ÖN BİLGİ

Bu belge

- https://www.bgasecurity.com/makale/dns-hizmetine-yonetlik-dosddos-saldirilari/

resmi adresindeki veya linkin kırık olması ihtimaline karşın alternatif olarak

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/DNS%20Hizmetine%20Y%C3%B6nelik%20DOS%20ve%20DDOS%20Sald%C4%B1r%C4%B1lar%C4%B1.pdf

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

1)

DNS UDP üzerinden çalışan basit bir protokoldür.

(page 4)

2)

DNS güvenliğinden kasıt genellikle DNS kullanılarak gerçekleştirilen DNS Cache Poisoning ve erişebilirliği hedef alan DoS saldırıları olmaktadır.

(Page 4)

3)

DNS'in UDP üzerine kurulmuş olması ve UDP üzerinden gerçekleştirilen iletişimde kaynak IP adresinin gerçek olup olmadığını anlamanın kesin bir yolunun olmaması saldırganın kendini gizleyerek saldırı gerçekleştirmesini kolaylaştırmakta ve engellemeyi zorlaştırmaktadır.

(page 4)

4)

DNS Nedir?

DNS temelde TCP/IP kullanılan ağ ortamlarında isim-IP ya da IP-isim dönüşümü yapar ve e-posta trafiğinin sağlıklı çalışması için altyapı sunar. Günümüzde DNS'siz bir ağ düşünülemez denilebilir. Her yerel ağda – ve tüm internet ağında – hiyerarşik bir DNS yapısı vardır.

Mesela bir epostanın hangi adrese gideceğine DNS karar verir. Bir web sayfasına erişilmek istendiğinde o sayfanın nerede olduğuna DNS üzerinden karar verilir.

(Page 5)

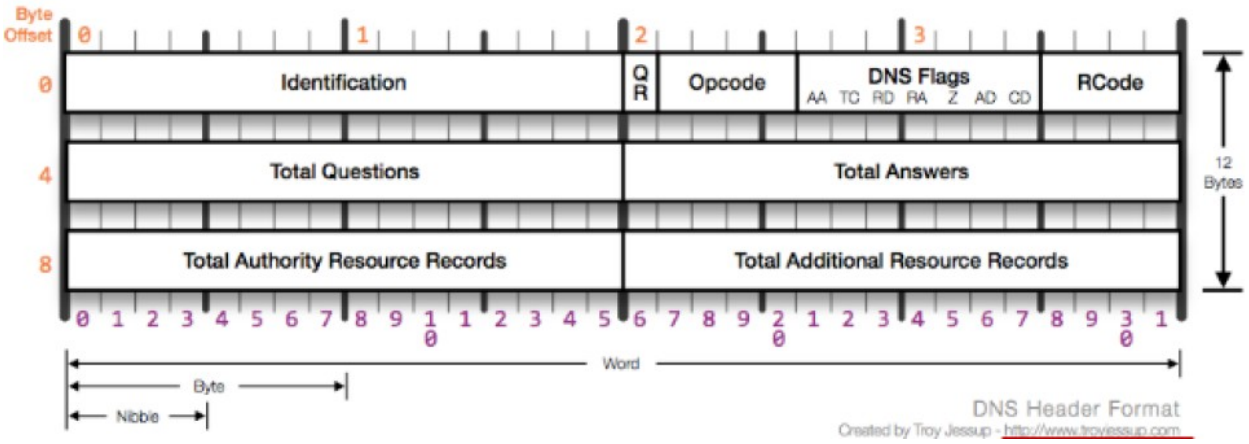
5)

Bir sistemin DNS sunucusunu ele geçirmek o sistemi ele geçirmek gibidir. (?)

(Page 5)

6)

DNS UDP temelli basit bir protokoldür. DNS başlık bilgisi incelendiğinde DNS sunucusundan gelen yanıtın başlık bilgisi aşağıda verilen resimdeki gibidir:



Detaylı DNS başlık bilgisi incelemesi için

http://www.networksorcery.com/enp/protocol/dns.htm

adresinden faydalınabilir

NOT: Yukarıdaki resimde bir satır 2 + 2 = 4 byte'tır. 3 tane satır olunca 12 olur. Dolayısıyla DNS header'I 12 byte'tır.

(Page 6)

7)

DNS Paket Boyutu

DNS paketi denildiğinde akla DNS isteği ve DNS cevabı gelmektedir. Bir DNS paketinin boyutunu Dig komutunun çıktısındaki son satırda yer alan MSG SIZE kısmını inceleyerek öğrenebiliriz.

> dig www.bga.com.tr @8.8.8.8 // Google'ın DNS sunucusuna bga'nın IP'sini sorduk.

Output:

```
; <<>> <<>> www.bga.com.tr @8.8.8.8
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15731
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
ADDITIONAL: 0

;; QUESTION SECTION:
;www.bga.com.tr. IN A

;; ANSWER SECTION:
www.bga.com.tr. 59 IN A 50.22.202.162

;; Query time: 225 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Sun Jan 8 07:28:27 2012
;; MSG SIZE rcvd: 48
```

Yukarıdaki dig komutu ile Google'ın DNS sunucusu yanıt olarak bize bir DNS paketi gönderdi ve dig komutu bu paketi inceleyerek IP'sini sorduğumuz web sitesinin IP'sini ;;ANSWER SECTION kısmında bize sundu. Aynı zamanda alınan dns paketinin boyutunun 48 byte olduğunu da en son satırda bize belirtti.

NOT: DNS cevap paketinin, yani google'ın DNS sunucusundan gelen paketin boyutu 512 Byte'ı aşarsa DNS cevap paketi UDP değil de TCP üzerinden bize gelmek ister ve bize bununla ilgili önceden bir bilgi (Truncated paketi) gelir.

(Page 6-7)

8)

DNS sorgusu istek ve cevap mantığıyla çalışan bir protokoldür. İsteklerin çeşitleri de kayıt tipleri olarak belirlenir. Kayıt tiplerinden en sık kullanılanlar aşağıdaki gibidir:

DNS Kayıt Tipi	İşlevi	Örnek Sorgulama
A	Belirtilen domain'in IP adresini ister.	$dig A abc.com
MX	Belirtilen domain'deki eposta sunucusunun adresini ister	$dig MX abc.com
NS	Belirtilen domain'deki sorumlu DNS sunucusunun adresini ister	$dig NS abc.com
TXT	Belirtilen domain'in DNS sunucusunun özelliklerini ister.	$dig TXT abc.com
PTR	Belirtilen IP adresine ait domain'leri ister.	$dig -x ipAdresi

nslookup, host veya dig komutları ile yukarıdaki DNS kayıt tipleri kullanılarak DNS sorgusu yapılabilir.

