Ön Söz

Bu belge bilgisayar ağlarında adli bilişim çalışmalarını (network forensic), kullanılan teknik ve terimleri anlatmaktadır.

Pratik olarak packet-o-matic network forensic aracı ile network forensic çalışmasının nasıl yapılacağını içermektedir.

Belge, **Ozan UÇAR** tarafından yazılmıştır ve yazarın adına sadık kalmak şartı ile paylaşılabilir.

mail@ozanucar.com

Network Forensic

Bilgisayar ağlarını dinleyerek, iletişim kanallarından orjinal verilerin (eposta, msn yazışmaları, ofis dökümanları, ses vb.) bir kopyasını elde işlemi "network forensic" olarak isimlendirilir.

IP telefonlar, anlık iletişim araçları, eposta ve web servisleri işimizi/hayatımızı kolaylaştıran ve hızlandıran vazgeçilmezler arasında.
Şirket verilerini , canlı para işlemlerini ve özel/genel tüm yazışmaları bilgisayar ağlarını kullanarak yapmaktayız.

Bu denli yoğun kullandığımız bilgisayar ağları ne kadar güvenli ? Hiiiiiiç düşündüğünüz gibi değil, bir önlem alınmadıysa tamamen güvensiz. İçeriği okunabilir (clear text), güvenli iletişim kanallarını kullanmayan tüm bağlantılar izlenebilir ve kaydedilebilir. Bu işlem için paket yakalamak ve daha sonra ayrıştırmak gerekir.

Paket Dinleme/Yakalama

Bilgisayar ağlarında iletişim protokoller üzerinden aktarılan paketler ile gerçekleşir.Tıpkı günlük yaşantımızda cümlelerimizi oluşturan sözcükler gibi, paketler'de ağ trafiğini oluşturur.

Paket Ayrıştırma

Sesli iletişimimiz bu konu içinde somut bir örnek, sesli iletişimimizi dinleyen/kaydeden biri dilimizi biliyorsa tüm konuşulanlara vakıf olabilir.

Bilgisayar ağlarında elde edilen paketler bir araya getirilerek, orjinal verilen, oturum bilgileri elde etme methodudur.

Promiscious Mod

İşletim sistemleri hedefi kendi olmayan paketleri layer 3 de reddeder.
Promisc. mod, bir makinenin hedefi kendisi olmayan paketleri alabilmesini sağlar.Tüm snifferlar aksi belirtilmediği taktirde otomatik olarak ağ arabirimini promiscious moda geçirir.

ifconfig em0

em0: flags=8843<UP,BROADCAST,RUNNING, T,**PROMISC**,SIMPLEX,MULTICAST> metric 0 mtu 1500

options=9b<RXCSUM,TXCSUMT,PROMISC,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM>

Packet-O-Matic Nedir ?

packet-o-matic, gerçek zamanlı paket ayrıştırıcısıdır. POM, **libpcap** paket yakalama kütüphanesini kullanır ve ağ arabirimine gelen/geçen paketleri kaydedip, ayrıştırma işlevine sahiptir.

Modüler bir yapıya sahiptir, geçerli modüller;

input	docsis, pcap
match	80211, docsis, ethernet, icmp, icmpv6, ipv4, ipv6, linux_cooked, ppi, prism, radiotap, rtp, tcp, udp, vlan
conntrack	ipv4, ipv6, rtp, tcp, udp
helper	docsis, ipv4, ipv6, tcp, rtp
target	display, dump_payload, http, inject, irc, null, msn, pcap, pop, rtp tap, tcpkill

Komut Referansları

config write dosya_adı	Geçerli ayarları, belirtilen dosya ismi ile kaydeder.
debug cli set <off,0-5>	Yönetim arayüzü için debug seviyesini belirler.
debug cli show	Geçerli olan debug seviyesini gösterir.
Exit	Konsoldan çıkar.
halt	Programı durdurur.
?	Kullanılabilir tüm komutları listeler, yardım almak için kullanılır.
password cli set <password>	Bağlantı kurulacak telnet oturumuna parola tanımlar.
version show	Packet-o-matic versiyonunu gösterir.

