterlidir.
Dosya içerisindeki etiketlerin tanımları şu şekildedir;

Etiket: Domain : Şifre üretilecek olan siteye ait tanımlamalar bu etiketin altında yapılır.
Özellik: domainName: Sitenin en genel alan adını tanımlar. Alt alan adı içermez.
Etiket: subDomain : Bir siteye ait farklı alt alanları burada tanımlanır. Bu sayede her farklı alt alanı için farklı şifre kuralları oluşturulabilir.
Özellik: subDomainName: Alt alan adını tanımlar.
Etiket: minLength : Minimum şifre uzunluğunu tanımlar.
Etiket: maxLength : Maksimum şifre uzunluğunu tanımlar.
Etiket: symbols : Üretilen şifrenin sembol içerip içermeyeceğini tanımlar
Etiket: numeric : Üretilen şifrenin numara içerip içermeyeceğini tanımlar
Etiket: initWithLetter: Üretilen şifrenin harfle başlaması gerekip gerekmediğini belirtir.
Etiket: excludeChars: Üretilen şifrede bulunmaması gereken karakterler varsa burada tanımlanır.

Özellik: obligationLevel: İlgili özelliğin üretilecek şifrede bulunma durumunu belirtir. 3 farklı değer alabilir.
Değer: Optional – Özellik şifrede bulunmayabilir, site bu özellik hakkında bir şart koymamıştır.
Değer: Must – Özellik üretilecek şifrede mutlaka bulunmalıdır.
Değer: Forbidden – Özellik üretilecek şifrede bulunamaz.

4. 1. 8 Seçenekler ve ara yüz ayarlamaları

“ArrangeInterface. js” dosyası uygulama penceresi olan “pwd. xul” içerisinde tanımlanmıştır ve uygulama penceresi açıldığında gerekli düzenlemeleri yapmaktan sorumludur. XPCOM nesneleri ile tercihler servisine ulaşarak uygulamanın ilk defa mı çalıştırıldığını kontrol eder. Eğer uygulama ilk defa çalıştırılıyorsa, bir XPCOM servisi olan klasör servisinden programlar için geçerli klasör yolunu alır ve bir “nsiLocalFile” bileşeni kullanarak 150 adet ikon dosyasını ilgili klasöre kopyalar. Daha sonra tercihler servisine ulaşarak ilk defa çalıştırmanın gerçekleştiğini bildirir. Bundan sonraki çalıştırmalarda bu dosyalar yeniden kopyalanmayacaktır. Burada bahsedilen resim dosyaları tanımaya dayalı grafik şifre yöntemi olan GPI ın kullanacağı dosyalardır. Bu javascript dosyası yine tercihler servisine ulaşarak ara yüz tipini eğer hiç belirlenmediyse varsayılan ara yüz tipine çevirir ve “pwd. xul” penceresini tercihlerde seçilmiş olan tasarıyı göstermek üzere (düğmeler, pencere eni-boyu gibi özellikleri) ayarlar.

“Options. xul” dosyası Firefox un Araçlar > Eklentiler > Seçenekler menüsüne tıklanılınca gösterilen pencereyi tanımlar. Bu pencerenin görüntüsü Şekil 4.10. de gösterilmiştir.

Şekil 4.10. Tercihler penceresi ara yüz görünümü

Grafik şifre ara yüz tercihleri pencerede görünen iki farklı tasarıdan birisi seçilerek belirlenir. Soldaki tasarı daha önce deneylerini yaptığımız hatırlamaya dayalı grafik şifre tasarısıdır, sağdaki tasarı ise deneyleri yapılmış olan tanımaya dayalı grafik şifre tasarısıdır. Penceredeki düğmelerden "Change Picture" bilgisayarda kayıtlı bir resmin tasarı içerisinde kullanılması için kullanılıyor. Kullanılacak olan resmin boyut ve üzerindeki ızgaralar gibi özelliklerini ayarlamak için Bölüm 4'de bahsettiğimiz ve eklentiye erişim bulunmadığı zamanlarda grafik şifre ile giriş imkânı sağlayan internet sitesinde sunduğumuz araç kullanılabilir. "Default Preferences" butonu varsayılan ayarlara dönerek, grafik şifre tasarısını hatırlamaya dayalı olan yönteme ve resmi de otopark resmine çevirir.

4. 2 Bir Şifre Yöneticisi olarak GPEX

Şifre yöneticileri hatırlaması zor fakat güvenli şifreler seçmek için kullanılan uygun bir yöntemdir. Bu yöneticiler masaüstü uygulaması, internet uygulaması veya

tarayıcı eklentisi gibi birçok şekilde gerçekleştirilebilir. Tarayıcı eklentisi olarak gerçekleştirilen şifre yöneticilerden bazıları hatırlaması kolay ve karmaşık olmayan bir şifreyi tahmin edilmesi zor ve ataklara karşı güçlü şifrelere dönüştürürler[16, 17, 35]. Şifre yöneticilerinin çözüm olabileceği bir diğer problem de şifrelerin yeniden kullanımı konusudur. İnsanlar çok fazla sayıda internet uygulaması arasında hatırlama kolaylığı açısından bir şifreyi birkaç sitede kullanma eğilimindedirler. Şifre yöneticileri her bir site için farklı bir şifre üreterek bu problemi ve bununla beraber yemleme ataklarını da engellemektedirler[16, 17, 35]. Yemleme kanuni bir internet sitesinin görüntüsüne sahip internet siteleri yaparak insanları şifrelerini bu sitelere girmeleri için ikna etmeye dayanır. Bu sitelerin alan adları kanuni olan siteyle aynı değildir ve kullanıcılar her giriş sırasında adres satırını kontrol etmelidirler. Tarayıcı eklentisi olarak gerçekleştirilmiş şifre yöneticileri alan adına göre şifre ürettikleri için, gerçek sitenin şifresini gerçek olmayan bir internet sitesine yazmaları söz konusu değildir. Bu açıdan da bir güvenlik problemine çözüm sunmaktadırlar.

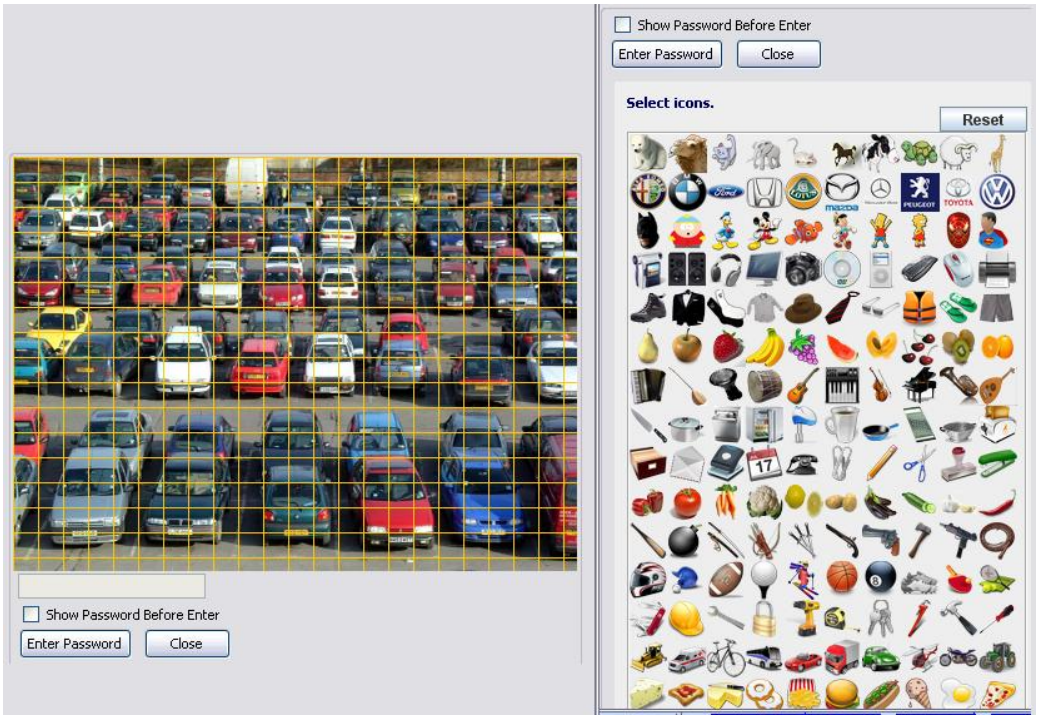
Şifre yöneticilerinin sağladıkları bu avantajların yanı sıra yapılan kullanışlılık çalışmaları şifre yöneticilerinin bazı problemleri olduğunu göstermektedir [35]. Araştırmaların sonucuna göre önemli bir bulgu, kullanıcıların sistemi tam kavrayamamaları ve eksik bir zihinsel modele sahip olmalarıdır ki bu durum önemli bir güvenlik açığı oluşturur[35].