9)

Dig İle DNS Sorgularını Yorumlama

Dig komutunun örnek bir çıktısını aşağıda görmektesiniz:

 # dig www.lifeoverip.net
; <<>> DiG 9.3.3 <<>> www.lifeoverip.net
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47172
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;www.lifeoverip.net. IN A

;; ANSWER SECTION:
www.lifeoverip.net. 14400 IN A 80.93.212.86

;; AUTHORITY SECTION:

lifeoverip.net. 30637 IN NS ns3.tekrom.com.
lifeoverip.net. 30637 IN NS ns4.tekrom.com.

;; ADDITIONAL SECTION:
ns4.tekrom.com. 91164 IN A 70.84.223.227
ns3.tekrom.com. 165971 IN A 70.84.223.226
;; Query time: 213 msec
;; SERVER: 1.2.39.40#53(1.2.39.40)
;; WHEN: Sat Jan 24 10:56:14 2009
;; MSG SIZE rcvd: 130

Çıktının detaylı açıklaması şu şekildedir:

- ➔ Status:NOERROR

  Sorgulanan domain adının sorgulanan DNS sunucusunda var olduğunu ve bu domainden sorumlu DNS sunucusunun sorgulara sağlıklı cevap verdiğini gösterir.

- ➔ Status:SERVFAIL

  Sorgulanan domain'in adının sorgulanan DNS sunucusunda var olduğunu fakat domainden sorumlu DNS sunucusunun sorgulara sağlıklı cevap veremediğini gösterir.

- ➔ Status:NXDOMAIN

  Olmayan bir domain adı sorgulandığında cevap olarak dig Status:NXDomain çıktısı üretir.

- ➔ DNS sunucusuna giden sorguyu gösterir.

  ;; QUESTION SECTION:

  ;www.lifeoverip.net. IN A

- ➔ DNS sunucusundan gelen yanıtı gösterir.

  ;; ANSWER SECTION:

  www.lifeoverip.net. 14400 IN A 80.93.212.86

- ➔ Sorgulanan domain'den sorumlu DNS sunucu adreslerini gösterir.

  ;; ADDITIONAL SECTION:
  ns4.tekrom.com. 91164 IN A 70.84.223.227
  ns3.tekrom.com. 165971 IN A 70.84.223.226

- ➔ Sorgulamanın ne kadar sürdüğünü gösterir.

;; Query time: 213 msec

➔ Sorgulamanın hangi DNS sunucusuna yapıldığını gösterir.

;; SERVER: 1.2.39.40#53(1.2.39.40)

(Page 10)

10)

Ön Belleğe Alma (caching)

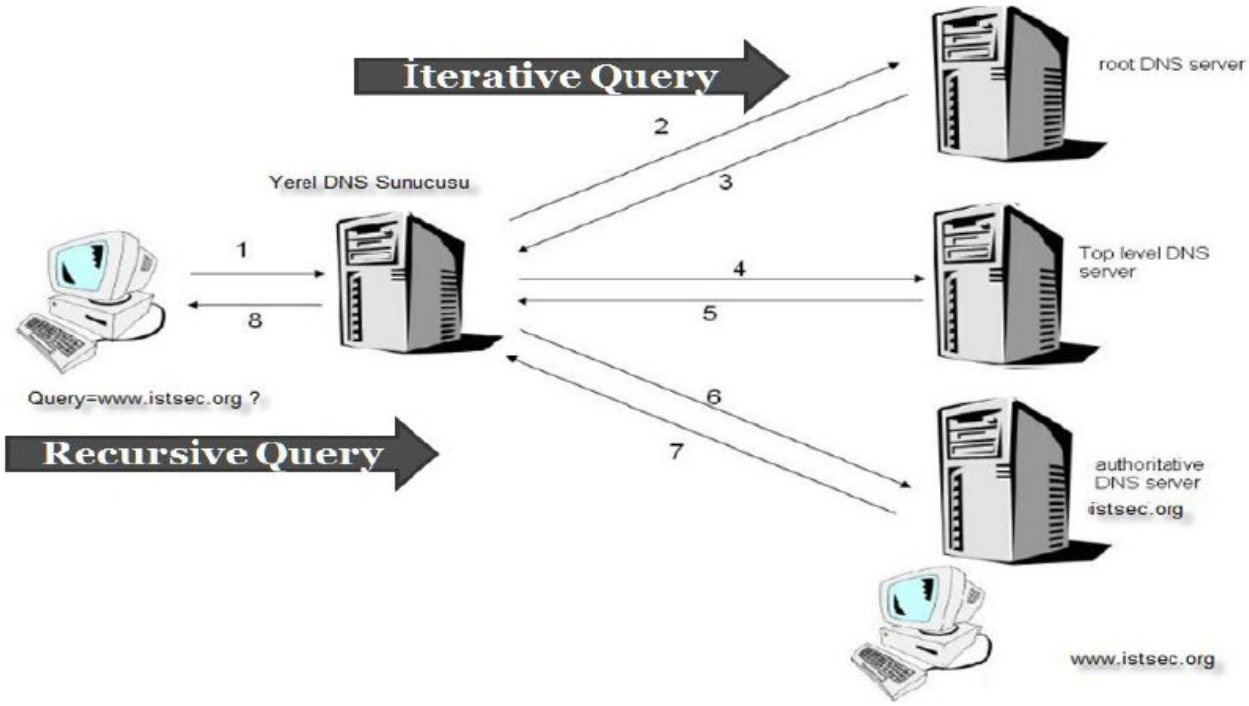
Yapılan bir dns sorgusu sonrası sunucudan dönen cevap bir TTL bilgisini içerir. Bu bilgi istemcinin “aynı” domain için yapacağı sonraki DNS sorgulamalarında sorgunun boş yere tekrar DNS sunucusunu meşgul etmesini önler ve sorgu istemcinin cache’ine gider. İstemcinin cache’i de daha önce aldığı IP adresini istemciye yanıt olarak gönderir.

(Page 10)

11)

DNS Sorgu Methodları

DNS sorguları gerçekleşirken iki çeşit sorgu tipi kullanılır. Bunlar iterative sorgular ve recursive sorgular olarak adlandırılır (NOT: A, MX, .. gibi şeyler DNS sunucusundaki DNS kayıt türleridir. Iterative ve Recursive ise DNS sorgusunu uygulama yöntemleridir).



Iterative DNS Sorguları

Iterative sorgu tipinde istemci DNS sunucusuna sorgu yollar ve DNS sunucusundan verebileceği en iyi cevabı vermesini bekler. Yani gelecek cevap ya “ben bu sorgunun cevabını bilmiyorum, şu DNS sunucuna sor” olmalı ya da “bu sorgunun cevabı şudur” şeklinde olmalıdır. Genellikle DNS sunucuları arasındaki sorgulamalar Iterative tipte olur.

Recursive DNS Sorguları

Recursive sorgulama tipinde istemci DNS sunucusuna recursive bir sorgu gönderir ve cevap olarak tam bir cevap alır. Yani sorgulanan domain’e ait bir IP ya da bir hata alır. Recursive DNS sorguları genellikle PC ile DNS sunucuları arasında olur.

