



Check Point[®]
SOFTWARE TECHNOLOGIES LTD.

Workshop in Information Security

Building a Firewall within the Linux Kernel

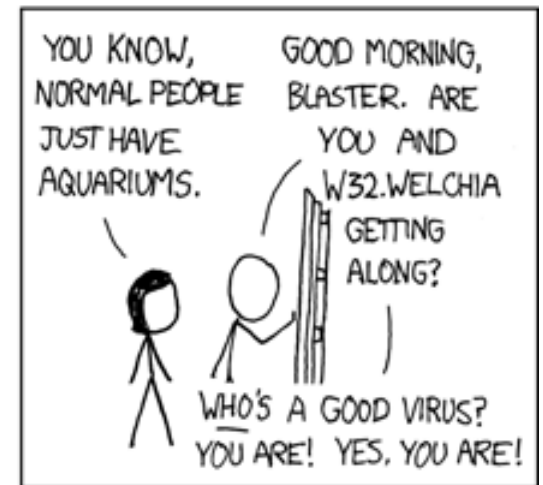
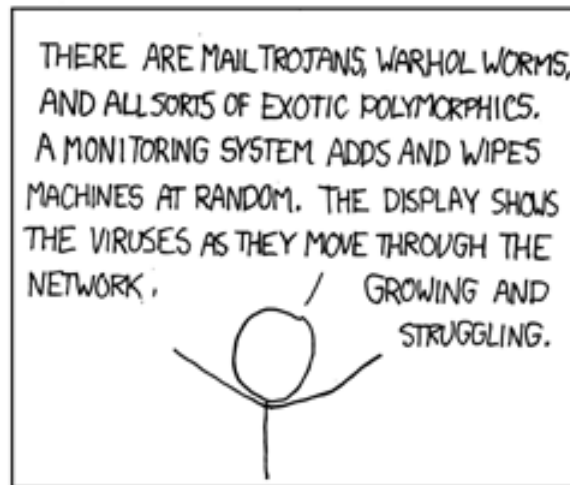
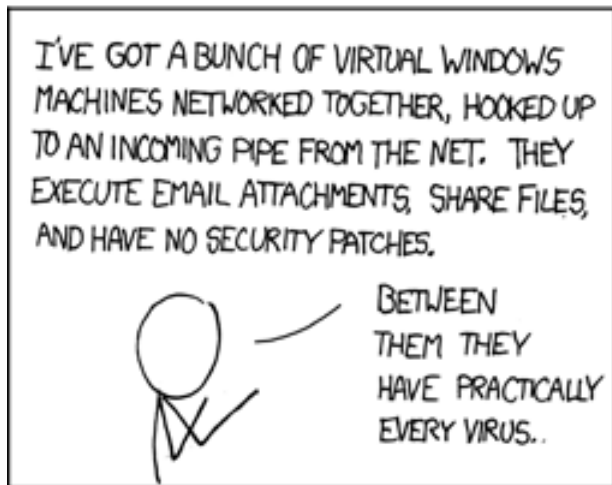
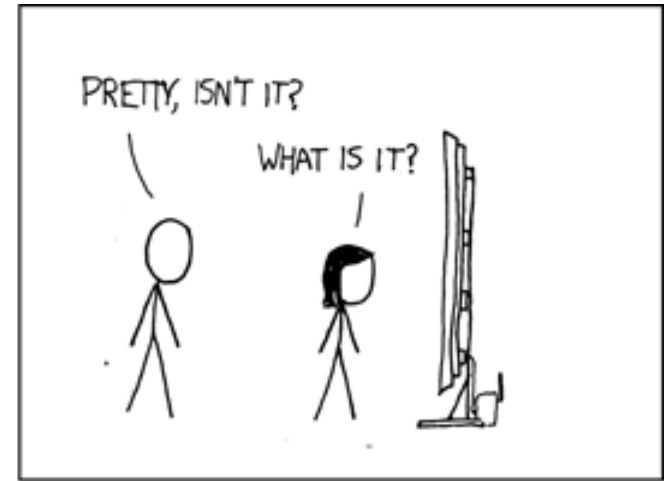
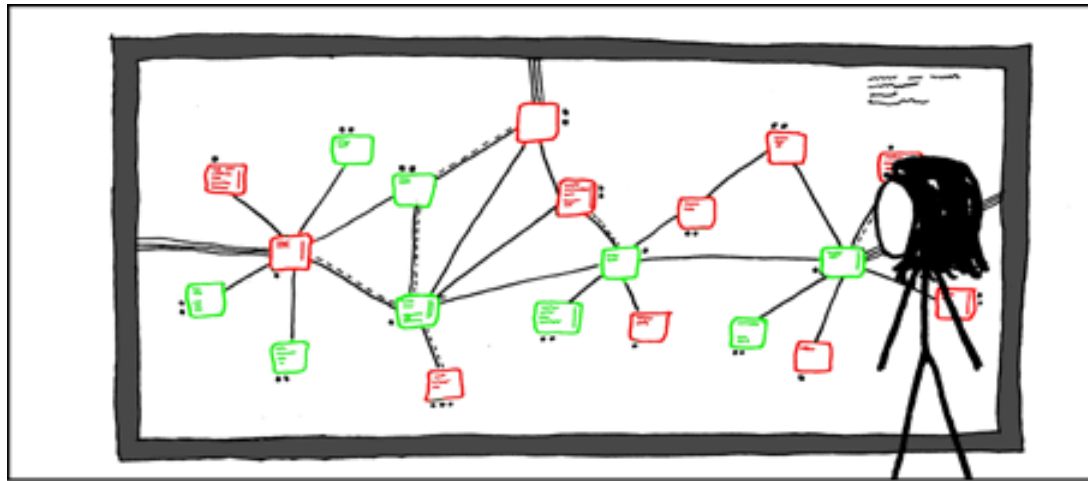
Virtualization, Networks & Communication.