Pratik ve Gelişmiş Bazı Örnekler;

ipv6 tcp bağlantılarını belirtmek için

ipv6 | tcp

Tüm tcp trafiğini dinlemek istiyoruz

ipv4

ipv4 ve hedef portu 80 olan trafiği takip etmek için

ipv4 | tcp.dport == 80

Tüm ipv4 tcp bağlantılarını yakala, kaynak portu 80 olanları hariç tut

ipv4 | !tcp.sport == 80

Biraz komplike bir kural, ipv4 tcp kaynak port 80 ve 443 arası trafiğini istiyorum, port 110 ı hariç tut

ipv4 | (tcp.sport >= 80 and tcp.sport <= 443) and !tcp.sport == 110

Sık kullanılan bazı örnekler

HTTP Trafiği

tcp.dport == 80

MSN Trafiği

tcp.dport == 1863

1.1.1.1 ip adresine ait RTP (VOIP) trafiği

ipv4.dst == 1.1.1.1 | udp | rtp

Kurulum;

```
#apt-get install libxml2-dev libxmlrpc-c3-dev libpcap-dev
#http://packet-o-matic.org/downloads/?C=M;O=D
wget http://www.packet-o-matic.org/downloads/packet-o-matic-svn-20100621.tar.gz
#tar zxvf packet-o-matic-svn-20100621.tar.gz
#cd packet-o-matic-svn
#./configure
#make
#make install
#export LD_LIBRARY_PATH=src/.libs
```

Not: Kurulum adımları ubuntu için geçerlidir, paket bağımlıkları ve kaynak kod temin edilerek benzer adımlarla tüm linux ve bsd sistemlere kurulum yapılabilinir.

#packet-o-matic -h

```
Usage : packet-o-matic [options]

Options :

 -c, --config=FILE        specify configuration file to use (default pom.xml.conf)

 -b, --background         run in the background as a daemon

 -h, --help               display the help

    --no-cli              disable the CLI console
```

-p, --port=PORT specify the CLI console port (default 4655)

-w, --password=PASS specify a password to enter the CLI console

-d, --debug-level=LEVEL specify the debug level for the console <0-5> (default 3)

-X --enable-xmlrpc enable the XML-RPC interface

-P, --xmlrpc-port=PORT specify the XML-RPC port (default 8080)

-W, --xmlrpc-password=PASS specify the password for XML-RPC calls

--pid-file specify the file where to write the PID

.....

Packet-o-matic iki farklı yönetim arabirimine sahiptir, konsol ve web arabirimi. Packet-o-matic'in esnekliğinden ve gerçek gücünden faydalanmak için konsol arabirimini öneririm.

Web Arabirimine bağlanmak;

Web arabirimi ile başlatmak için **-X** paremetresini kullanabiliriz.Genel hatları konusunda görsellik sağlayacaktır.