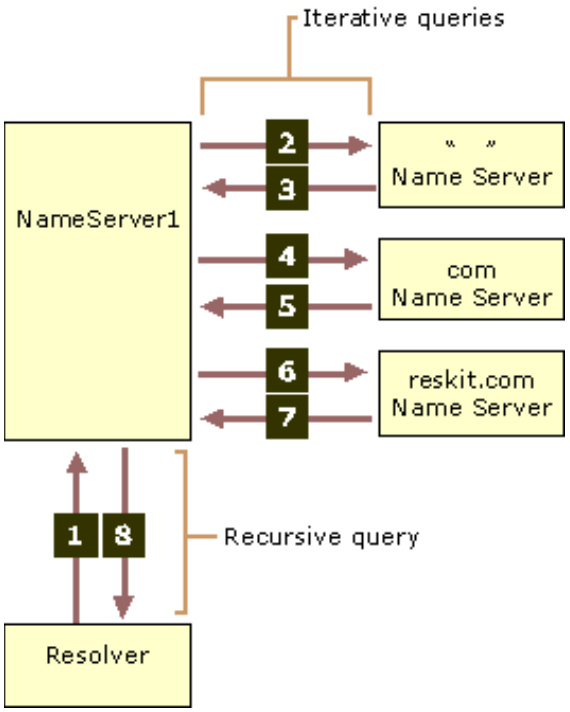
NOT:

Eğer bir PC iterative şekilde DNS sorgusu gönderirse ve sorgu yapılan DNS sunucusunda ilgili kayıt bulunamazsa bu DNS sunucusu yanıt olarak bir

başka DNS sunucusunun IP'sini içeren pointer hükmünde olan bir paket gönderir. Bunun üzerine alıcı PC bu pointer'la diğer DNS sunucusuna sorgu gönderir. Sonuca ulaşana kadar ya da tüm seçenekler (sunucular) elenene kadar bu böyle devam eder.

Eğer bir PC Recursive şekilde DNS Sunucusuna sorgu gönderirse ve sorguyu alan DNS sunucusu kendi içinde istenilen kaydı bulamazsa bu durumda o DNS sunucusu iterative yöntemle diğer DNS sunucularına ilgili kaydı sorar ve en sonunda istenilen cevabı edindiğinde bu cevabı yanıt olarak istemci PC'ye gönderir.

Yani görüldüğü üzere PC'nin Iterative sorgu tipinde yapmak zorunda kaldığı ekstra sorgulamalar Recursive sorgu tipi ile sunucuya yaptırtılmaktadır.



Yukarıdaki resimde yer alan Resolver (client) ile DNS sunucusu arasındaki sorgu tipi Recursive olarak, DNS sunucusu ile diğer DNS sunucuları arasındaki sorgu tipi de Iterative olarak gösterilmiştir.

Yararlanılan Kaynak:
https://technet.microsoft.com/en-us/library/cc961401.aspx

(Page 10-11)

12)

Genele Açık DNS Sunucuları

Herkese açık DNS sunucuları (public DNS sunucuları) kendilerine gelen tüm istekleri cevaplamaya çalışan türde bir DNS sunucu tipidir. Bu tür sunucu genele açık olmaması gerekirken genele açıksa bu durumda o DNS sunucusu eksik ve yanlış yapılandırılmış demektir.

(Page 11)

13)

DNS Sunucusunu Public Olup Olmadığını Anlama

dig komutu ile yapılacak sorgu sonucunda cevap olarak belirtiğimiz domain'in IP adresi dönerse hedef DNS sunucu public anlamına gelir. Örneğin aşağıda sorguyu gönderdiğimiz DNS sunucusunun dönen yanıtına bakarak public olduğunu anlayabiliriz:

> dig www.karabuk.edu.tr @8.8.8.8

Output:

```
; <<>> DiG 9.5.0-P2.1 <<>> www.google.com @91.93.119.70
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26294
;; flags: qr rd ra; QUERY: 1, ANSWER: 7, AUTHORITY: 0,
ADDITIONAL: 0

;; QUESTION SECTION:
;www.google.com. IN A

;; ANSWER SECTION:
www.karabuk.edu.tr      10800   IN   A      193.140.9.45

;; Query time: 16 msec
;; SERVER: 91.93.119.70#53(91.93.119.70)
;; WHEN: Sat Jul 24 13:23:59 2010 ;; MSG SIZE rcvd: 148
```

status: NOERROR yerine status: SERVFAIL bildirimi gelseydi bu durumda hedef DNS sunucusu istenen domain'in IP'sini barındırmıyor anlamına gelirdi. Yani bu durumda IP'yi alamamak demek DNS sunucusunun public olmadığı anlamına gelmez. Bilakis status: SERVFAIL demesi yine public olarak hizmet verdiğini gösterir.

Eğer hedef DNS sunucu belirtilen domain'in IP adresini döndürmezse, yani ;;ANSWER kısmı çıktıda yer almazsa ve status: SERVFAIL bildirimi gelmezse bu durumda hedef DNS sunucu public değil anlamına gelir. Aşağıda sorguladığımız bir dns server'dan dönen cevaba bakarak public olmadığını anlayabiliriz:

> dig www.google.com @ns1.gezginler.net

Output:

```
; <<>> DiG 9.6.1-P1 <<>> @ns1.gezginler.net www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33451
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL:
0
;; WARNING: recursion requested but not available

;; QUESTION SECTION: ;www.google.com. IN A

;; AUTHORITY SECTION:
. 518400 IN NS H.ROOT-SERVERS.NET.

;; Query time: 140 msec
;; SERVER: 208.43.98.30#53(208.43.98.30)
;; WHEN: Sat Aug 7 16:18:15 2010
;; MSG SIZE rcvd: 243
```

;;ANSWER SECTION kısmı yoktur. Üstelik status: SERVFAIL de denmemiş. Bu durumda hedef DNS sunucu public değildir sonucuna varırız.

14)

Nmap ile DNS Sunucunun Public Olup Olmadığını Anlama

Yukarıdaki deneme yanılma yöntemini Nmap'e yaptırabiliriz. Bunun için Nmap'in NSE denen Nmap Scripting Engine'i kullanılabilir. Örneğin KBÜ'nün DNS sunucusunu sınayalım. Bunun için önce KBÜ'nün DNS sunucusu adresini öğrenmemiz gerekir:

> dig NS www.karabuk.edu.tr

Output:

...

;; AUTHORITY SECTION:

karabuk.edu.tr, 10800 IN NS ns1.karabuk.edu.tr

...

Ardından Nmap ile hedef DNS sunucu public mi değil mi sınayalım:

> nmap -PN -n -sU -p 53 --script=dns-recursion.nse ns1.karabuk.edu.tr

Output:

Starting Nmap 6.47 (http://nmap.org) at 2016-01-09 12:10 CST

Nmap scan report for ns1.karabuk.edu.tr (193.140.9.2)

Host is up (0.042s latency)

PORT STATE SERVICE

53/udp open domain

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds.