Biz de yaptığımız çalışmada grafik şifrelerin ve şifre yöneticilerin güçlü yönlerini birleştirerek daha kullanışlı ve güvenli bir kimlik kanıtlama sistemi öneriyoruz. Önerdiğimiz bu sistemde ara yüz olarak sunulacak olan tıklamaya dayalı grafik şifre tasarıları kullanıyoruz. Önerdiğimiz bu sistemde kullanıcılar bir artalan resminin üzerine birkaç defa tıklayarak şifrelerini oluşturacaklar ve tarayıcı eklentisi bu noktaları siteye özel karmaşık ve site kurallarına uygun bir şifre haline getirecek. Grafik şifreyi metin tabanlı şifreye dönüştürerek sayfadaki forma bu şifrenin girilmesi sağlamak çok da karışık olmayan bir işlemdir ve bu sebeple bu sistemin anlaşılması kolay bir zihinsel modele sahip olduğunu düşünüyoruz. Eklentimiz kullanıcı dostu ve güvenli bir kullanım sağlayacak şekilde tasarlanmıştır. Örneğin, sistemimizde eklentinin bulunduğu durumlarla ilgili karmaşıklık yoktur, yani

eklentinin aktif edilme ve kullanıcıdan girdi bekleme gibi durumları anlaşılır ve çok açıktır. Bu sistem sunucu tarafında hiçbir değişikliğe ihtiyaç bırakmadığı için de kolay bir adaptasyon sürecine sahiptir.

Sunduğumuz grafik şifre yöntemi olan GPEX güvenli ve kullanışlı bir etkileşim sağlayan yeni bir metottur. Mozilla Firefox tarayıcısında çalışan bir kimlik kanıtlama sistemidir ve uygulama http://myuceel.etu.edu.tr/gpex sitesinden ulaşılabilir durumdadır. Üzerinde ızgara çizgileri olan 451 x 331 piksel boyutlarındaki bir resim üzerinde fare tıklamasıyla seçim yapılarak şifrenin oluşturulduğu bir grafik şifre tasarısına ve ayrıca diğer bir seçenek olarak 150 adet ikon içerisinden seçim yapılarak şifrenin oluşturulduğu bir grafik şifre tasarısına sahiptir. (Şekil 4.11.)

GPEX in aktive edilebilmesi için kullanıcı sayfa üzerindeki bir şifre alanına çift tıklamalıdır. Eğer tıklanılan alan gerçekten bir şifre alanıysa eklenti bir pencere olarak açılır ve tıklanılan alanın rengi sarıya dönüşerek bir girdi beklendiğini gösterir. Pencere içerisinde 391 farklı kare vardır ve aynı kare içerisine yapılan tıklamalar aynı noktayı tıklamak anlamındadır. Çizgilerin tam üzerine yapılan tıklamalar ise çizginin solundaki ve üstündeki karenin içine tıklamakla aynı anlamdadır. Kullanıcı bu noktalara tıklayarak grafik şifresini oluşturmalıdır. Diğer bir seçenek olarak da 150 ikon arasından seçilen ikonların bir şifre oluşturması vardır. Meyveler, mutfak eşyaları, araba markaları gibi15 farklı kategoriden 10ar tane nesnenin ikonlar şeklinde gösterildiği bu pencerede kullanıcılar istedikleri kadar ikonu seçerek şifrelerini oluştururlar.



Şekil 4.11. GPEX eklentisinde kullanılan grafik şifre tasarıları

4. 2. 1 PwdHash

PwdHash[16, 17] de bir tarayıcı eklentisidir ve kullanıcıların girdiği metin tabanlı şifrelerden, sitelere özel güçlü şifreler üretmektedir. Kurulumdan sonra PwdHash i başlatmak için kullanıcı şifre alanına şifresini yazmaya @@ karakterleriyle başlamalı veya yazmaya başlamadan önce klavyeden F2 tuşuna basmalı. F2 tuşuna basıldığında şifre girilecek alan eğer gerçekten bir şifre alanı değilse eklenti kullanıcıyı uyarmaktadır. Bu eklenti her bir site için o siteye özgü bir şifre üretmektedir ve bu sayede şifrenin birden fazla sitede kullanım problemi ve yemleme saldırılarına engel olunmaktadır.

PwdHash eklentisinin görsel bir arayüzü yoktur, bu sebeple kullanıcılar eklentiyi kullanarak veya kullanmadan şifre girilmesi arasındaki farkı anlayamamaktadırlar. Bu ve bunun gibi kullanışlılık problemleri incelenecektir. Eklentinin ulaşılabiliyor olmadığı durumlarda ise kullanıcılar www. pwdhash. com sitesini kullanarak şifrelerini elde edebiliyorlar.

4. 2. 2 PwdHash ve GPEX Kullanışlılık karşılaştırması

Kullanışlılık testi yapmanın bir yöntemi laboratuar ortamında sistemin potansiyel kullanıcılarıyla yapılacak olan testlerdir. Bu bağlamda bir şifre yöneticisi olarak test etmek istediğimiz GPEX eklentisini ve metin tabanlı bir şifre yöneticisi olan PwdHash eklentisini kullanışlılık yönünden karşılaştırdık. PwdHash den başka eklenti olarak gerçekleştirilmiş "Password Multiplier"[17] gibi şifre yöneticileri de vardır fakat yapılmış olan çalışmalar [35] PwdHash eklentisini daha kullanıcı dostu ve güvenli buldukları için PwdHash ile karşılaştırma yapmayı uygun gördük. Deney tasarımımız ve metodolojimiz [35] in yazarlarınınkiyle aynı olduğu için sonuçlarımızı onların sonuçları ile karşılaştırdık.

4. 2. 2. 1 Deney

Ortadoğu Teknik Üniversitesinde öğrenci olan ve 10 bayan ve 10 erkekten oluşan 20 gönüllü katılımcının yaş ortalaması 24. 8 dir. Katılımcılar hem GPEX eklentisini hem de PwdHash eklentisini kullanmışlardır ve kullanım önceliği katılımcılar arasında dengelenmiştir.

Deneyin başlangıcında katılımcılar internet kullanımları ve internet güvenliği anlayışları ile ilgili bir anket doldurmuşlardır. Bu ankette bütün katılımcılar interneti günlük olarak kullandıklarını belirtmişlerdir ve internet güvenliğine karşı tutumları Çizelge 4. 1 de gösterilmektedir.

