Sending IPv6 packets to check firewall rules

Introduction

IP version 6 is available in most recent products such as routers, firewalls and operating systems.

Administrators and security professionals are faced to new challenges when configuring or checking an IPv6 implementation. They need IPv6 compatible tools.

Network testing requires two simple components : a tool to send network packets, and a sniffer to intercept and display them.

Most sniffers already recognize IPv6 packets (Ethereal, tcpdump, etc.).

This article describes netwox toolbox which can be used to send IPv6 packets.

Learning

It is important to note that computer sending IPv6 packets must use an Ethernet LAN, or already be IPv6 compatible. As the latter is less common, we send Ethernet frames containing IPv6 packets. Tools must be ran as root (Administrator under Windows) to have privileges to send Ethernet frames. Finally, we suppose both computers are on the same LAN (do not have routers between them).

The first example is a TCP packet over IPv6 over Ethernet. Install netwox or netwag and run as root (text wrapped):

```
# netwox 142 --device "Eth0" --eth-dst "0:8:9:a:b:c"
--ip6-src "fec0:0:0:1::1" --ip6-dst "fec0:0:0:1::2" --tcp-src
"1234" --tcp-dst "80" --tcp-syn
```

Following packet is sent:

```
Ethernet_________________________________________________________.
|.00:11:22:33:44:55->00:08:09:0A:0B:0C.type:0x86DD...............|
|________________________________________________________________|
IP_______________________________________________________________.
|version|.traffic.class.|..............flow.label...............|
|___6___|_______0_______|___________________0___________________|
|........payload.length.........|..next.header..|...hop.limit...|
|___________0x0014=20___________|____0x06=6_____|_______0_______|
|............................source..............................|
|_________________________fec0:0:0:1::1__________________________|
|..........................destination...........................|
|_________________________fec0:0:0:1::2__________________________|
TCP______________________________________________________________.
|..........source.port..........|.......destination.port.........|
|__________0x04D2=1234__________|___________0x0050=80____________|
|............................seqnum..............................|
|_____________________0x686F31E7=1752117735______________________|
|............................acknum..............................|
|_________________________0x00000000=0___________________________|
|.doff..|r|r|r|r|C|E|U|A|P|R|S|F|............window..............|
|___5___|0|0|0|0|0|0|0|0|0|0|1|0|___________0x0000=0_____________|
|...........checksum............|............urgptr..............|
|__________0x12E4=4836__________|___________0x0000=0_____________|
```

Ethernet and IP header indicates that destination Ethernet address is 0:8:9:a:b:c, source IPv6 address is fec0:0:0:1::1 and destination IPv6 address is fec0:0:0:1::2.

To learn about possible parameters for tool number 142, run:

```
# netwox 142 --help
# netwox 142 --help2
```

Real world example

Suppose we want to check if a host has its firewall correctly configured to block some IPv6 packets destined to itself. Its IPv6 address is fec0:0:0:1::2. Its Ethernet address is 0:8:9:a:b:c (obtained with "netwox 3 fec0:0:0:1::2"). Suppose port 80/tcp is allowed for computer fec0:0:0:1::1, but all other ports and computers are blocked.

We simulate computer fec0:0:0:1::1 using another computer on the LAN. This computer does not need to be IPv6 compatible because we directly send IPv6 packet without using computer's IP stack. This computer has Ethernet address 00:11:22:33:44:55 (can be real or random). All command listed below are to be run on this computer.

First, we send a TCP SYN packet destined to port 80 of firewall. It is accepted because port 80 is open, so server sends back a TCP SYN-ACK packet. In order to send this SYN-ACK, server first asks for client Ethernet address using ICMP6 neighbor solicitation (IPv4 uses ARP). So we need 2 more tools: one to answer to Ethernet requests, and the other to see the SYN-ACK.

Netwox contains one tool to simulate the presence of a computer. This tool automatically answers to Ethernet requests. Open another window and keep running:

```
# netwox 73 --device "Eth0" --ips "fec0:0:0:1::1" --eths
"00:11:22:33:44:55"
```

This command answers "computer fec0:0:0:1::1 has Ethernet address 00:11:22:33:44:55" to every question.