Görüldüğü üzere hedef DNS sunucunun public olduğunu öğrenmiş olduk.

15)

Public DNS sunucular güvenlik açısından tehdit altındadırlar. Bir saldırgan public DNS sunucuya Amplification DNS flood saldırısı düzenleyerek hedef DNS sunucuda ciddi oranda trafik oluşturabilir ve sunucuyu normal muhataplarına hizmet veremez duruma getirtebilir.

NOT: Amplification DNS Flood saldırısı ileride bahsedilecektir.

(Page 13)

16)

DNS Sunucusu Yazılımları

DNS sunucularının DNS hizmeti vermesini sağlayan çeşitli sayıda yazılım vardır. Bunlara ISC Bind, DjbDNS, Maradns ve Microsoft DNS yazılımları örnek olarak verilebilir. Bu yazılımlar arasında en yoğun kullanıma sahip olan ISC Bind'tır. İnternetin %80'lik gibi büyük bir kısmı Bind Dns yazılımı kullanmaktadır.

(Page 13)

17)

DNS Sunucu Yazılımını Belirleme

DNS sunucu yazılımlarını belirlemek için temelde iki araç kullanılır:

1. Nmap
2. Dig, nslookup gibi klasik sorgulama araçları

Nmap Kullanarak DNS Sunucu Yazılımını Belirleme

> nmap -PN -sU -sV ns1.karabuk.edu.tr

Output:

…

PORT STATE SERVICE VERSION

53/udp open domain NSD 3.2.8 – 3.2.10

...

Görüldüğü üzere hedef DNS sunucusunda çalışan server yazılımı NSD (Name Server Daemon) imiş. NSD açık kaynak kodlu bir isim sunucusu yazılımıdır.

Dig Kullanarak DNS Sunucu Yazılımını Belirleme

> dig version.bind chaos txt @ns1.karabuk.edu.tr

NOT: Bu yöntem sadece bind kullanan sistemlerde sağlıklı sonuçlar üretir.

> dig version.bind chaos txt @tr1.lnwdns.net // lncludeKarabuk DNS

Output:

...

;; QUESTION SECTION:

;version.bind. CH TXT

;; ANSWER SECTION:

version.bind. 0 CH TXT "9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.5"

;; AUTHORITY SECTION:

version.bind. 0 CH NS version.bind.

...

18)

İsteğe Göre DNS Paketi Üretmek

TCP/IP paket üreten yazılımları kullanarak dilenilen şekilde DNS paketi oluşturulabilir. Bu işlemi TCP/IP paket üreteçleri aracılığıyla gerçekleştirebilmek için bir DNS paketinde ne gibi header'lar bulunuru bilmek gerekir.

DNS Paketi Üretim Araçları

Güvenlik ve performans testlerinde kullanılmak üzere tercih edilen DNS paketi üretim araçları aşağıdaki gibidir:

Scapy

Mz

Hping

Netstress

(Page 14)

19)

DNS çok önemli bir protocol olduğu için yaygın kullanılan DNS sunucu yazılımları hem güvenlik uzmanları hem de hackerlar tarafından sık sık kurcalanır. Genel olarak DNS sunucularda bulunan güvenlik zafiyetlerini üç kategoride inceleyebiliriz:

• DNS sunucusunun çalışmasını durdurabilecek zafiyetler

- DoS

• DNS sunucusunun güvenliğini sıkıntıya sokacak zafiyetler

• DNS sunucusunu kullanan istemcilerin güvenliğini sıkıntıya sokabilecek zafiyetler

- Saldırgan DNS kaydını değiştirerek mesela facebook'a erişmek isteyen adamı sahte bir facebook'a yönlendirebilir.

(Page 17)

20)

DNS Protokolünde IP Sahteciliği (IP Spoofing)

DNS UDP tabanlı bir protocol olduğu için IP spoofing yapmak mümkündür. IP spoofing yapılabiliyor olması demek hem DNS isteklerinin hem de DNS cevaplarının kaynak IP'sinin sahte olabileceği anlamına gelir. Sahte DNS isteği üretmeyi engelleyecek herhangi bir yöntem bulunmamaktadır (URPF hariç).

UDP katmanında IP spoofing için bir önlem olmaması nedeniyle DNS IP sahteciliğini önlemek için uygulama seviyesinde iki temel önlem almıştır. Bu önlemlerden ilki DNS TXID başlık bilgisinin random olmasıdır, diğeri ise kaynak port numarasının random olarak belirlenmesidir.

(Page 18)

21)

Kaynak Portun Rastgeleliğinin Sorgulanması

DNS cevabı olarak dönen paketlerin kaynak portlarının sabit mi yoksa rastgele mi belirlendiği aşağıdaki Nmap komutuyla belirlenebilir.

> nmap -sU -p 53 --script=dns-random-srcport 8.8.8.8

Output:

Starting Nmap 5.59BETA1 (http://nmap.org) at 2011-12-18 17:43 EET

Nmap scan report for google-public-dns-a.google.com (8.8.8.8)
Host is up (0.041s latency).

PORT STATE SERVICE
53/udp open domain

|_dns-random-srcport: 74.125.38.86 is GREAT: 6 queries in 3.0 seconds from 6 ports with std dev 6324

Nmap done: 1 IP address (1 host up) scanned in 0.66 seconds

Görüldüğü üzere 6 dns sorgusunun 6'sı da farklı kaynak port numarası ile gelmiş. Demek ki google'ın DNS sunucusu (8.8.8.8) kaynak port rastgeleliğini kullanmaktadır.

DNS Transaction ID (TXID) Değerinin Rastgeleliğinin Sorgulanması

DNS cevabı olarak dönen paketlerdeki TXID değerinin sabit mi yoksa rastgele mi belirlendiği aşağıdaki Nmap komutuyla belirlenebilir.

> nmap -sU -p 53 --script=dns-random-txid ns1.abc.com.tr

Output:

```
Starting Nmap 5.59BETA1 ( http://nmap.org ) at 2011-12-18 17:45
EET
Nmap scan report for ns1.abc.com.tr (1.1.3.3)
Host is up (0.0035s latency).

PORT STATE SERVICE
53/udp open domain
|_dns-random-txid: 91.199.73.23 is GREAT: 26 queries in 5.2 seconds
from 26
txids with std dev 21394
Nmap done: 1 IP address (1 host up) scanned in 6.10 seconds
```

Görüldüğü üzere 26 sorgudan da 26 farklı txid başlık bilgisi gelmiştir. Dolayısıyla hedef DNS sunucu txid rastgeleliğini kullanıyor demektir.