 # packet-o-matic -X

mgmtsrv: Management console listening on 0.0.0.0:4655

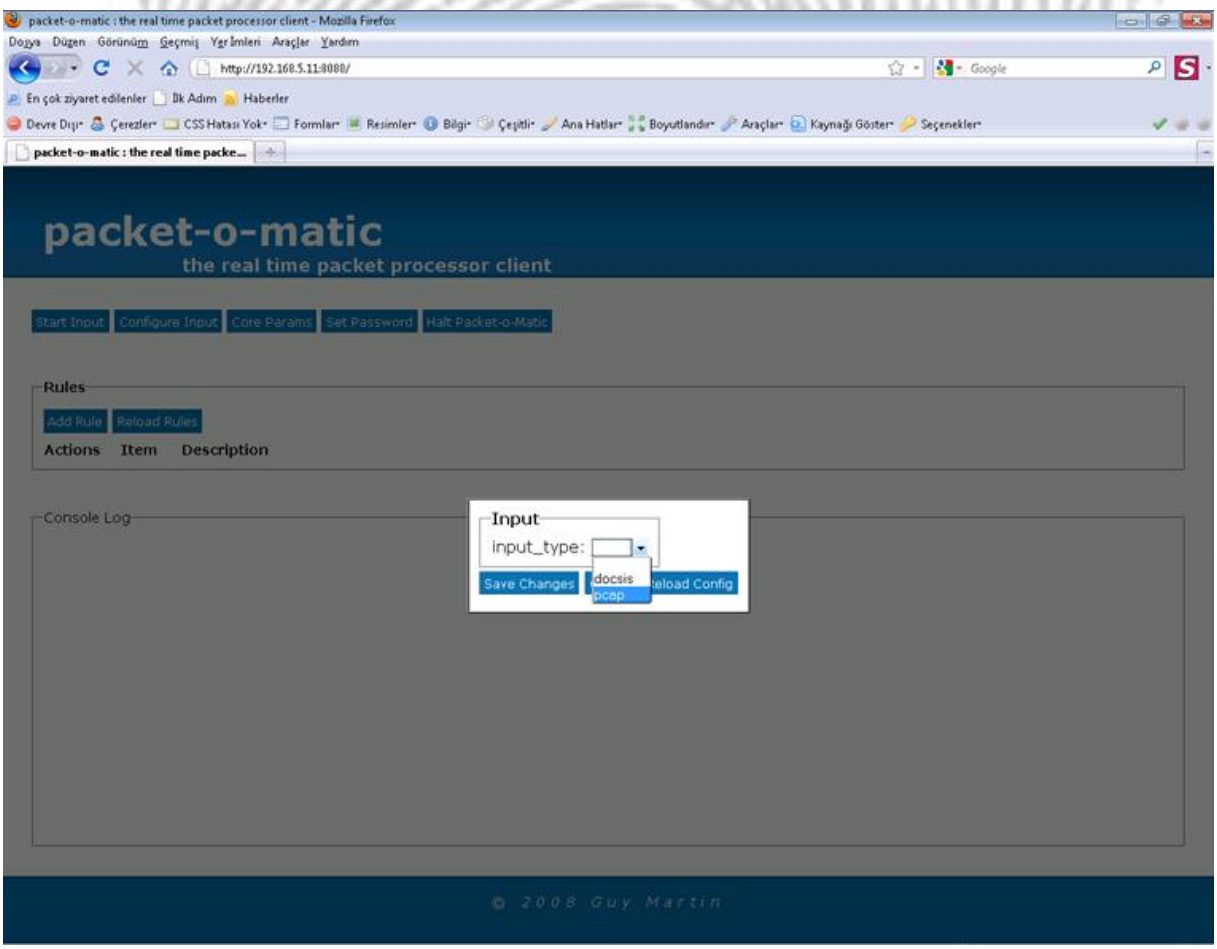
xmlrpcsrv: XML-RPC server listening on 0.0.0.0:8080

main: Could not open config file pom.xml.conf : No such file or directory

main: Starting with and empty configuration

main: packet-o-matic dist-20100621 started

Web arabirimi erişim adresi **http://ipadresi:8080**



Konsol arayüzünü çalıştırmak;

 # packet-o-matic

komut satırından yönetim için, telnet ile **port 4655** bağlantı kurarak, **pom** satırına düşüyoruz.

 # telnet 127.0.0.1 4655

```
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
This is packet-o-matic dist-20100621
Copyright Guy Martin 2006-2010
Type '?' for command list.
pom> ?
```

Yardım almak için "**?**" yazarak, kullanılabilir komutları listeleyebilirsiniz.

