ADIM ADIM KREDİ KART NUMARASI GÜVENLİĞİ

PCI DSS[1] güvenlik standardının gereği kredi kart numaralarını, PIN, CVV2,vb gibi bilgileri korumak gerekir. Kredi kart numaraları Luhn[2] algoritmasına göre doğrulanır.

Luhn Algoritması adım adım aşağıdaki gibi doğrulanır:

Örneğin hayali **1234 5678 9123 4563** numarasını ele alalım:

1.Adım:

1234 5678 9123 4563

Tek hanelerdeki rakamlar toplanır.
1, 3, 5, 7, 9, 2, 4, 6

1+3+5+7+9+2+4+6 = 37

2.Adım:

1234 5678 9123 4563
Çift hanelerdeki rakamlar 2 ile çarpılır ve elde edilen sayıların basamakları toplanır.
2, 3, 4, 6, 8, 1, 3, 5, 3 rakamlarının iki katı alınır. Sonuç olarak 4, 6, 8, 12, 16, 2, 6, 10, 6 sayıların rakamları toplanır.

4+6+8+3+7+2+6+1+6 = **43**

3.Final:

Son olarak hesaplanan iki sayı toplanır ve 10'un katı olması beklenir.

37+43 = **80.** Görüldüğü üzere 10'un katı olduğundan kredi kartı geçerlidir. Daha önce kredi kart numarasını bir siteden[3] deneyerek bulmuştum. Bunun içinde istediğiniz kart numarasını yazın ve son hanesini 0'dan 9'a deneyin. En kötü ihtimalle 10 denemede bir geçerli kart numarası elde edersiniz.

Kredi kart numaralarının başlangıç numaralarına göre hangi sağlayıcıya ve bankaya ait olduğu anlaşılabilmektedir.[4]

Son olarak python 3.x kullanarak kredi kart numarasının geçerli olup olmadığını gösteren ve hangi sağlayıcıya ve bankaya (db'de var ise) ait olduğunu bulan bir script yazdım. Bunu PCI DSS testleri yaparken ele geçirdiğimiz kart numaralarını en azından Luhn algoritmasına uyuyor mu diye kontrol etmek için kullanmıştım. Script aşağıda: (geliştirilmeye açık)

```
# Written by Bilgehan TURAN at 01.03.2012
# license:Copyleft :)
```

```
import sys
import re

def luhn_checksum(card_number):
    def digits_of(n):
        return [int(d) for d in str(n)]
    digits = digits_of(card_number)
    print ("length of digits:",len(digits))
    if len(digits)< 14 or len(digits) > 16:
        print("Wrong number of creditcard digits. Sorry!! Try Again")
        exit()
    odd_digits = digits[-1::-2]
    #print ("odd digits", odd_digits)
    even_digits = digits[-2::-2]
    #print ("even digits:", even_digits)
    checksum = 0
    checksum += sum(odd_digits)
    for d in even_digits:
        checksum += sum(digits_of(d*2))
    return checksum % 10

def is_luhn_valid(card_number):

    return luhn_checksum(card_number) == 0
def searchVendorId(id):
    twoDigit=int(id[0:2])
    def mainSupplier(twoDigit):
        if twoDigit in range(40,49):
            return 'Visa'
        elif twoDigit in range(50,56):
            return 'MasterCard'
        elif twoDigit == 56:
            return 'Maestro'
        elif twoDigit == 37:
            return 'AmericanExpress'
        else:
            return 'Not Main Issuer.Other company'
    print ("Main Supplier is:",mainSupplier(twoDigit))
    f = open(ccDbPATH,mode='r')
    lines=f.readlines()
    for i in range(6):
        for l in lines:
            #print (l)
            if id in l:
                #primitive search. Just find if it is inside. But it is ok to use by knowing this!!
                print ("Found while searching",id,":",l)
                exit()
        id=id[0:6-i]
   # print (id)
    f.close()
#************** MAIN FUNCTION *********************
ccDbPATH='CheckCreditCardValidity.db'
cc = input("Please enter creditCardNumber:")
print ("CreditCard Entered:",cc)
VendorId=cc[0:6]
if is_luhn_valid(cc):
    print("CreditCard Number is valid")
    print("Searching for db to identify potential issuer & bank...")
    searchVendorId(VendorId)
else:
    print(" CreditCard Number is invalid!!")

**********************************************************************
```

Kredi kart numaraları için koyulan doğrulama sadece kredi kart numarasının geçerli olup olmadığını kontrol eden bir mekanizmadır ve ekstra bir güvenlik getirmemektedir.