(page 19)

22)

DNS ve TCP İlişkisi

DNS paketleri normalde UDP üzerinden gider ve gelir. Fakat eğer DNS sunucusundan gelecek olan DNS paketi 512 byte'ı aşarsa sunucu bu DNS paketini UDP üzerinden değil de TCP üzerinden göndermek ister. Bu isteğini istemciye yollar. İstemci gelen bu isteğe karşı DNS talebini UDP üzerinden değil de TCP üzerinden göndererek DNS sunucusunun isteğini karşılamış olur ve böylece DNS sunucusu DNS cevap paketini TCP üzerinden istemciye yollar.

NOT: DNS sunucusunun “aynı DNS talebini TCP üzerinden gönder” talebi için yolladığı pakette başlık bilgisi olarak TRUNCATED=1 yapılır. Bu şekilde istemci kendisinden aynı DNS talebinin TCP üzerinden gönderilmesi istenildiğini anlamış olur ve öyle de yapar.

NOT2: DNS sunucusu üzerinde TCP/53 portunun açık olup olmadığı bu porta gönderilecek SYN paketlerine SYN/ACK cevabının dönmesi ile anlaşılabilir.

> hping3 -S -p 53 8.8.8.8 -c 2

Output:

HPING 8.8.8.8 (eth0 8.8.8.8): S set, 40 headers + 0 data bytes
len=46 ip=8.8.8.8 ttl=47 id=46413 sport=53 **flags=SA** seq=0 win=5720 rtt=47.1 ms
len=46 ip=8.8.8.8 ttl=47 id=62723 sport=53 **flags=SA** seq=1 win=5720 rtt=47.3 ms

--- 8.8.8.8 hping statistic ---
2 packets transmitted, 2 packets received, 0% packet loss round-trip min/avg/max = 47.1/47.2/47.3 ms

Görüldüğü üzere google’ın DNS sunucusunun TCP/53 portu açıktır. Çünkü hedef DNS sunucusunun TCP/53 portuna gönderdiğimiz 2 pakete karşılık 2 paket gelmiştir.

(Page 20-22)

23)

DNS sunucusu üzerinde TCP/53 portu açık ise bu porta yönelik SYN flood ve TCP Connection flood tipinde DDoS atakları gerçekleştirilebilir. DNS sunucu önünde SYN cookie, SYN Proxy ya da benzeri bir koruma sistemi yoksa DNS sunucu bu saldırılarla kısa sürede hizmet veremez hale gelecektir.

(Page 22)

24)

DNS hizmetine yönelik DoS/DDoS saldırıları iki kategoriye ayrılır:

- Yazılım temelli DoS saldırıları
- Tasarım temelli DoS saldırıları

(Page 23)

25)

Netstress Kullanarak DNS Flood DDoS Atağı Gerçekleştirme

Makinenin bandwidth'ini test için kullanılan Netstress saniyede ortalama 400.000 DNS isteği gönderebilmektedir. Örnek kullanımı;

> ./netstress fullrandom.config -d 8.8.8.8 -a dns -t a -n 4 -P 53

Google'ın DNS sunucusuna DNS Flood saldırısı böylelikle yapılır.

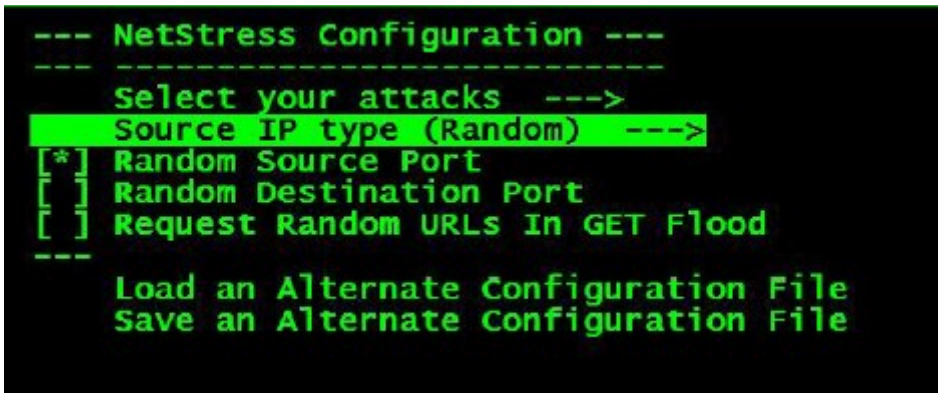
NOT: netstress Kali'de yok. Fakat bir sonraki maddede bu yazılıma değinildiği için burada bahsedilmiştir. İndirme linki şudur: http://sourceforge.net/p/netstressng/wiki/Home/

26)

DNS Flood DoS ve DNS Flood DDoS saldırıları genellikle sahte IP adresleri kullanılarak gerçekleştirilir. Sahte IP adresi kullanımı temelde iki şekilde olmaktadır:

1. Rastgele seçilmiş IP adresleri
2. Bilinen DNS sunucularının IP adreslerinin kaynak olarak kullanımı

Netstress her iki yöntemi de gerçekleştirebilmektedir. Aşağıda rastgele IP seçilerek saldır ayarının yapıldığı aşamayı görmektesiniz:



Bilinen DNS sunucularının IP adreslerinden DNS Flood gerçekleştirme saldırısına gelince bu saldırıda saldırgan kendini bir subnet, bir IP aralığı ya da bir ülke IP adresinden geliyormuş gibi gösterir. Hedef DNS sunucunun önünde IPS varsa IPS aldığı paket sayısının anormalliğinden ötürü ilgili IP'yi kara listeye alır ve engeller. Böylece saldırgan istediği IP'yi DNS Server'a engelletme kabiliyetine sahip olmuş olur.

Saldırgan bir IP'yi ban'latma kabiliyetine sahip olduğu gibi aynı zamanda hedef DNS sunucunun IPS'ini geçip DNS sunucuyu hizmet veremez duruma da sokabilir. Şöyle ki son zamanlarda Türk Telekom, Google ve OpenDNS'in IP adresleri kullanılarak gerçekleştirilen DNS Flood saldırılarına rastlanmaktadır. Bu tip ataklar ile hedef DNS sunucuya sanki diğer DNS sunucuların IP adreslerinden paket geliyormuş izlenimi verilerek hedef sunucunun IPS'inin gelen paketleri engellemesinin önüne geçilmektedir. Çünkü IPS cihazı gelen paketleri legal bir DNS sunucudan geliyor diye çok da olsa kabul eder, engellemez. Sonuçta Türk Telekom'un DNS sunucusu Google'ın DNS sunucusunu niye engellesin ki? İşte saldırganlar bu boşluğu yakaladıkları için son zamanlarda başarılı oluyorlar ve hedef DNS sunucuyu hizmet dışı bırakabiliyorlar.