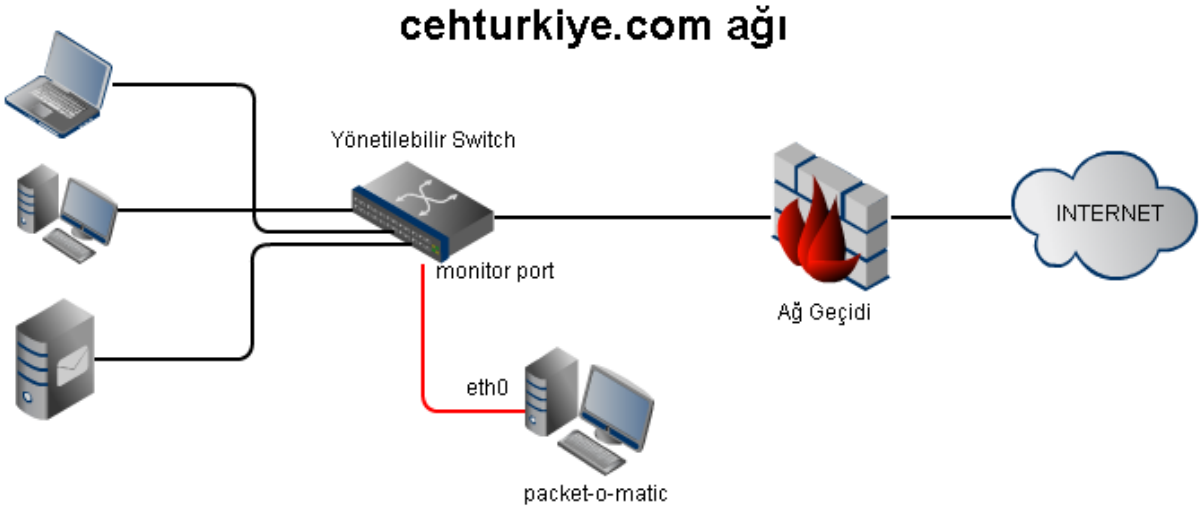
Başlangıç

packet-o-matic çalıştırmak için, aşağıdaki ayarları işlem sırasına göre yapılandırmak durumundasınız;

- Bir giriş türü seçin ve onu yapılandırın
- İhtiyacınız olan kuralları ekleyin
- İhtiyacınız olan hedefleri ekleyin
- Yapılandırmanızı kaydedin

Örnek Uygulamalar

Fabrikam.com ağındaki kullanıcıların internette gezinirken indirdikleri verilerin (resim,döküman,binary dosyalar, text metinler vs.) bir kopyasını almak istiyorum. Aynı zamanda şirket ağımdan dışarı web üzerinden gönderilen dosyalarında (sözleşme belgeleri gibi gibi) bir kopyasını yedeklemek istiyorum (Biri veri kaçırıyor olabilir mi ?)



Switch üzerindeki monitor port'dan pasif modda dinleme yapıyoruz, network'de araya girmeden tüm trafiği bu şekilde izleyebiliriz.
"**eth0**" ağ arabirimini *promisc. mod*'a geçirip, HTTP trafiğinden tüm resimler, dökümanlar ve binary dosyaları (upload&download edilenleri) almak isteyelim;

Birinci adımda, giriş türü **"input pcap"** olarak atanmalı, varsayılan olarak **eth0** ağ arabirimini dinlemeye alır.

pom> *input type set pcap*

promiscuos modu aktif edelim

pom> *input parameter set promisc yes*

pom> *input show*

Current input : pcap, mode interface (0 packets, 0 bytes, up 00:00.00)

interface = eth0

snaplen = 1522 bytes

promisc = yes

filter =

İkinci adımda kuralımızı yazmamız gerekiyor.Biz *hedef portu 80* olan http trafiğini izlemek istiyoruz.Kuralımız ;

pom> *rule add tcp.dport == 80*

Added rule with id 0

pom> *rule enable 0*

pom> *rule show*

Rule 0 (0 packets, 0 bytes, up 00:02.02) :

tcp.dport == 80

Üçüncü adımda hedeflerimiz yer alıyor. HTTP trafiğinden **resimler, dökümanlar ve binary** dosyaları elde etmek. Bu işlem için **"target_http"** modülü tamda bize göre.