Çizelge 4. 1 İnternet Güvenliği Anketi

Soru	**Kullanıcı Sayısı**
Bir şifrenizi bazen birkaç internet sitesinde kullanır mısınız?	%85 (N=17)
Şifrenizin güvenliği ile ilgili kaygı duyuyor musunuz?	%80 (N=16)
Şifre Seçimi Kriterleri	
Hatırlaması kolay	%80(N=16)
Başkaları için tahmin etmesi zor	%65(N=13)
Sistem tarafından önerilen	%0(N=0)
Diğer şifrelerimle aynı	%15(N=3)
Diğer	%15(N=3)
Kişisel veya finansal ayrıntılar gerektiren çevrimiçi aktivitelere katılım	
Alışveriş	%65(N=13)
Bankacılık	%65(N=13)
Fatura ödeme	%35(N=7)
Diğer aktiviteler	%90(N=18)

Bu tabloda çarpıcı bir sonuç katılımcıların %80 inin şifrelerinin güvenliği ile ilgili kaygı duyuyor olmalarıdır ve bu durum yeni ve güvenli bir şifre sistemine ihtiyaç olduğunu göstermektedir.

Gerçekleştirilen deneyde katılımcılar giriş(login), taşınma (migrate), güncelleme(update) ve ikinci giriş(second login) işlemlerini hem PwdHash için hem de GPEX için gerçekleştirmişlerdir. Bu metodoloji [35] deki metodoloji ile aynıdır fakat uzaktan erişim (eklentinin ulaşılamadığı durumlarda şifreyi bir internet sitesi vasıtasıyla elde etmek) bizim deneylerimizin arasında yoktur çünkü bu kısım GPEX ve PwdHash ile yaptığımız giriş deneyinden çok farklı değildir. Katılımcılar deneyi www. gmail. com üzerinde tamamladılar ve görevlerin sırası da deney sırasında dengelenmiştir. Giriş görevinde kullanıcılar bir eklentiyi kullarak e-posta hesaplarına ulaşmak zorundadırlar. Taşınma görevinde kullanıcılar eklenti kullanmadan oluşturdukları eski şifrelerini eklenti vasıtasıyla bir şifre üretip bununla değiştirirler.

Güncelleme kısmında kullanıcılar eklenti ile ürettikleri şifreyi yine eklenti ile ürettikleri başka bir şifreyle değiştirmelidirler. Son olarak kullanıcılar değiştirilmiş şifreleriyle ikinci defa e-posta hesaplarına ulaşırlar. Katılımcılar bir görev bittikten sonra zaman kaybetmeden diğer göreve geçmektedirler ve bütün görevler bittiğinde kullanıcılara kullanışlılık, memnuniyet ve güvenlik algılayışları ile ilgili sorular sorulmuştur. GPEX ve PwdHash eklentileri iki farklı dizüstü bilgisayara kurularak burada anlatılan görevler bu bilgisayarlarla gerçekleştirilmiştir. Deneye başlamadan önce katılımcılar GPEX ve PwdHash hakkında bilgilendirilmişlerdir. Aynı [35] deki gibi öğrenme ve hatırlama zorluğunun etkilerini en az seviyeye indirmek için kullanıcılara sabit bir şifre verilmiştir. Kullanıcılar deneyi tamamlarlarken sesli düşünmeleri de istenmiştir.

4. 2. 2. 2 Sonuçlar

Her bir görevi bir kategori altında sınıflandırarak değerlendirdik. Bu kategoriler, başarılı (görevin ilk denemede tamamlanması), tehlikeli başarılı (görevin birden fazla denemede tamamlanması), başarısız (görevin tamamlanamaması), yanlış tamamlama (görev tamamlanamadığı halde görevin tamamlandı sanılması). Kullanıcının eklentiyi aktif hale getirmeden şifresini yazması gibi durumlar birer güvenlik tehlikesi sayılır[35]. Açıkça görülebilir ki, görevi birden fazla denemede tamamlamak GPEX için bir güvenlik açığı değildir. Deney sonuçları Çizelge 4. 2 de gösterilmektedir.

PwdHash eklentisinde bütün görevler için başarı oranı %100 ün altındadır fakat hiçbir katılımcı başarısız olmamıştır veya yanlış tamamlama yapmamıştır. Deney sonuçlarına gore katılımcıların şifre yöneticisi kullanım performansları GPEX için daha başarılıdır.

Katılımcılar Çizelge 4. 3 de gösterildiği gibi bir ankete tabi tutulmuşlardır ve bu anket [35] dekiyle aynı ankettir. Soruların bazıları önyargıyı engellemek için olumlu ve olumsuz şekillerde sorulmuştur ve katılımcıların tamamı ankete katılmışlardır.

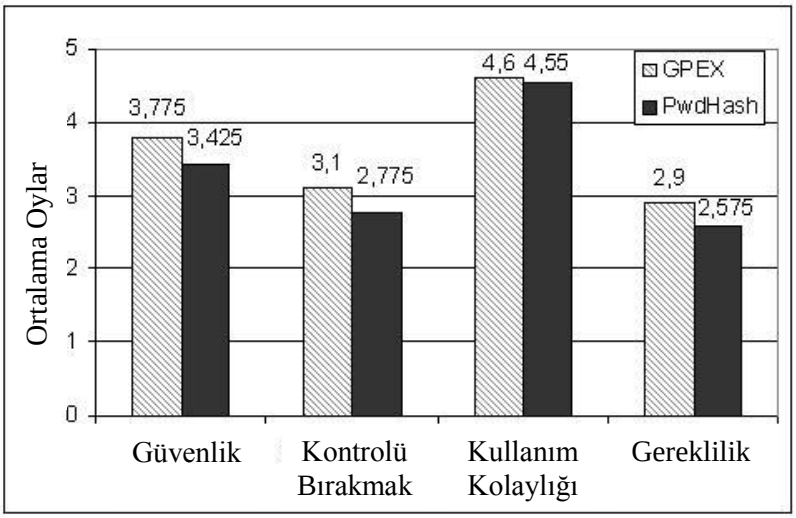
Çizelge 4. 2 GPEX ve PwdHash için görev tamamlama sonuçları

GPEX – Görev tamamlama sonuçları				
	Başarılı	**Tehlikeli Başarılı**	**Başarısız**	**Yanlış Tamamlama**
GPEX – Giriş	100%	N/A	0	0
GPEX-Taşınma	100%	N/A	0	0
GPEX-Güncelleme	100%	N/A	0	0
GPEX-İkinci Giriş	100%	N/A	0	0
PwdHash – Görev tamamlama sonuçları				
	Başarılı	**Tehlikeli Başarılı**	**Başarısız**	**Yanlış Tamamlama**
PwdHash – Giriş	70%	30%	0	0
PwdHash -Taşınma	80%	20%	0	0
PwdHash - Güncelleme	95%	5%	0	0
PwdHash -İkinci Giriş	85%	15%	0	0

Çizelge 4. 3 GPEX – PwdHash Deney Sonu Anket Soruları

Güvenlik
PwdHash – GPEX kullanırken şifrelerim güvendedir.
PwdHash – GPEX in şifrelerimi internet suçlarına karşı koruyabileceğine inanmıyorum
Şifre kontrolünü programa bırakmaktaki rahatlık
Gerçek şifremi bilmiyor olmamdan rahatsız değilim
Kullanıcılar gerçek şifre değerlerini bilmediklerinde daha güvenlidirler.
Kullanım Kolaylığı
PwdHash – GPEX in kullanımı zordur.
PwdHash – GPEX kullanarak kolaylık sitelere giriş yapabildim ve şifrelerimi control edebildim.
Gereklilik ve Kabul
Şifrelerimi korumak için PwdHash – GPEX kullanmalıyım.
PwdHash – GPEX olmadan da şifrelerim güvendedir.