(Page 27-29)

27)

DNS Performans Ölçümü

Linux sistemlerinde bir DNS sunucunun performansını ölçmenin en temel ve kolay yolu dig tool'unu kullanmaktır. Dig tool'u ile performansını ölçmek istediğimiz bir DNS sunucuya sorgu gönderip gelen cevaptaki ;; Query Time : kısmına bakarak DNS sunucusunun performansını öğrenebiliriz.

> dig www.bga.com.tr @8.8.8.8

Output:

```
; <<>> DiG 9.3.6-P1-RedHat-9.3.6-16.P1.el5 <<>> www.bga.com.tr
@8.8.8.8
;; global options: printcmd
;; Got answer: ;; ->>HEADER<<- opcode: QUERY, status: NOERROR,
id: 57086
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
ADDITIONAL: 0

;; QUESTION SECTION:
;www.bga.com.tr.   IN      A

;; ANSWER SECTION:
www.bga.com.tr. 37        IN      A       50.22.202.163
;; Query time: 41 msec

;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Sat Oct 1 21:26:56 2011
;; MSG SIZE rcvd: 48
```

Yukarıda Google'ın DNS sunucusuna www.bga.com.tr sitesinin IP'sini sormuş bulunmaktayız. Sorgumuzun cevabını ;;ANSWER SECTION kısmından görebiliriz (BGA'nın IP'sini aldık). DNS sunucunun cevap verme süresi ;; Query Time kısmında belirtilmiştir: 41 mili saniye

Şimdi bir de saldırı altındaki bir DNS sunucunun cevap gönderme süresini gözlemleyelim.

> dig www.bga.com.tr @4.2.2.1

Output:

```
; <<>> D <<>> www.bga.com.tr @4.2.2.1
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 17532
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.bga.com.tr. IN A

;; Query time: 326 msec
```

;; SERVER: 4.2.2.1#53(4.2.2.1)
;; WHEN: Sat Oct 1 21:27:54 2011
;; MSG SIZE rcvd: 32

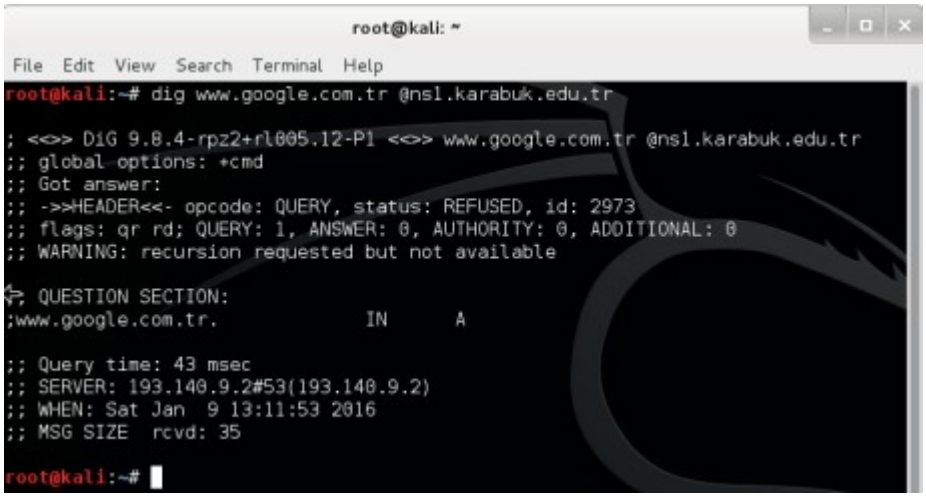
(page 30-31)

28)

Daha önce bahsedildiği gibi client'lar DNS sunucularına recursive sorgu gönderirler. Dolayısıyla public DNS sunucuları recursive sorgulara açıktır. Mesela KBÜ'nün DNS sunucusuna bir sorgu gönderdiğimizde

> dig www.google.com.tr @ns1.karabuk.edu.tr

Output:



status: REFUSED yanıtını alıyoruz ve sorguladığımız domain'in IP'si bize gelmiyor. Bunun anlamı hedef DNS sunucu Recursive sorgulara kapalıdır. Bunu şu örnekle daha net görebiliriz:

> nmap -sU -p 53 -Pn --script=dns-random-srcport ns1.karabuk.edu.tr

Hatırlarsan yukarıdaki komut hedef DNS sunucunun güvenlik önlemi olarak gönderdiği paketlerin kaynak port numaralarını rastgele yapıyor mu yapmıyor muyu öğrenmemizi sağlıyordu. Detaylı bilgi için Madde **21)**'e bakabilirsin. Bu komutu KBÜ'ye uygularsak

çıktıda görüldüğü üzere “ERROR: Server refused recursion” denmiş. Bu KBÜ’nün DNS sunucusunun recursive sorgulara, yani client’ların sorgularına kapalı olduğu anlamına gelir.

Client’ların sorgularına recursive sorgu dendiği burada hatırlatılmak istendi, çünkü bir sonraki maddede bu bilgi işlenecektir.

(Benim Notum)

29)