pom> *target add 0 http*

Added target with id 0 to rule 0

Bu modülü kullandığımızda, yakalanan verilen /tmp altına kaydedilecektir.

```
pom> target parameter set 0 0 dump_img yes

pom> target parameter set 0 0 dump_bin yes

pom> target parameter set 0 0 dump_doc yes

pom> target parameter set 0 0 ds_log_format %a %f %D

pom> target show

Rule 0 : targets (0 packets, 0 bytes, up 07:45.51) :

  0) http, mode default (0 packets, 0 bytes, up 00:00.00) (stopped)

      prefix = /tmp/

      decompress = yes

      mime_types_db = /usr/local/share/packet-o-matic/mime_types.db

      log_file =

      log_format = %v %a %u %t "%r" %s %b

      ds_log_path =

      ds_log_format = %a %f %D

      dump_img = yes

      dump_vid = no

      dump_snd = no

      dump_txt = no

      dump_bin = yes

      dump_doc = yes
```

Ve kuralı işleme koyuyoruz

pom> *target start 0 0*

pom> *input start*

Dördüncü ve son adımımız da ayarlarımızı kaydetmeyi unutmayalım =))

pom> *config write*
Configuration written in ***pom.xml.conf***

Trafiği bir süre dinledikden sonra, bakalım ***/tmp*** altında, bizi bekleyen neler var ?

Binary dosyaları listeleyelim;

tmp# ls *.bin

20101027-154335-390507.bin

PDF dosyalar;

tmp# ls *.pdf

20101027-154045-643459.pdf

Resim dosyaları;

tmp# ls *.jpg

20101027-154014-12330.jpg 20101027-154042-222028.jpg 20101027-154331-972440.jpg 20101027-154333-293015.jpg

20101027-154014-489799.jpg 20101027-154042-287532.jpg 20101027-154332-269582.jpg 20101027-154335-49309.jpg

20101027-154037-51617.jpg 20101027-154042-290989.jpg 20101027-154332-91725.jpg

Ve daha bir çok şey ....

Dosyalar, tarih saat ismi ile kaydedilmiş durumda, log formatını değiştirmek isterseniz bakınız ;
http://wiki.packet-o-matic.org/target_http

Diğer bir örnek

MSN Konuşmalarını, avatar resimlerini ve msn'den yapılan dosya transferlerini kaydetmek istiyorum.

Firmamda, msn görüşmeleri yapan çalışanların benden izinsiz gönderdikleri ve aldıkları bilgiler ile yazışmalarınıda kaydetmek istiyorum, istemediğim kullanıcılarda msn açamasın engelleyebilir miyim ?

Namümkünü mümkün kılmak mümkündür ....

pom> input show

Current input : pcap, mode interface (28312 packets, 10M bytes, up 11:08.45) (running)

interface = eth0

snaplen = 1522 bytes

promisc = yes

filter =

pom> rule add tcp.dport == 1863

Added rule with id 0

pom> rule enable 0

pom> rule show

Rule 0 (0 packets, 0 bytes, up 00:29.23) :

tcp.dport == 1863

pom> target add 0 msn

Added target with id 0 to rule 0

MSN dosya transfelerini yakalayan hedefi aktif edelim,

pom> target parameter set 0 0 dump_file_transfer yes

Yakaladığı verileri /tmp/msnlive klasoru altına kaydetsin.

pom> target parameter set 0 0 path /tmp/msnlive

pom> target show

Rule 0 : targets (0 packets, 0 bytes, up 04:12.19) :