Sonuçlar Şekil 4.12. de gösterilmiştir. Katılımcılar şifrelerinin GPEX ve PwdHash ile güvende olduğuna inanmaktadırlar fakat bu iki system için algılanan güvenlik önemli oranda değişmemektedir($t(19) = 1.677$, $p = .110$). İki sistemin de kullanılışı kolay olarak algılanmıştır fakat yine aralarında önemli bir fark görülmemiştir($t(19) = 0.335$, $p = .741$). Katılımcılar kontrolü bir şifre yöneticisinin eline vermekten rahatsızlardır fakat bu GPEX ve PwdHash için farklılık göstermemektedir($t(19) = 1.748$, $p = .097$). Kullanıcılar bir şifre yöneticisini gerekli bulmamaktadırlar fakat GPEX in sahip olduğu gereklilik algılaması PwdHash den daha yüksektir($t(19) = 2.668$, $p < .05$).



Şekil 4.12. Anket Sonuçları

Bu deneyin bazı sonuçları [35] deki deneyle paralellik göstermektedir. Önceki çalışmalar PwdHash in sahip olduğu büyük bir problem olarak görünür bir ara yüze sahip olamamasını gösteriyor. [35] de kullanıcıların programın nasıl çalıştığını anlamadıklarını söyleyerek bunu kullanıcıların eksik bir zihinsel modele sahip olmalarına bağlıyorlar. GPEX in görünür şifre üretme ara yüzü sayesinde eksik bir zihinsel model problem olmamakta ve bu sebeple meydana gelen başarısız giriş denemeleri önlenmektedir.

Şunu da belirtmek gereklidir ki, GPEX de kullanışlılık problemlerinden arınmış değildir. Bazı katılımcılar boş şifre alanına çift tıklamayı tuhaf bulmuşlardır ve bazıları da klavye kullanarak şifre girmeyi tercih ettiklerini söylemişlerdir.

Bizim yaptığımız deney sonuçları ile [35] de yapılan deney sonuçları arasında görev tamamlanması yönünden önemli bir fark görülmektedir. Yaptığımız deneyde tüm görevlerin sahip olduğu başarı oranı [35] de verilen oranlardan önemli ölçüde yüksektir. Bu farklardan en çok dikkat çekeni şifre güncelleme işlemini [35] de katılımcıların %16 sı başarılı bir şekilde tamamladığı halde bizim yaptığımız deneyde bu oran %95 dir. Bizim deneyimizde katılımcılar internet kullanımına daha alışkın olduklarını ve şifrelerinin güvenliği konusunda daha hassas olduklarını belirtmişlerdir. Ayrıca katılımcılar yaptığımız deneye katılmak için büyük ilgi göstererek gönüllü olmuşlardır. Fakat yaptığımız deney sonuçlarının yüksek oranda farklı çıkmasının sebepleri arasında katılımcıların değişmesinden başka sebeplerin de bulunabileceğini düşünüyoruz.

Yaptığımız karşılaştırmalar ve deney sonuçlarına göre daha somut bir zihinsel model sağladığı için bir şifre yöneticisi olarak GPEX eklentisini kullanmak daha kolaydır.

4. 3 Grafik Şifre Sistemi olarak GPEX

Gpex yaptığımız literatür taramaları ve araştırmalar sonucunda oluşan bilgimize göre grafik şifre yöntemlerini bir tarayıcı eklentisi olarak geliştirme konusunda şu ana kadar yapılmış ilk çalışmadır. Grafik şifre yöntemleri yeni geliştirilen yöntemlerdir ve bu yöntemlerin kullanılmaya başlaması aniden olmayacaktır. GPEX in grafik şifrelerini de alternatif bir kimlik denetim sistemi olarak kullanıma sunmak ve insanların alışkanlıklarını değiştirmek konusunda etkili bir adım olacağını düşünüyoruz.

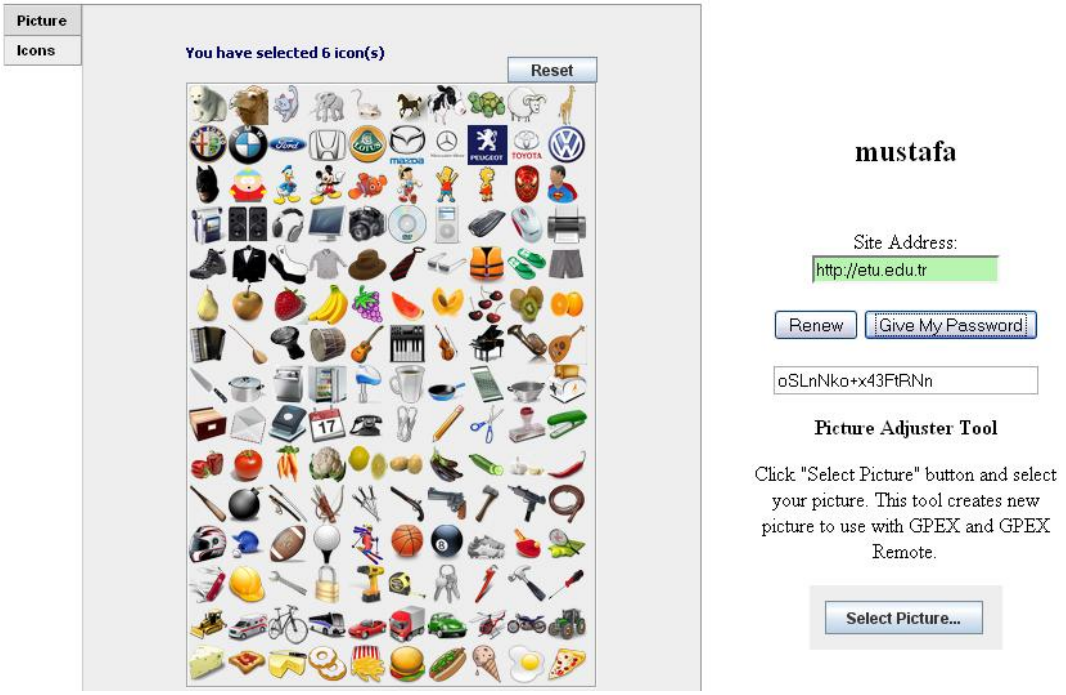
Kolay kurulup kaldırılabilmesi, bir tarayıcıya bağlı kalmadan birçok tarayıcıda farklı kopyalarının kullanılabilmesi, kurulamadığı veya erişilemediği durumlarda uzaktan

erişim imkânı sağlayan internet sitesi, http://test. graphpwd. com/register. php sorunsuz ve kolay kullanım için tasarlanmıştır. Açık kaynak yazılımların içerisinde bir eklenti olarak erini alan GPEX açık kaynak geliştiricilere bir örnek niteliği taşımaktadır. Bunun yanında açık kaynak olmanın getirdiği faydalara da sahiptir. Dünya üzerinde binlerce geliştirici ve kullanıcının kullandığı Firefox ve eklentilerinin sahip olabileceği güvenlik açıkları, eksiklikleri bu kullanıcılar tarafından rapor edilmektedirler ve bu sayede hem güvenliğin hem de gelişimin sürekliliği sağlanmaktadır.