Then open another window and run a sniffer (netwox in this example, but it can be Ethereal):

```
# netwox 7 -p --device "Eth0"
```

Send the IPv6 packet destined to port 80 and see what happens in the sniffer window (don't forget to change source port "--tcp-src" for each call, for example incrementing it):

```
# netwox 142 --device "Eth0" --eth-src "00:11:22:33:44:55" --eth-dst
"0:8:9:a:b:c" --ip6-src "fec0:0:0:1::1" --ip6-dst "fec0:0:0:1::2"
--tcp-src "1235" --tcp-dst "80" --tcp-syn
```

If port 80 is open, the sniffer will display a SYN-ACK. Here is an extract of a TCP header containing flags Ack and Syn set to 1:

```
|.doff..|r|r|r|r|C|E|U|A|P|R|S|F|............window.............|
|___5___|0|0|0|0|0|0|0|1|0|0|1|0|__________0x1680=5760__________|
```

Meaning of receiving a SYN-ACK packet is "port 80 is open, and you are allowed to connect".

Send an IPv6 packet destined to port 81 ("--tcp-dst 81"). Depending on firewall configuration, we receive a RST (flag R set in the TCP header) or nothing, and firewall's log contains an alert message. If a SYN-ACK is received, then firewall is badly configured because port 81 is open and available.

Now, we can pick another client address such as fec0:0:0:1::3 and check everything is forbidden.

Other tools

Tools 140 to 147 of netwox send UDP, ICMP or raw IPv6 packets. Depending on firewall rule to check, they can also be used.

```
# netwox 141 --device "Eth0" --eth-src "00:11:22:33:44:55" --eth-dst
"0:8:9:a:b:c" --ip6-src "fec0:0:0:1::1" --ip6-dst "fec0:0:0:1::2"
--udp-src "1236" --udp-dst "80"
Ethernet________________________________________________________.
|.00:11:22:33:44:55->00:08:09:0A:0B:0C.type:0x86DD..............|
|_______________________________________________________________|
IP______________________________________________________________.
|version|.traffic.class.|..............flow.label...............|
|___6___|_______0_______|___________________0___________________|
|........payload.length.........|..next.header..|...hop.limit...|
|___________0x0008=8____________|____0x11=17____|_______0_______|
|............................source.............................|
|_________________________fec0:0:0:1::1_________________________|
|..........................destination..........................|
|_________________________fec0:0:0:1::2_________________________|
UDP_____________________________________________________________.
|..........source.port..........|.......destination.port........|
|__________0x04D4=1236__________|___________0x0050=80___________|
|............length.............|...........checksum............|
|___________0x0008=8____________|_________0xFD33=64819__________|
```

```
 # netwox 143 --device "Eth0" --eth-src "00:11:22:33:44:55" --eth-dst
"0:8:9:a:b:c" --ip6-src "fec0:0:0:1::1" --ip6-dst "fec0:0:0:1::2"
--icmp-type "128" --icmp-code "0"
Ethernet________________________________________________________.
|.00:11:22:33:44:55->00:08:09:0A:0B:0C.type:0x86DD..............|
|_______________________________________________________________|
IP______________________________________________________________.
|version|.traffic.class.|..............flow.label...............|
|___6___|_______0_______|___________________0___________________|
|........payload.length.........|..next.header..|...hop.limit...|
|___________0x0008=8____________|____0x3A=58____|_______0_______|
|............................source.............................|
|_________________________fec0:0:0:1::1_________________________|
|..........................destination..........................|
|_________________________fec0:0:0:1::2_________________________|
ICMP6_echo.request______________________________________________.
|.....type......|.....code......|...........checksum............|
|___0x80=128____|____0x00=0_____|__________0x065B=1627__________|
|..............id...............|............seqnum.............|
|_________0xCD94=52628__________|_________0xAE46=44614__________|
|.data:.........................................................|
|_______________________________________________________________|
```

Conclusion

Ability to send an IPv6 packet is an elementary step for solving network problems or checking configurations. Netwox contains tools to achieve this step. Netwox also provides clients and servers supporting IPv6 : FTP client, web client, etc.

Download

Netwox comes with netwag, a graphical front-end, which is easier to use than command line tools. It depends on libpcap, libnet and netwib libraries.

You can download netwox at :

http://www.laurentconstantin.com/en/netw/#download
http://go.to/laurentconstantin/
http://laurentconstantin.est-la.com/