0) msn, mode dump (0 packets, 0 bytes, up 00:00.00) (stopped)

path = /tmp/msnlive

dump_session = yes

dump_avatar = yes

dump_file_transfer = yes

pom> target start 0 0
pom> input start

Eğer birileri yerel ağda msn kullanıyorsa, yazışmaları ve transfer edilen dosyaların bir kopyasını alacak,

İlgili dizinimize **/tmp/msnlive** bakalım;

/tmp/msnlive# ls -l

total 8
drwxr-xr-x 2 root root 4096 2010-11-12 12:55 **dali_sen@hotmail.com**
drwxr-xr-x 3 root root 4096 2010-11-12 13:03 **mail@ozanucar.com**

İki kişi msn kullanıyormuş, "**mail@ozanucar.com**" adresine ait klasore bakalım**;**

 # ls -l mail\@ozanucar.com/

total 12
-rw-r--r-- 1 root root 422 2010-11-12 13:03 ***dali_sen@hotmail.com-20101112-13.txt***
drwxr-xr-x 2 root root 4096 2010-11-12 13:03 ***files***
-rw-r--r-- 1 root root 3028 2010-11-12 13:02 ***mail@ozanucar.com-display-picture.png***

Transfer edilen dosyalar "**files**" klasorü altında tutuluyor,

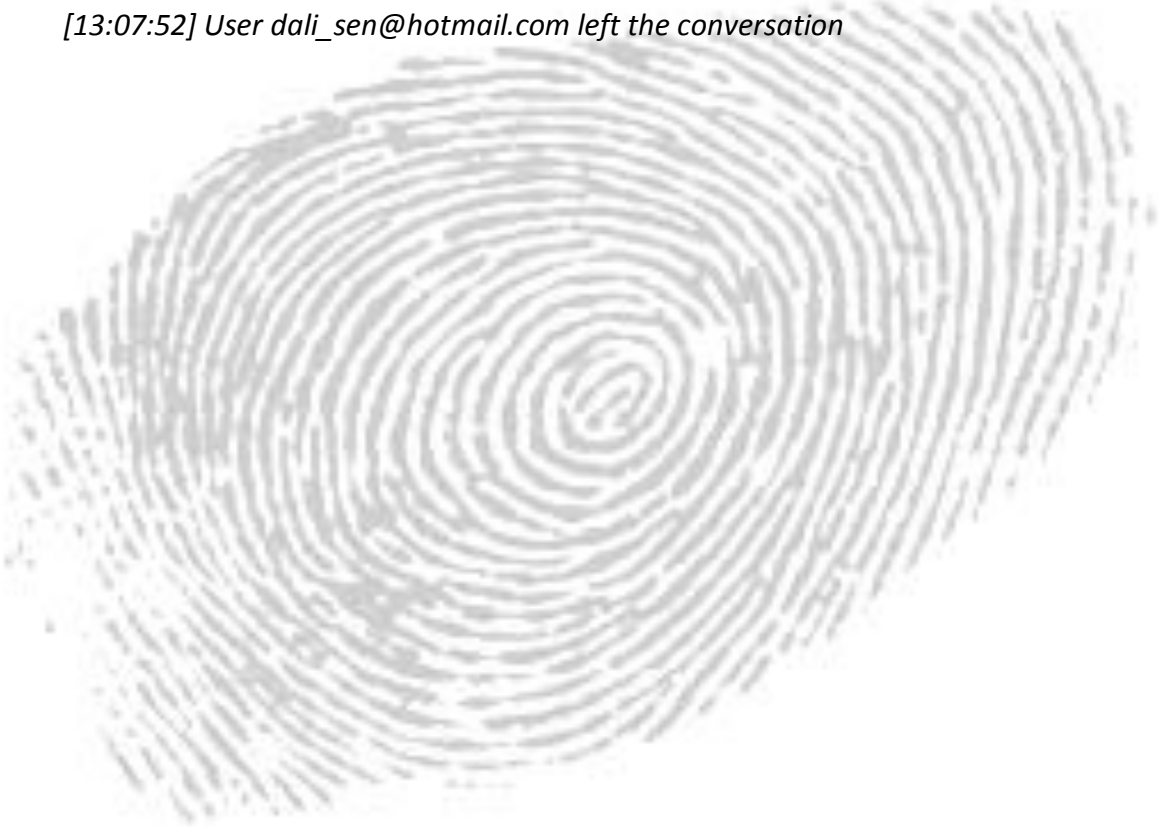
ls mail\@ozanucar.com/files/

20101112-130344-etkinlestir.php

Yazışmalar, ilgili kişinin adı ile **.txt** dosyası halinde saklanıyor,

 # cat dali_sen\@hotmail.com-20101112-13.txt

[13:02:03] User dali_sen@hotmail.com joined the conversation
[13:02:17] File transfer started with user dali_sen@hotmail.com
[13:02:18] File transfer ended with user dali_sen@hotmail.com
[13:02:26] mail@ozanucar.com: kodu çözdüm, zend ile şifrelenmiş
[13:02:52] mail@ozanucar.com: dezend adında bir zımbırtı var
[13:03:44] File transfer started with user dali_sen@hotmail.com : "etkinlestir.php"
[13:03:45] File transfer ended with user dali_sen@hotmail.com : "etkinlestir.php"
[13:07:52] User dali_sen@hotmail.com left the conversation



Engellemek istesem ?

Şirketimde 192.168.5.233 ip adresli makina "**msn açamasın**" istiyorum.target modülü olarak "**tcpkill**" seçilerek kaynak ve hedefe **tcp rst** paketleri gönderilerek msn oturumu engellenebilir.

pom> *input show*

Current input : pcap, mode interface (7593 packets, 9025K bytes, up 00:03.08) (running)