Sıcak nokta problemi şifrenin zorunlu atanması ile giderilebilir bir problemdir. Fakat bunu başarılı bir şekilde gerçekleştirebilmek için kullanışlılığı zorlaştırmamak esastır. Kullanışlılığı kabul edilebilir seviyelerde tutsa da şifrelerin kullanıcılara sistem tarafından atandığı bir yöntemin GPEX içerisinde kullanılması şimdilik mümkün görülmemektedir. Bunun sebebi GPEX in bir eklenti olması ve herhangi bir internet sitesinin sabit bileşeni olmamasıdır. Kullanıcılar sisteme kayıt olurlarken veya şifrelerini değiştirirlerken “eski şifre”, “yeni şifre”, “yeni şifre onay”, “eski şifre onay” gibi şifre alanları bulunmaktadır. Şifre değiştirme sayfasında “eski şifre”, “yeni şifre”, “yeni şifre onay” isimlerinde şifre alanları bulunan bir internet sitesinde GPEX kullanılarak şifre değiştirme işleminin yapılacağı düşünülürse;
Eski şifre alanına GPEX kullanılarak serbest girişe izin verilmelidir. Kullanıcı bu alana serbest bir şekilde eski şifresini girdikten sonra yeni şifre alanına geçtiğinde GPEX artık serbest şifre girişine izin vermeden zorunlu bir şifre atamalı ve bu şifre alanına girmelidir. Yeni şifre onay alanına geçildiğinde ise GPEX yeniden serbest şifre girilmesine izin vermelidir ki kullanıcı zorunlu olarak atanmış şifreyi yeniden girerek onaylayabilsin. Burada anlatılan basit bir şifre değiştirme senaryosunda GPEX in hangi şifre alanına giriş yapıldığını algılaması ve o alana uygun moda geçerek gerekli şifrenin girilmesine imkân sağlaması gereklidir. İnternet formlarında şifre alanları belirli standartlara göre tasarlanmadığı için ve hangi şifre alanının hangi maksatla kullanıldığının GPEX tarafından tespiti mümkün olmadığı için, zorunlu atama ile kimlik denetimini öneren tasarıların GPEX ile kullanılması şimdilik mümkün değildir.

4. 3. 1 Uzaktan Erişim

GPEX in kurulmasının veya kullanılmasının mümkün olmadığı durumlar da olabilir. Böyle durumlarda kullanıcılar şifrelerini elde etmek için uzaktan erişim sistemini kullanabilirler, http://test. graphpwd. com/register. php. Bu sistem eklentiyle aynı grafik şifre ara yüzüne sahip bir internet sitesidir. (Şekil 4.13.)



Şekil 4.13. Uzaktan Erişim Ara Yüz Görüntüsü

Kullanıcılar bu siteye girerek GPEX kullanırken alışık oldukları ara yüzleri bulabilecekler ve kullanabileceklerdir. Kullanıcılar giriş yapmak istedikleri sitenin adını girerler ve şifrelerini tıkladıktan sonra “Give My Password” tuşunu kullanarak kendileri için üretilen şifreye ulaşabilirler. İnternet sitesi kullanıcıların resimlerini yükleme imkânı sunmakla beraber, GPEX ile birlikte kullanıma uygun resimler üretmek için tasarlanmış bir resim ayarlama aracı da vardır. Bu araç resimlerin enini ve boyunu GPEX e uygun bir hale getirdikten sonra resim üzerine ızgara çizgileri çizer. Böylece kullanıcılar uygun gördükleri resimleri kullanabilirler. Kullanıcılar internet sitesine yükledikleri resimlere istedikleri zaman ulaşabilirler.

4. 4 İnternet güvenliği açısından GPEX

Bu bölümde, tarayıcı eklentisi olarak kullanılması düşünülen GPEX i, internet saldırı çeşitleri açısından değerlendireceğiz. Atak çeşitleri ve GPEX in durumu şu şekildedir:

Sözlük Saldırısı (Dictionary Attack): Bu saldırı türünde bir şifrenin ne olduğunun tahmin yöntemi kullanılarak denenmesi söz konusudur. Yapılacak tahminler rastgele değildir. Denenecek olan kelimeler bir sözlük içerisinde toplanır ve bu sözlükteki kelimeler birer birer denenir. GPEX in ürettiği şifreler son derece karmaşık ve site kurallarına uygundur, bu sebeple GPEX ile üretilen şifrelerin bu tür bir atakla tahmin edilebilmesi mümkün değildir. GPEX de kullanılan grafik şifre tasarılarında şifreyi sistem atamamaktadır ve seçim kullanıcıya bırakılmaktadır. Bu sebeple kullanıcılar belli bölgelere yoğunlaşabilirler (sıcak-nokta problemi). Resim üzerinde yapılacak bir çalışma ile belirlenen noktalardan (veya ikonlardan) bir sözlük oluşturularak deneme saldırıları yapılabilir. Bu durumda sıcak-noktalara karşı alınan önlemlere bakılmalıdır. GPEX de kullandığımız tasarılarda sıcak nokta problemi tamamen giderilmemiştir ve bu sebeple GPEX bu saldırıya karşı zayıf sayılabilir. İnternet sitelerinin birkaç denemeden sonra girişi engellemeleri veya belli bir süre sonra girişe izin vermeleri bu saldırıyı yavaşlatır. GPEX bu açıdan sıcak-nokta problem tamamen çözülene kadar internet sitelerinin aldığı önlemlere dayanmaktadır.

Kaba Kuvvet (Brute Force) : Bu saldırı deneme yanılma yöntemiyle mümkün olan bütün ihtimalleri deneyerek doğru şifreyi bulma esasına dayanır. GPEX in ürettiği şifreler site kurallarına uygun olarak üretilir. Eğer girilecek site şifre kuralı olarak bu saldırıya dayanabilecek uzunlukta bir sınır koymuş ise üretilen şifre yeterince karmaşık ve uzun olacaktır. GPEX üzerindeki noktaların veya ikonların denenmesiyle yapılacak bir saldırıda GPEX in kullandığı tasarıların şifre alanı önemlidir. GPEX tasarıları 2^{43} büyüklüğünde bir şifre alanı sağlamaktadır. Bu şifre alanı oldukça büyük bir alan olsa da çevrimdışı saldırılara karşı yeterli olmayabilir. Bu sebeple önlem olarak bazı şifre genişletme teknikleriyle ve kullanılan özet

algoritmasının yavaşlatılmasıyla [33, 17] çevrimdışı ataklara karşı yeterli önlem alınabilir.

Yan Gözle Şifre Çalmak (Shoulder Surfing): Bu saldırı türü, kullanıcı şifresini girerken onu izleyerek bilgilerinin elde edilmesine dayanır. Burada bahsedilen izleme birçok şekilde olabilir, kullanıcının el veya fare hareketlerini izlemek, bir kamerayla izlemek veya kaydetmek veya kullanıcının bilgisayarındaki hareketleri, resimleri, yazıları bir yazılım vasıtasıyla kaydetmek şeklinde olabilir. GPEX bu gibi ataklara karşı bir savunma sistemine sahip değildir. Tuş kaydedici programlarla metin şifrelere karşı yapılan ataklarda sadece klavye tuşlarının kaydedilmesi ve gönderilmesi gerekirken GPEX ile yapılan bir girişi izleyebilmek için bir seri resim veya video kaydedilmesi gerekmektedir. Böyle bir durum daha yoğun veri transferi gerektireceği için bu atak metin tabanlı şifrelere yapılacak bir ataktan daha zordur.

Sosyal Mühendislik (Social Engineering): Sosyal mühendislik saldırıları direk kişilere yönelik oldukları için önlenmesi genellikle kişilerin bilinçlenmeleriyle sıkı sıkıya bağlıdır. GPEX sisteminde özellikle bir tasarıda birçok nesne vardır ve kişi bu nesnelerden seçerek şifresini belirler. Kişilerin seçtikleri nesneleri, ilgi alanlarından ve beğenilerinden bağımsız olarak düşünmek mümkün değildir. Bu sebeple bir kişinin ilgi alanları ve beğenileri öğrenildikten sonra GPEX üzerinde deneme yanılma atakları yapılabilir.