Lecturer: Eran Tromer

Teaching assistant: Ariel Haviv

Advisor: Assaf Harel

Network (<http://xkcd.com/350/>)



Virtual Networking

1

Virtual Machines & Networks

2

Network Tools

3

TCP/IP

4

Firewall Functionality

Virtual Networking

1

Virtual Machines & Networks

2

Network Tools

3

TCP/IP

4

Firewall Functionality

Virtual Machines - Basics

- Virtual Machine:
 - "completely **isolated** guest operating system installation within a normal host operating system"
- Host – the OS that is really installed on the **physical** machine.
- Guest – the OS that sits inside a folder in that machine, and **'thinks'** it is installed on a physical machine.

Virtual Machines

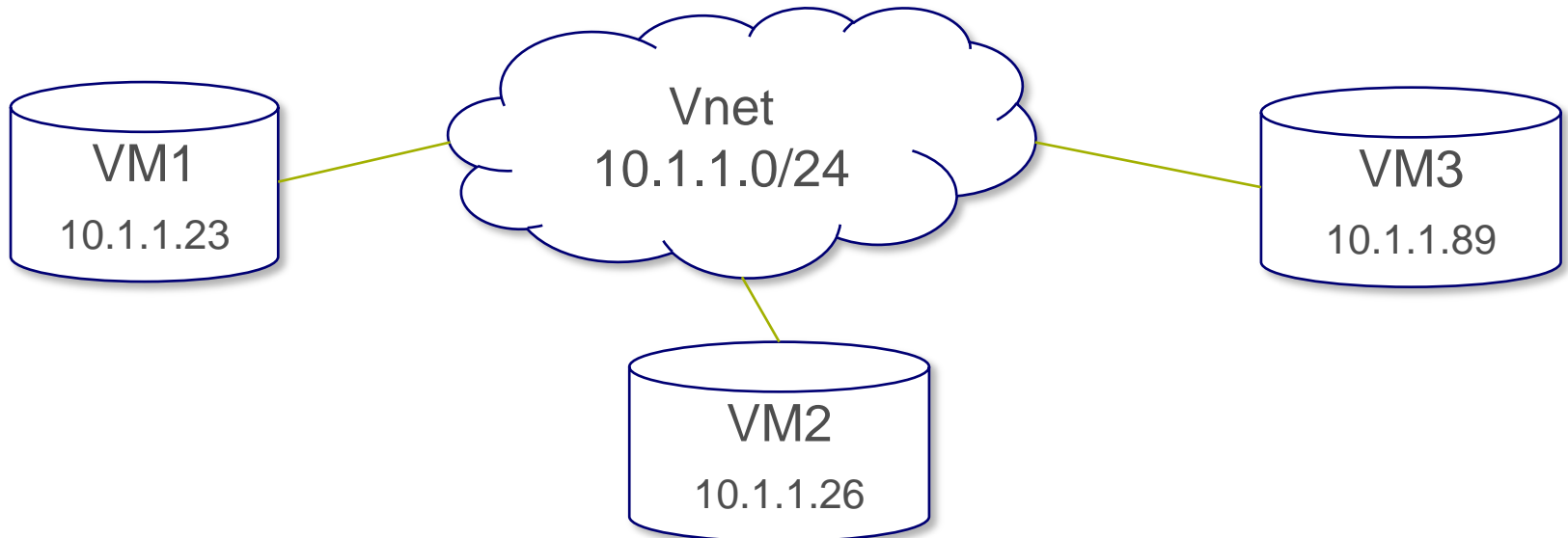
- The Host supplies virtualized hardware to the guest.
 - A hard-disk (really a few files on the host)
 - RAM (part of the host's RAM, with complex virtual memory algorithms)
 - IO (screen, keyboard, mouse, NIC)
 - Interrupts to/from the guest (like clock interrupts)
- NICs are connected either to the outer world, or to internal virtual networks.