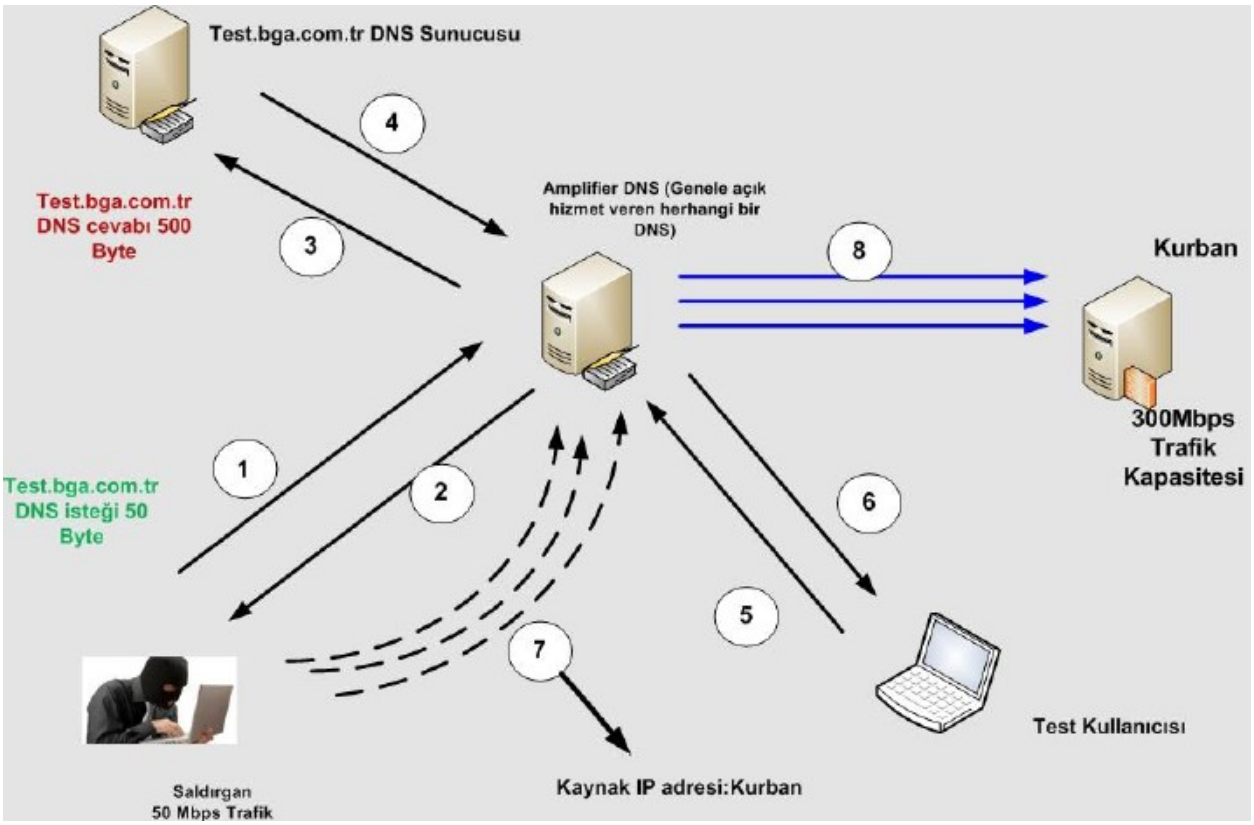
Amplified DNS DoS Saldırısı

Bu saldırı tipi gönderilen DNS sorgusuna karşın dönecek DNS paketinin cevabının kat kat fazla olması özelliğine dayanır. Bu saldırıda saldırgan saldırıyı aracı bir DNS sunucusuna yaptırtır (Yani saldırgan bir DNS sunucusunu kurbana saldırtır). Bu saldırıya Amplified denmesinin nedeni saldırganın mesela 50 byte’lık göndereceği pakete karşılık aracı sunucunun 500 byte’lık bir paketle kurbana yanıt göndermesinden dolayıdır.

DNS Amplification DoS Saldırısı Adımları

1. Saldırgan recursive sorguya açık bir DNS sunucu bulur ve daha önce hazırladığı özel alan adını sorgulatır (Reel hayatta sorgulanan alan adı sadece bir noktadır: “.”) Bu DNS talebinin boyutu 50 byte’tır.
2. Talebi alan aracı DNS sunucu kendi önbelleğinde olmayan bu isteği gidip ana DNS sunucuya sorar. Bu sorgu da 50 byte’tır.
3. Ana DNS sunucu kendisine gelen talebi 500 byte’lık bir paketle yanıtlar.
4. Aracı DNS sunucu cevabı önbelleğine alır ve bir kopyasını saldırgana yanıt olarak gönderir.
5. Saldırgan kendisinin kontrolündeki bir başka bilgisayardan aynı alan adını aracı DNS sunucusuna sorar ve cevabın cache’de olup olmadığını anlamaya çalışır.

6. Aracı DNS sunucu ön belleğinden (cache'den) 500 byte boyutunda bir cevap gönderir.
7. Saldırgan kurbanın IP adresinden geliyormuş gibi en başta sorgulanan alan adına ait DNS paketlerinden aracı sunucuya gönderir (Böylece aracı sunucunun göndereceği DNS yanıtları kurbana gidecektir).
8. Saldırgan göndereceği her 50 byte'lık pakete karşılık aracı sunucu kurbana 500 byte'lık paket gönderecektir. Görüldüğü üzere saldırgan kendi trafiğinin 10 katı kadar trafikle kurbana saldırı yapmaktadır. Saldırgan saniyede ortalama 100.000 DNS sorgusu gönderebileceğini göz önüne alırsak bu üretilen trafiğin saldırgana maliyeti saniyede 100.000 x 50 byte'ken aracı sunucuya saniyedeki maliyeti 100.000 x 500 byte'tır. Sonuç olarak bu yükseltilmiş trafik saldırgan tarafından aracı sonucu üzerinden kurbana gönderilmiş olur.



Özetle aracı sunucu saldırganın trafiğini 10 katı kadar çoğaltarak kurbana saldırmış olur. Buna DNS Amplification DoS saldırısı denir.

> dig . @ns1.tr.net // DNS sunucusuna nokta sorgulanıyor.

Output:

```
; <<>> DiG 9.7.0-P1 <<>> . @ns1.tr.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27323
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL:
14
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;      .  IN      A

;; AUTHORITY SECTION:
. 512544 IN NS k.root-servers.net.
. 512544 IN NS l.root-servers.net.
. 512544 IN NS m.root-servers.net.
. 512544 IN NS a.root-servers.net.
. 512544 IN NS b.root-servers.net.
. 512544 IN NS c.root-servers.net.
. 512544 IN NS d.root-servers.net.
. 512544 IN NS e.root-servers.net.
. 512544 IN NS f.root-servers.net.
. 512544 IN NS g.root-servers.net.
. 512544 IN NS h.root-servers.net.
. 512544 IN NS i.root-servers.net.
. 512544 IN NS j.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net. 598944 IN A 198.41.0.4
a.root-servers.net. 598944 IN AAAA 2001:503:ba3e::2:30
b.root-servers.net. 598944 IN A 192.228.79.201
c.root-servers.net. 598944 IN A 192.33.4.12
d.root-servers.net. 598944 IN A 128.8.10.90
d.root-servers.net. 598944 IN AAAA 2001:500:2d::d
e.root-servers.net. 598944 IN A 192.203.230.10
f.root-servers.net. 598944 IN A 192.5.5.241
f.root-servers.net. 598944 IN AAAA 2001:500:2f::f
g.root-servers.net. 598944 IN A 192.112.36.4
h.root-servers.net. 598944 IN A 128.63.2.53
h.root-servers.net. 598944 IN AAAA 2001:500:1::803f:235
i.root-servers.net. 598944 IN A 192.36.148.17
i.root-servers.net. 598944 IN AAAA 2001:7fe::53

;; Query time: 14 msec
;; SERVER: 195.155.1.3#53(195.155.1.3)
;; WHEN: Mon Jan 23 13:52:52 2012
;; MSG SIZE rcvd: 512
```

Görüldüğü üzere DNS sunucusuna . sorgulandığında 512 byte'lık bir cevap gelmiştir. Bu DNS sorgularının kaynak IP'si kurbanınki yapılarak saldırı düzenlenebilir.

(page 32-34)

30)

DNS Amplified DoS Saldırısı Yapma

[PDF'te anlatılan tool ölmüş artık. Başka bir tane bul]

(Page 34)

31)

DNS Flood DDoS Saldırılarını Tespit Etme

http://www.adotout.com/dnsflood.html adresindeki tool'u indirip aşağıdaki gibi kullanabilirsin.