Yemleme(Phishing): Yemleme atakları bir sitenin sahte kopyasını yaparak ve kullanıcıları bu siteye yönlendirerek bilgilerini girmelerini sağlamak esasına dayanır. Yemleme ataklarına karşı adres satırında yazan alan adına dikkat edilmelidir ve eğer şüpheli bir alan adı varsa kritik bilgiler bu sayfaya girilmemelidir. GPEX ürettiği şifreleri sitelerin alan adlarını kullanarak üretmektedir ve bu alan adlarını açık olan siteden otomatik olarak almaktadır. Bu durumda bir site için üretilen şifreyi başka bir site için de üretmesi ve sayfaya yazması mümkün değildir. Sahte kopya olan site içerisinde iken bu siteye özgü bir şifre üretilir ve bu şifre başka hiçbir sitenin değildir.

JavaScript saldırıları: Sahte şifre alanı oluşturulup kullanıcıları kandırarak şifrelerini bu alana girmeleri için ikna eden ve bu gibi esaslara dayanan saldırılara karşı GPEX güvenlidir. GPEX sadece gerçek şifre alanlarında çalıştığı için sahte şifre alanlarıyla yapılan saldırılara karşı güvenlik sağlar.

Bazı zararlı JavaScript kodları kullanıcıların klavye tuşlarını veya fare tıklamalarını kaydederek ya da başka yollarla kullanıcı şifresini girerken veya formu gönderdiğinde şifreyi elde etmeye çalışmaktadırlar. GPEX şifre girişinde klavye tuşları yerine fare imleciyle girişe izin verdiği için klavye tuşlarını dinleyen zararlı kodlar saldırıda başarılı olamazlar. Fare tıklamalarının koordinatlarını kaydeden JavaScript kodları da vardır. GPEX şifre alanına çift tıkladıktan sonra açılan bir pencerede çalışmaktadır ve kullanıcı şifresini açılan bu pencerede girmektedir. Bilgimize göre, ziyaret edilen sayfaya enjekte edilmiş olan ve fare imlecinin hareketlerini dinlemeye yönelik olarak yazılmış bir JavaScript kodu, açılmış olan diğer bir penceredeki fare hareketlerini dinleyemez. GPEX bu tür bir saldırıya karşı güvenlidir. Şunu da belirtmek gerekir ki, tarayıcıda kurulu olan bütün eklentiler güvenilir olmalıdır, bu sayede GPEX ile girilen şifreyi elde etmeye çalışan başka bir tarayıcı uygulaması olmadığına emin olunabilir.

BÖLÜM 5 SONUÇ VE GELECEKTEKİ ÇALIŞMALAR

GPEX için bir tanesi PassPoint tasarısından örnek alınarak geliştirilen bir tasarı olmak üzere iki farklı tasarı kullanılmaktadır. Literatüre bakıldığında ve önceki bölümlerde de anlattığımız gibi PassPoints tasarısı incelenerek sahip olduğu problemlerden en önemlisi olan sıcak-nokta problemi üzerinde durulmuştur [37-41]. Bu probleme çözüm olarak passpoints geliştirilerek yeni grafik şifre tasarıları önerilmiştir ve Cued Click-Points (CCP)[29] de bunlardan bir tanesidir.

CCP de birçok resim arasından her defasında kullanıcın önüne bir adet resim getirilir ve her bir resim üzerinden sadece bir nokta tıklanılabilir. Kullanıcıya gösterilen resim tıklanılan noktaya göre değiştirilir ve kullanıcı değişen resimden bir önceki adımda doğru noktayı tıklayıp tıklamadığına dair bir ipucu elde eder. CCP tasarısını önerenler tarafından gerçekleştirilen çalışmalarda kullanıcıların PassPoints tasarısına göre daha başarılı oldukları ve CCP tasarısını tercih ettikleri bildirilmiştir[29].

CCP tasarısının GPEX sisteminde kullanılmasının uygun olup olmayacağını bu bölümde inceleyeceğiz. Eğer PassPoints tasarısından daha avantajlı sistemlerin GPEX sistemi içerisinde kullanılması mümkünse bunun gelecek çalışmalar arasında yer almasında fayda vardır.

CCP sistemi passpoints sistemine göre bazı üstün yönlere sahiptir. Bunlardan bir tanesi şifre girilmesi tamamlanmadan şifrenin yanlış giriliyor olduğuna dair ipucu vermesidir. Bu durum şifre girişlerinde kullanıcının yanlış şifre girip sayfayı göndermeden (ve sayfa yenilenmesini beklemek zorunda kalmadan) önce hatalı giriş yaptığını fark etmesini sağlayarak zaman kazandırır. Bu ipucu sistemi, resimleri daha önce görmemiş olan, dolayısıyla tanıma imkânı bulunmayan bir saldırgana, saldırısında yardım edecek bir ipucu oluşturmamaktadır.

CCP sisteminin diğer bir avantajı ise, kullanıcıların tıkladıkları noktaların sırasını hatırlamaları gerekmemesidir. Kullanıcılar her resimden sadece 1 tane nokta

hatırlamak zorunda oldukları için, hatırlanması gereken noktaların sırasının karışması mümkün değildir. Bu durum passpoints yönteminde hatırlamayı zorlaştıran ve hatalı girişlere sebep olan sırasıyla hatırlamanın zorluğunu ortadan kaldırmıştır.

CCP tasarısını geliştirenler bu tasarının diğer grafik şifre yöntemleri gibi yan gözle şifre çalma saldırılarına karşı zayıf olduğunu bildiriyorlar. Fakat CCP tasarısını diğer tasarılardan ayıran ve yan gözle çalma ataklarına karşı daha da zayıf bir duruma düşüren bir özelliği, saldırganlar için uzaktan resmin tamamını görüp tanımanın saldırgan için önemli bir bilgi olmasıdır. Passpoints yönteminde bir saldırganın uzak bir bölgeden resmi görmesi, tam tıklama yapıldığı anda fare imlecinin yerini görmedikçe saldırgan için önemli bir bilgi değildir. Fakat CCP de fare imleci kadar küçük bir alanın değil, 331 x 451 (gerçekleştirime göre daha da büyük bir resim olabilir) piksellik bir alanın beş resim için görülmesi önemli bir güvenlik zayıflığıdır. Resimleri gören ve bir daha gördüğünde tanıyabilecek olan bir saldırgan hangi noktanın hangi resmin gösterilmesini tetiklediği bulduğunda, resimleri tanıyarak ilgili noktaları bulabilir. Yan gözle çalma atakları CCP için normal koşullarda ve diğer grafik şifre tasarıları için oluşturduğu tehditten daha büyük bir tehdit oluşturmaktadır.

CCP tasarısında gösterilecek resim belirlenirken kullanıcı adı, gösterilen resim ve tıklanılan bölge olmak üzere üç parametreden faydalanılıyor. İnternet sayfalarında metin kutusu gibi alanların hangi bilgileri istediğini GPEX sayesinde tespit etmek belirli bir standart olmadığı için mümkün görülmüyor. Bu durumda bir sonraki resmi göstermek için gerekli olan kullanıcı adını eklenti penceresine kullanıcının kendisi girmelidir ki, bu durum kullanıcı adının ikinci defa yazılması gibi kullanıcı dostu olmayan bir durumdur. Kullanıcı adının bir şekilde internet sayfasından alındığını ve bir sonraki resmin gösterilmesinde kullanılacağını varsaysak bile, kullanıcı adı her sitede aynı olmadığı için her sitede farklı resimler gösterilecektir ve bu da her site için farklı bir grafik şifre ezberleme durumunu doğurur. GPEX in passpoints ile kullanımında sadece bir tane şifrenin hatırda tutulması yeterlidir. Eğer kullanıcı adı her site için değiştiğinden dolayı, internet sitesinden alınmadığını ve sabit bir kullanıcı adının kullanıcı tarafından girileceğini düşünürsek, bu durumda kullanıcı

GPEX için ayrıca bir kullanıcı adı belirlemiş olmalıdır. Eğer CCP i GPEX için uyarlamak adına sonraki resmi gösterme yöntemini değiştirerek sabitlersek ve her seferinde her nokta aynı resmi gösterecek şekilde ayarlarsak, bu durumda da yukarıda bahsedilen ve CCP için önemli bir tehdit oluşturabilecek yan gözle çalma saldırısı için yapılacak olan analizi kolaylaştırır.