Kredi Kart numaranızı korumak için aşağıdakileri adım adım yapmanızı tavsiye edilir

1) Kredi kartlarınızı internette ve "mail order" a kapatın. Sanal kart yaratıp limitinizi ihtiyacınıza göre ayarlayın. Böylelikle kredi kart bilgileriniz ele geçirilse bile zararınızı minimuma indirmiş olursunuz.

2) Emin olmadığınız sitelerden alışveriş yapmayın. Son zamanlarda birçok banka sanal alışveriş sitelerine sanal pos hizmeti vermektedir. Bazı bankalar PCI DSS için istenen bir ASV (Approved Scanning Vendor)[5] tarafından internet güvenlik denetimi istese de çoğu banka herhangi bir zorunluluk koşmadan bu hizmeti vermektedir. Sanal alışveriş sitesi girdiğiniz bilgileri depolayabilir ve kötü niyetli olarak kullanabilir/kullandırtabilir veya istemeden kendi güvenlik zafiyetlerinden dolayı herhangi bir kötü niyet gütmeden sizin kredi kart bilgilerinizin de çalınmasına sebep olabilir.

3) 3D Secure hizmetini kullanın[6]. Yukarıda bahsedilen riski ortadan kaldırmak için Visa tarafından öncülüğü yapılan ve Mastercard'ın da katılmasıyla yaygınlaşan bu hizmeti kullanmanızı öneriyorum. Büyük sanal alışveriş sitelerinde bu yaygınlaştı. Bu hizmet ile kritik bilgileri (CVV2,Expire date, vb) bankanın sağladığı servis aracılığı ile giriyorsunuz. Ayrıca 2.faktör doğrulama olarak SMS OTP(One Time Password) olarak kullanılıyor. Ayrıca bankaya bağlandığınızdan da emin olmanız için daha önce ayarladığınız sadece size özel mesaj, resim sizi karşılıyor. Böylelikle kredi kart bilgilerinin sanal alışveriş siteleri tarafından depolanma olasılığı da ortadan kalkıyor.(PCI DSS standardı gereği şifreli dahi olsa CVV2, PIN gibi bilgiler depolanamaz.)

4) Eğer kredi kartınız internete ve mail order'a kapatmadıysanız, kartınız ile işlem yaptığınızda kesinlikle kartınızın yanında bulunun ve herhangi bir şekilde bilgilerin çalınıp çalınmadığına dikkat edin. Çünkü internet üzerinden alışveriş yapmak için kredi kart üzerinde tüm bilgiler mevcuttur.(isim soyisim, kart numarası, CVV2, Son Kullanma tarihi) Buna ek olarak kredi kart ekstrenizi kontrol etmiyorsanız, sizden çekilen küçük miktarları anlamazsınız. (bkz.Salami Attacks). Bu işlem sanal kartınız için de geçerlidir. Özet olarak ekstrenizi kontrol edin. En azından bundan sonraki işlemleri kurtamış olursunuz.

5) Kredi kartınızdan yapılan alışverişlerde size uyarı gelmesini (SMS, email, vb) sağlayın. Zararın neresinden dönerseniz o kadar iyi. Bunu bir erken uyarı sistemi olarak düşünebilirseniz.

KAYNAKLAR

[1] https://www.pcisecuritystandards.org/
[2] http://en.wikipedia.org/wiki/Luhn_algorithm
[3] http://creditcardity.com/
[4] https://en.wikipedia.org/wiki/List_of_Issuer_Identification_Numbers
[5] https://www.pcisecuritystandards.org/approved_companies_providers/approved_scanning_vendors.php
[6] http://en.wikipedia.org/wiki/3-D_Secure