KONPUTAGAILU SAREEN OINARRIAK

5. PRAKTIKA

Scapy pakete maneiagailua: Web bezero baten programazioa

TESTUINGURUA

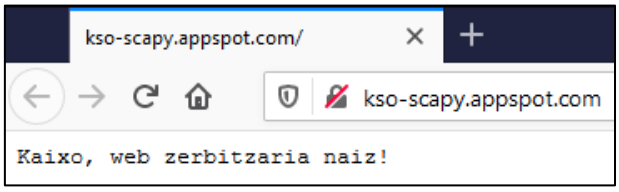
http://kso-scapy.appspot.com/ URI-a atzitzeko, nabigatzaileak web zerbitzariari ondorengo HTTP eskaera bidaltzen dio:

```
GET / HTTP/1.1
Host: kso-scapy.appspot.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Zerbitzariak bezeroari HTTP erantzuna bueltatzen dio, bere edukian testu lauz osotutako mezua dagoelarik.

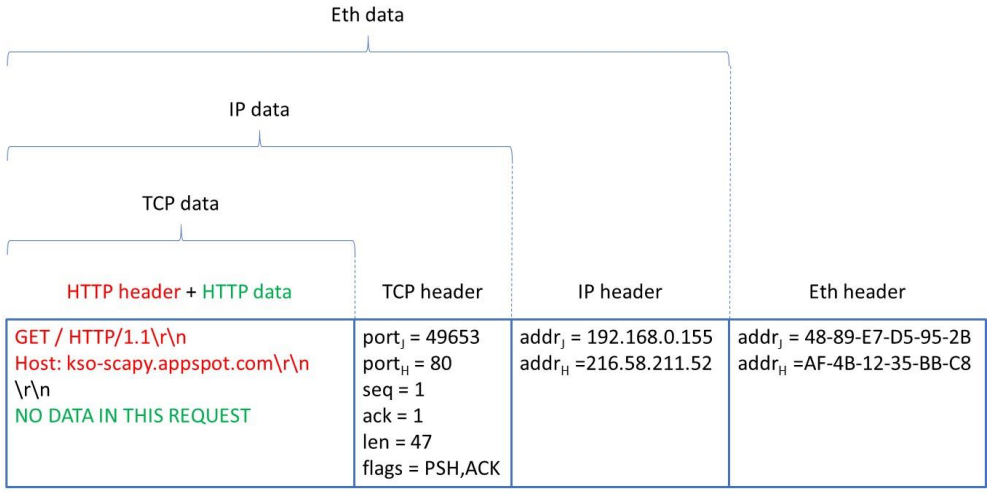
```
HTTP/1.1 200 OK
Cache-Control: no-cache
Content-Type: text/plain; charset=utf-8
Content-Encoding: gzip
X-Cloud-Trace-Context: c03a5f96352576373809 40cfb66ec8ac;o=1
Vary: Accept-Encoding
Date: Thu, 26 Mar 2020 16:16:30 GMT
Server: Google Frontend
Content-Length: 48

Kaixo, web zerbitzaria naiz!
```

Nabigatzaileak HTTP erantzunetik edukia atera eta bera bistaratzeko aukera al duen begiratzen du (hots, HTTP erantzunaren *Content-Type* goiburua irakurtzen du). Horrela bada, edukia pantailan bistaratzen du; horrela ez bada, edukia bistaratzeko plugin-ik al duen begiratzen du (adibidez, PDF-en kasuen). Nabigatzaileak edukia zuzenean bistaratu ezin badezake edo edukia bistaratzeko plugin-ik konfiguraturik ez badauka, orduan erabiltzaileari edukia fitxategi bezela gordetzeko aukera ematen dio.



HELBURUA

http://kso-scapy.appspot.com/ URI-a eskatzerakoan, nabigatzailearen funtzionamendua simulatzen duen web bezero bat programatu nahi da. Bezeroa eta zerbitzariaren arteko komunikazioa bilbe-mailakoa izan dadila nahi da, bilbeak goiburuz-goiburu eraiki behar direlarik. Horretarako, Scapy pakete maneiagalu liburutegia erabiliko dugu.



Zelan eraikiko litzateke bilbe hau Scapy erabiliz?