CCP 432 farklı resim gösterebilmektedir, Passpoints tasarısında ise sadece bir adet resim kullanılmaktadır. Passpoints e karşı yapılabilecek bir sıcak nokta saldırısında sadece bir resmin üzerindeki noktaların belirlenmesi yeterliyken, CCP de 432 resmin incelenmesi gerekmektedir, bu durum da saldırgana fazlasıyla büyük bir yük getirir.

CCP in sıcak nokta saldırılarında getirdiği avantajın yanı sıra, nokta sırasını ayrıca akılda tutmayı gerektirmemesi ve anlık ipucu sağlaması GPEX sisteminde kullanılması için tercih ettirici sebeplerdir. Gelecekte yapılacak çalışmalarda CCP tasarısının da bir GPEX seçeneği olarak sunulması GPEX için bir ilerleme sayılabilir.

CCP çalışması saldırganların sıcak noktalara yapabilecekleri ataklara karşı zorluklar getirmektedir fakat kullanıcıların sıcak noktaları seçmeleri hala engellenmemiş veya azaltılmamıştır. Grafik şifre sistemlerinde seçim kullanıcılara bırakıldığında bazı noktalara rağbet olması kaçınılmaz bir durum olarak görünüyor. Bu problemi azaltmak veya ortadan kaldırmak için yapılan bir çalışma PCCP (Persuasive Cued Click - Point) çalışmasıdır[25]. Bu çalışmada kullanıcılara daha güvenli bir şifre seçmeleri için yol göstermek temel fikirdir. Kullanıcılar için zorunlu bir şifre atamadan fakat zayıf şifre seçmelerini de zorlaştırarak daha güvenli şifrelere yönlendirmeyi amaçlamışlardır. CCP sistemin üzerine yeni bir özellik eklenerek elde edilmiş olan bu sistemde, şifre oluşturma aşamasında kullanıcıya resim üzerinde belirlenmiş kare bir alandan seçim izni veriliyor. Kullanıcı gösterilen alan hariç, başka bir alandan seçim yapamıyor. Eğer kullanıcı kendisine gösterilen alandan seçim yapmak istemezse arayüzde bulunan “shufle” düğmesine tıklayarak resim üzerinde başka bir alanın seçilmesini sağlayabiliyor. Burada bahsedilen sistem kullanıcıları daha iyi şifre seçmeleri için ve sıcak noktalara yönelmemeleri için

yönlendirebilir bir sistemdir. Fakat GPEX içinde kullanılması mümkün değildir. Bunun sebebi zorunlu atama yapan sistemlerin GPEX ile kullanılmamasının sebebiyle aynıdır. PCCP de "shuffle" düğmesi ve sadece belirli bir bölgeden seçime izin verilmesi, şifre oluşturulma safhasındadır. Kullanıcı şifresini girerken ise tüm şifre alanını kullanabilmelidir. GPEX ise kullanıcının yeni bir şifre mi belirlediğini yoksa önceden var olan şifresini mi gireceğini otomatik olarak tespit edemediği için bu sistem de zorunlu atamalı sistemler gibi GPEX için uygun değildir. Zorunlu sistem ataması kullanan sistemlerin veya kullanılan tasarının sistemde bilgi kaydetmesi gerektiğinde meydana gelen taşınabilirlik problemine sahip sistemlerin ya da PCCP gibi sistemlerin güvenli ve kullanışlı bir şekilde GPEX içerisinde kullanımına yönelik çalışmalar yapılabilir. Örneğin, kullanılan tasarı sistemde bilgi tutmalı ise başka bir bilgisayara eklenti kurulduğunda bu bilgilerin de taşınması gereklidir. Bu işlem, kritik olamayan kullanıcı bilgilerini bir sunucuya yükleyerek ve eklenti kullanılırken bilgileri bu sunucudan alarak giderilebilir ancak kullanılacak tasarıya ve saklanılacak bilgiye göre, veri alışverişinin nasıl yapılacağı, verilerin hangi formatta tutulacağına karar verilmelidir. Bu çözüm önerisinde daima ayakta olması gereken bir sunucuya ihtiyaç duyulmaktadır. Şifreyi kullanıcılara zorunlu atayan tasarıların kullanımı için ise sunucu tarafında değişiklik yaparak bazı şifre alanlarının kimliklerini standartlaştırmak bir çözüm olabilir ancak, bu çözüm, birçok site ile anlaşmaya varmakla yürürlüğe koyulabileceği için pratik olmayan bir çözüm önerisidir. Bu gibi tasarılar şimdilik GPEX içinde kullanılamamaktadır fakat umut vaat eden tasarılardır bu sebeple bir internet sitesine tümleşik olacak şekilde uyarlanarak uzun vadede deneylerin yapılması ve sonuçlar doğrultusunda tasarıların kullanışlılığının ve güvenliğinin geliştirilmesi gelecek çalışmalar arasında yer alabilir.

Gerçekleştirmiş olduğumuz uygulama içinde kullanılan grafik şifre tasarıları, yapılan incelemeler sonucunda ortaya atılan tasarı teklifleri ve deneyler sonucunda oluşmuştur. Ancak geliştirilen bu uygulamanın henüz başarısı test edilmemiştir ve uzun vadede kullanımı yapılarak kullanıcı memnuniyeti ölçülmemiştir. Bu da gelecekte yapılması uygun olan çalışmalardan birisi olabilir.

Çalışmamızın farklı alanlarda katkıları bulunduğuna inanıyoruz. PassPoints, PwdHash gibi başkaları tarafından sunulan sistemlerin kullanışlılık ve güvenlik deneylerini yaptık ve eskiden yapılmış olan deneylerle karşılaştırdık, bu açıdan farklı kullanıcılarla yaptığımız deneylerle diğer çalışmaları teyit ettik veya farklılıklar gözlemledik. Bu sayede deneydeki değişkenlerin sonuçları nasıl etkilediği konusunda fikir edinilebilir ve bu yapılacak başka çalışmalarda yol gösterici olabilir. Bunun yanında yeni tasarılar sunarak ve gerekli deneyleri ve karşılaştırmaları yaparak grafik şifre alanında yeni gelişmelerde bulunduk. Geliştirdiğimiz uygulama vasıtasıyla, yapılan çalışmaları gerçek hayatta kullanıma sunmuş olduk ve gerçek kullanıcılar tarafından geri bildirim alınmasına imkân sağladık. Geliştirilecek olan sistemler için eklentilerin geliştirime uygun bir ortam olabileceğini gösterdik. Yeni çalışmalar yapanların uygun gördükleri takdirde uygulamamızı inceleyerek kendi çalışmaları için model almaları mümkündür.

KAYNAKLAR

[1] Weinshall, D. and Kirkpatrick, S. Passwords you'll never forget, but can't recall. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'04) (Vienna, Austria, April 24-29, 2004). ACM Press, New York, NY, 1399-1402.

[2] Real User Corporation. The Science Behind Passfaces. http://www.realusers.com, erişim tarihi:21 Eylül 2009

[3] Dhamija, R. and Perrig, A. Déjà Vu: User study using images for authentication. In Ninth Usenix Security Symposium (Denver, CO, USA, Aug. 14-17, 2000). http://www. usenix.org/publications/library/proceedings/sec2000/dhamija.html, erişim tarihi:21 Eylül 2009

[4] B. Kirkpatrick. An experimental study of memory. Psychological Review, 1:602–609, 1894.