> cd dns_flood_detector

> dns_flood_detector -i eth0 -t 100 -v –b

Output:

[22:16:17] source [85.95.238.172] - 0 qps tcp : 419 qps udp [22:16:27] source [85.95.238.172] - 0 qps tcp : 139 qps udp

(Page 34)

32)

DNS Flood DDoS Saldırılarını Engelleme

DNS Flood saldırılarını engellemek için kullanılan yöntemler şunlardır:

- Rate Limiting
- DFAS
- DNS Caching
- DNS Anycast

Rate Limiting Yöntemi

Saldırgan UDP/DNS flood saldırısında DNS sunucusuna göndereceği DNS paketlerinin kaynak IP'sini kurbanınki yapar ve aşırı sayıda bu paketlerden DNS sunucusuna gönderir. Böylece DNS sunucusu gelen aşırı paketler sonrası paket göndereni (kurbanı) kara listeye alır ve bundan böyle ondan gelen paketleri kabul etmez. Halbuki paketi gönderen saldırgandı, fakat saldırgan paketlerin kaynak IP'sini değiştirerek sanki kurbandan paketler gidiyormuş gibi yaptı. İşte bu saldırı yöntemine Rate Limiting denir.

UDP üzerinden gerçekleştirilecek DDoS saldırılarını engellemek zordur. Çünkü saldırıyı gerçekleştirenin IP adresinin gerçek olup olmadığını anlamanın kesin bir yolu yoktur. TCP üzerinden gerçekleştirilecek DDoS saldırılarını engellemek ise göreceli olarak UDP'ye göre daha kolaydır. Bunun temel nedeni TCP üzerinden yapılacak saldırılarda saldırganın gerçek IP adresle mi yoksa sahte IP adresle mi saldırıp saldırmadığının anlaşılabiliyor olmasıdır (3'lü el sıkışma sorunsuz tamamlanıyorsa IP gerçektir).

NOT: 3'lü el sıkışma sorunsuz tamamlanıyorsa IP gerçektir. Çünkü diyelim ki saldırgan kaynak IP'sini değiştirdiği bir paketi (SYN) sunucuya yolladı. Sunucu 3'lü el sıkışma için göndereceği SYN/ACK paketini değiştirilmiş kaynak IP'ye gönderecektir. Yani 3'lü el sıkışma paketi saldırgana değil, kurbana gidecektir. E haliyle kurban böyle bir bağlantı kurmaya teşebbüs etmediğinden dolayı herhangi bir yanıt paketi (ACK) göndermeyeceğinden 3'lü el sıkışma tamamlanamayacak ve DNS sunucusu ile kurban arasında bağlantı kurulmayacaktır. Dolayısıyla saldırgan kurbana DDoS yapamamış olacaktır.

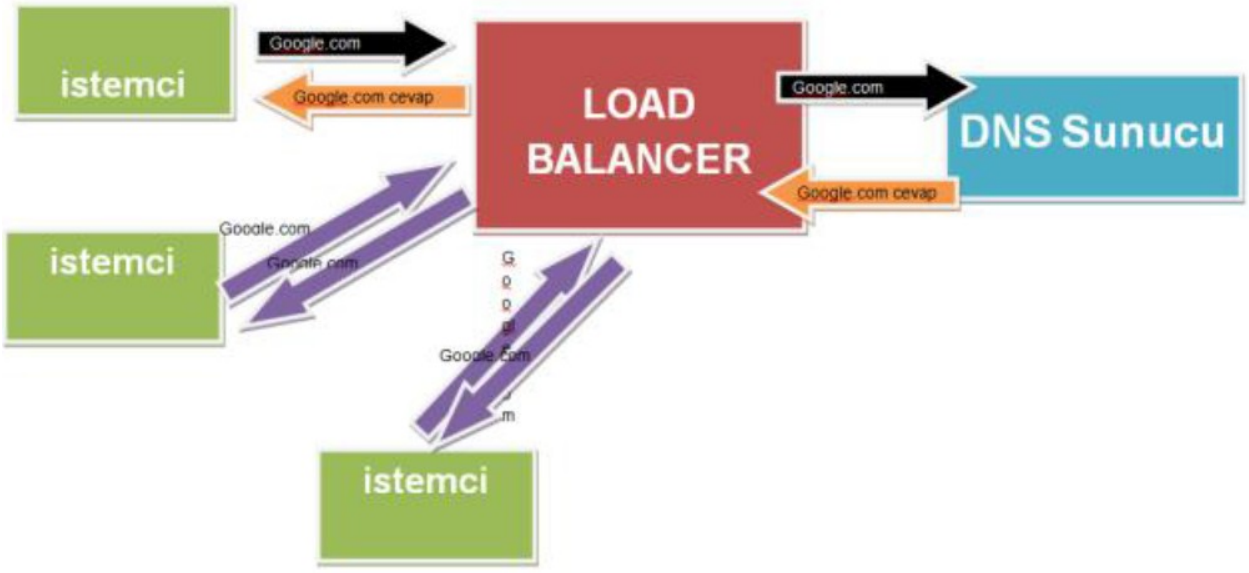
NOT2: 3’lü el sıkışma eğer tamamlanıyorsa client gerçekten talebi yapan kişi anlamına gelir. Çünkü sunucu bir paket (SYN) alıyor ve o paketin kaynak IP’sine bakarak bir paket (SYN/ACK) gönderiyor. Ardından sunucu bir paket (ACK) daha alıyor ve görüyor ki bu paketin kaynak IP’si ilk paketle aynı. Ayrıca aldığı son paketin TCP sıra numarası beklediği sıra numarası. Böylelikle sunucu emin oluyor. Burada akla şu soru gelebilir? Saldırgan kurbanın IP’siyle sunucuya SYN paketi yollasa ve bir süre sonra da tekrar kurbanın IP’siyle bir ACK paketi yollasa sunucu TCP 3’lü el sıkışmayı tamamlamaz mı? Cevap hayır. Çünkü saldırganın ACK paketinde sadece kurbanın IP’sini girmesi yetmiyor. Aynı zamanda saldırgan sunucunun alması gereken sıradaki TCP sıra numarasını da pakete koymalıdır, yani tahmin etmelidir ki bu çok zordur. Halbuki kurban SYN/ACK paketini sunucudan aldığı için sıra numarasını bilecektir ve bir sonraki numarayla sunucuya ACK paketini gönderebilecektir. Böylelikle sunucu da kaynaktan emin olacaktır.

DFAS

UDP kullanarak gerçekleştirilen saldırılarda genellikle davranışssal engelleme yöntemleri ve ilk paketi engelle ikinciyi kabul et gibi yöntemler kullanılır. Buna DFAS denir.

DNS Caching Cihazları

Caching cihazları aynı tipte gelen sorgulamalar için caching işlemi yapabilmektedir ve yoğun saldırılarda DNS sunucularının en az seviyede etkilenmesini sağlayabilmektedirler.


istemci
Google.com
Google.com cevap
LOAD
BALANCER
Google.com
DNS Sunucu
Google.com cevap
istemci
istemci

(Page 35-37)