interface = eth0

snaplen = 1522 bytes

promisc = yes

filter =

pom> *rule add ipv4.src == 192.168.5.233 | tcp.dport == 1863*

Added rule with id 0

pom> *rule enable 0*

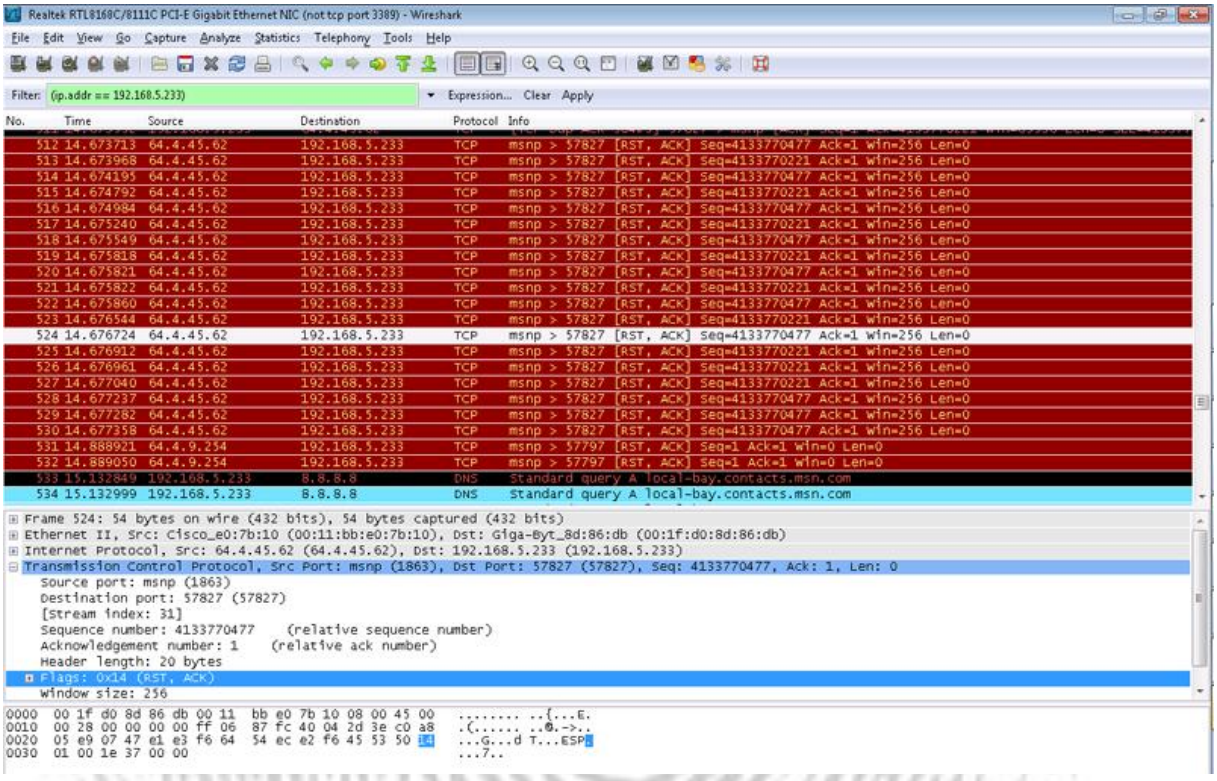
pom> *target add 0 tcpkill*

Added target with id 0 to rule 0

pom> *target start 0 0*

Tcpkill ile sonlandırılan oturuma ait trafik bilgisi,

Wireshark Çıktısı;



tcpdump Çıktısı;

tcpdump -nn -ttt -i eth0 host 192.168.5.233 and port 1863

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode

listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes

00:00:00.000000 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [S], seq 2737163465, win 8192, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0

00:00:00.000042 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [S], seq 2737163465, win 8192, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0

00:00:00.000007 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [S], seq 2737163465, win 8192, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0

00:00:00.000987 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 0, ack 2737163466, win 8192, length 0

00:00:00.000309 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 8192, ack 1, win 8192, length 0

00:00:00.000187 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 0, ack 1, win 8192, length 0

00:00:00.000154 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 8192, ack 1, win 8192, length 0

00:00:00.000154 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 0, ack 1, win 8192, length 0

00:00:00.000123 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 0, ack 1, win 8192, length 0

00:00:00.000076 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 0, ack 1, win 8192, length 0

00:00:00.000117 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 8192, ack 1, win 8192, length 0

00:00:00.000106 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 8192, ack 1, win 8192, length 0

00:00:00.000073 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 8192, ack 1, win 8192, length 0

00:00:00.286329 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [S.], seq 2958369682, ack 2737163466, win 16384, options [mss 1460,nop,wscale 0,nop,nop,sackOK], length 0

00:00:00.000035 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, length 0

00:00:00.000006 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [S.], seq 2958369682, ack 2737163466, win 16384, options [mss 1460,nop,wscale 0,nop,nop,sackOK], length 0

00:00:00.000003 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, length 0

00:00:00.000003 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, options [nop,nop,sack 1 {0:1}], length 0

00:00:00.000003 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, length 0

00:00:00.000003 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, options [nop,nop,sack 1 {0:1}], length 0

00:00:00.000342 IP 192.168.5.233.58401 > 64.4.45.62.1863: Flags [.], ack 1, win 256, options [nop,nop,sack 1 {0:1}], length 0

00:00:00.000396 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000293 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000178 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000179 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000203 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000162 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000175 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000202 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000006 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000998 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000106 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000249 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000010 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000097 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000066 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000055 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000119 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000006 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0

00:00:00.000113 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 1, ack 1, win 256, length 0

00:00:00.000005 IP 64.4.45.62.1863 > 192.168.5.233.58401: Flags [R.], seq 257, ack 1, win 256, length 0