[5] S. Madigan. Chapter 3: Picture memory. In J. Yuille, editor, Imagery, Memory, and Cognition: Essays in Honor of Allan Paivio, chapter 3. Picture Memory, pages 65–89. Lawrence Erlbaum Associates, 1983.

[6] A. Paivio, T. Rogers, and P. Smythe. Why are pictures easier to recall than words?, Psychonomic Science, 11(4):137–138, 1968.

[7] R. Shepard. Recognition memory for words, sentences, and pictures. Journal of Verbal Learning and Verbal Behavior, 6:156–163, 1967.

[8] Sonia Chiasson. Usable Authenticaııon and Click-Based Graphical Passwords. *Doctor of Philosophy* School of Computer Science at Carleton University, Ottawa, Ontario December 2008

[9] http://www. pwc.co.uk, erişim tarihi:21 Eylül 2009

[10] http://www.pwc.co.uk/eng/publications/ dti_information_security_breaches_survey_2006. html, Son erişim 21 Eylül 2009.

[11] http://www.computerweekly.com/Articles/2009/03/10/235217/web-users-stick-to-one-password-survey-reveals. htm, erişim tarihi:21 Eylül 2009

[12] http://www. computerweekly. com/Articles/2009/03/10/235217/web-users-stick-to-one-password-survey-reveals. htm, erişim tarihi:21 Eylül 2009

[13] D. Klein. Foiling the cracker: A survey of, and improvements to, password security. In Proceedings of the 2nd USENIX Security Workshop, August 1991.

[14] http://www.schneier.com/blog/archives/2006/12/realworld_passw.html, erişim tarihi:21 Eylül 2009

[15] http://en.wikipedia.org/wiki/Leet, erişim: 21 Eylül 2009

[16] B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. Mitchell. Stronger password authentication using browser extensions. In 14th USENIX Security Symposium, Baltimore, August 2005.

[17] J. Halderman, B. Waters, and E. Felten. A convenient method for securely managing passwords. In 14th International World Wide Web Conference (WWW), 2005.

[18] G. Blonder. Graphical passwords. United States Patent 5559961, 1996.

[19] B. Kirkpatrick. An experimental study of memory. Psychological Review, 1:602–609, 1894.
[20] S. Madigan. Chapter 3: Picture memory. In J. Yuille, editor, Imagery, Memory, and Cognition: Essays in Honor of Allan Paivio, chapter 3. Picture Memory, pages 65–89. Lawrence Erlbaum Associates, 1983.
[21] A. Paivio, T. Rogers, and P. Smythe. Why are pictures easier to recall than words? Psychonomic Science, 11(4):137–138, 1968.
[22] R. Shepard. Recognition memory for words, sentences, and pictures. Journal of Verbal Learning and Verbal Behavior, 6:156–163, 1967.
[23] S. Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy, and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system, " International Journal of Human Computer Studies, to appear.
[24] T.Valentine, "An evaluation of the Passface personal authentication system, " Technical Report, Goldsmiths College, University of London 1998.
[25] S.Chiasson, A. Forget, R. Biddle, P. C. van Oorschot, Influencing Users Towards Better Passwords: Persuasive Cued Click-Points. HCI 2008, September 1-5 2008.
[26] T.Valentine, "Memory for Passfaces after a Long Delay, " Technical Report, Goldsmiths College, University of London 1999.
[27] S.Brostoff and M. A. Sasse, "Are Passfaces more usable than passwords: a field trial investigation, " in People and Computers XIV Usability or Else: Proceedings of HCI. Sunderland, UK: Springer-Verlag, 2000.
[28] D. Davis, F. Monrose, and M. K. Reiter, "On user choice in graphical password schemes, " in Proceedings of the 13th Usenix Security Symposium. San Diego, CA, 2004.
[29] S.Chiasson, P. van Oorschot, and R. Biddle. Graphical password authentication using Cued Click Points. In European Symposium On Research In Computer Security (ESORICS), LNCS 4734, pages 359–374, September 2007.
[30] S.Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy, and N. Memon. Authentication using graphical passwords: Basic results. In Human-Computer Interaction International (HCII 2005), 2005.
[31] S. Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy, and N. Memon. Authentication Using Graphical Passwords: Effects of Tolerance and Image Choice. In Symp. Usable Priv. & Security (SOUPS), 2005.
[32] J. Thorpe, P. C van Oorschot, Human-Seeded Attacks and Exploiting Hot-Spots in Graphical Passwords, pages 103-118. In 16th Usenix Security Symposium, Boston, USA, 2007.
[33] N. Provos and D. Mazieres. A Future-Adaptable Password Scheme. In Proceedings of the USENIX Annual Technical Conference, 1999.
[34] J. P. Van Overschelde, K. A. Rawson, and J. Dunlosky, Category norms: An updated and expanded version of the Battig and Montague (1969) norms Journal of Memory and Language 50 (2004) 289–335

[35] S. Chiasson, P. C. van Oorschot, R. Biddle. A Usability Study and Critique of Two Password Managers. In 15th USENIX Security Symposium 2006, Vancouver, Canada, 2006.
[36] S. Chiasson, J. Srinivasan, R. Biddle, and P. van Oorschot. Centered discretization with application to graphical passwords. In USENIX Usability, Psychology and Security (UPSEC), April 2008.
[37] A. Dirik, N. Menon, and J. Birget. Modeling user choice in the Passpoints graphical password scheme. In 3rd ACM Conference on Symposium on Usable Privacy and Security (SOUPS), July 2007.
[38] K. Golofit. Click passwords under investigation. In 12th European Symposium On Research In Computer Security (ESORICS), LNCS 4734, September 2007.
[39] A. Salehi-Abari, J. Thorpe, and P. van Oorschot. On purely automated attacks and click-based graphical passwords. In 24th Annual Computer Security Applications Conference (ACSAC), 2008.
[40] J. Thorpe and P. van Oorschot. Human-seeded attacks and exploiting hotspots in graphical passwords. In 16th USENIX Security Symposium, August 2007.
[41] P. van Oorschot and J. Thorpe. On predicting and exploiting hot-spots in click-based graphical passwords. Technical report, School of Computer Science, Carleton University, November 2008.
[42] Chris Zarate, http://labs.zarate.org/passwd, erişim tarihi:21 Eylül 2009
[43] Paul Johnston, http://pajhome.org. uk/crypt/md5, erişim tarihi:21 Eylül 2009
[44] Icons, http://www.iconarchive.com, erişim tarihi:21 Eylül 2009

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: YÜCEEL, Mustafa
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 26.07.1983 Ankara
Medeni hali	: Bekâr
Telefon	: 0 (312) 263 83 16
e-mail	: mustafayuceel@gmail.com

Eğitim

Derece	**Eğitim Birimi**	**Mezuniyet tarihi**
Lisans	ODTÜ / Bilgisayar ve Öğretim Teknolojileri	2007

İş **Deneyimi**

Yıl	**Yer**	**Görev**
2007-2009	TOBB Ekonomi ve Teknoloji Üniversitesi	Araştırma Görevlisi
2009- hâlen	TURKSAT A.Ş.	Yazılım Uzmanı

Yabancı Dil

İngilizce

Yayınlar

K. Bicakci, M. Yuceel, B. Erdeniz, H. Gurbaslar, N. B. Atalay, *Graphical Passwords as Browser Extension: Implementation and Usability Study*, Third IFIP WG 11.11 International Conference on Trust Management June 15-19, 2009, Purdue University, West Lafayette, USA.

Kemal Bicakci, Mustafa Yuceel, Nart Bedin Atalay, Hakan Gurbaslar, Burak Erdeniz, *Towards Usable Solutions to Graphical Password Hotspot Problem*, 4th IEEE International Workshop on Security, Trust, and Privacy for Software Applications (STPSA'09), Seattle, USA, July 20 - July 24, 2009.