```
# HTTP eskaera: HTTP goiburua + HTTP datuak
http_req = "GET / HTTP/1.1\r\n" \
           + "Host: kso-scapy.appspot.com\r\n\r\n"

# HTTP eskaera: TCP goiburua
psh_ack= TCP(sport=49653, dport=80, flags="PA", seq=1, ack=1)

# HTTP eskaera: IP goiburua
ip = IP(dst="kso-scapy.appspot.com")

# HTTP eskaera: bilbea bidali eta berrespena jaso
ACK = srp1(Ether()/ip/psh_ack/http_req)
```

OHARRAK

- TCP konexioaren abiaraztea (SYN paketea) Scapy-k egingo du. SYN,ACK paketea bueltan datorrenean, sistema eragileak ez du berak hasitako konexio baten erantzun bezela ikusiko eta zerbitzariari RST pakete bat bidaliko dio konexioa atzera botaz.

  Hori gertatu ez daiten, irteerako RST paketeak suebakiaren bitartez baztertuko ditugu:

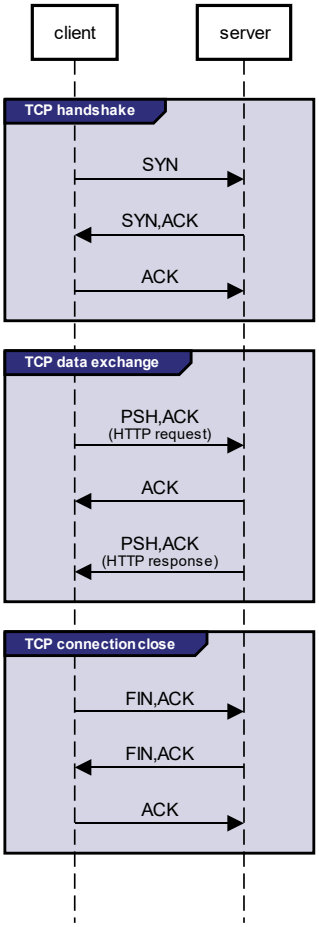
  sudo iptables -A OUTPUT -p tcp --tcp-flags RST RST -s IP_addr -j DROP

- *Wireshark* erabili beharrean, **tcpdump** protokolo aztertzailea erabiliko dugu

sudo tcpdump -i ens33 host kso-scapy.appspot.com and port 80

- Inplementatu beharreko TCP fluxua:

No.	Time	Source	Destination	Protocol	Length	Info
12	10.239095669	192.168.5.21	172.217.17.20	TCP	54	62854 → 80 [SYN] Seq=0 Win=8192 Len=0
13	10.252210953	172.217.17.20	192.168.5.21	TCP	60	80 → 62854 [SYN, ACK] Seq=0 Ack=1 Win=60720 Len=0
14	10.269680408	192.168.5.21	172.217.17.20	TCP	54	62854 → 80 [ACK] Seq=1 Ack=1 Win=8192 Len=0
15	10.276497254	192.168.5.21	172.217.17.20	HTTP	101	GET / HTTP/1.1
16	10.288958766	172.217.17.20	192.168.5.21	TCP	60	80 → 62854 [ACK] Seq=1 Ack=48 Win=60720 Len=0
17	10.377672447	172.217.17.20	192.168.5.21	HTTP	310	HTTP/1.1 200 OK (text/plain)
18	10.380837615	192.168.5.21	172.217.17.20	TCP	54	62854 → 80 [FIN, ACK] Seq=48 Ack=257 Win=8192 Len=
19	10.396525906	172.217.17.20	192.168.5.21	TCP	60	80 → 62854 [FIN, ACK] Seq=257 Ack=49 Win=60720 Ler
20	10.499230213	192.168.5.21	172.217.17.20	TCP	54	62854 → 80 [ACK] Seq=49 Ack=258 Win=8192 Len=0



- TCP flag bakoitzak daukan balioa:

CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
128	64	32	16	8	4	2	1

Flag bat baino gehiago aktibatuta badago, beraien balioak batzen dira; adibidez: FA=17