IPv4 Networks

- In **IPv4**, every host has at least one name with the form:
www.xxx.yyy.zzz
 - Each of these four fields ranged from 0 to 255 (1 byte).
- Networks have names too:
 - Left x bits are network name.
 - Right $(32 - x)$ bits are zeroed.
- A network consists of several interconnected machines:
 - All have the same left x bits.
 - Each machine with rightmost $(32 - x)$ bits unique.
- A **router** forwards traffic from one network to another.

Virtual Networks

- Virtual networks are the same as a physical network, but virtualized. The cables only exist in the **memory** of the host OS.
- We can create virtual clusters of VMs, by connecting all of their virtual NICs to the same virtual network.



Virtual Networking

1

Virtual Machines & Networks

2

Network Tools

3

TCP/IP

4

Firewall Functionality

Network Monitor Tools

- Watch and record traffic: Wireshark.
- Inspects every packet – headers & data.

The screenshot shows a Wireshark capture of network traffic. The top section displays a list of captured packets with columns for packet number, time, source IP, destination IP, protocol, and length. The selected packet (Frame 1) is expanded to show its details in the 'Packet Details' pane. The details pane shows the following layers:

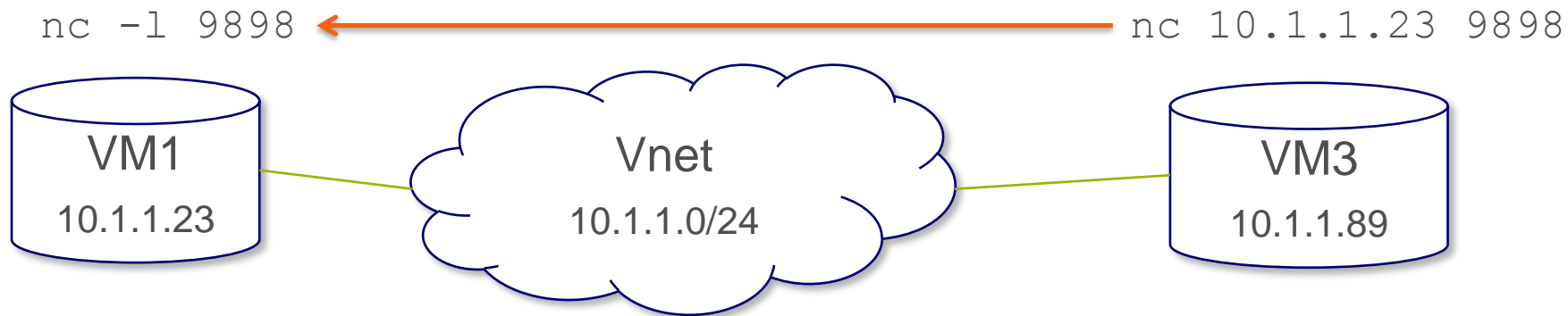
- Ethernet II, Src: Hewlett_6c:9f:1c (3c:d9:2b:6c:9f:1c), Dst: IPv4mcast_7f:ff:fa (01:00:5e:7f:ff:fa)
- Internet Protocol Version 4, Src: 91.90.130.72 (91.90.130.72), Dst: 239.255.255.250 (239.255.255.250)
- User Datagram Protocol, Src Port: sstp (1900), Dst Port: sstp (1900)
- Hypertext Transfer Protocol

The bottom section of the screenshot shows the raw packet data in hexadecimal and ASCII format:

```
0000 01 00 5e 7f ff fa 3c d9 2b 6c 9f 1c 08 00 45 00  ..^...<. +]....E.
0010 02 10 04 32 00 00 01 11 e6 0e 5b 5a 82 48 ef ff  ...2.... ..[Z.H..
0020 ff fa 07 6c 07 6c 01 fc 86 e7 4e 4f 54 49 46 59  ...].l.  ..NOTIFY
0030 20 2a 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73  * HTTP/ 1.1..Hos
0040 74 3a 32 33 39 2e 32 35 35 2e 32 35 35 2e 32 35  t:239.25 5.255.25
0050 30 3a 31 39 30 30 0d 0a 4e 54 3a 75 72 6e 3a 73  0:1900.. NT:urn:s
0060 63 68 65 6d 61 73 2d 75 70 6e 70 2d 6f 72 67 3a  chemas-u pnp-org:
0070 73 65 72 76 69 63 65 3a 43 6f 6e 6e 65 63 74 69  service: Connecti
0080 6f 6e 4d 61 6e 61 67 65 72 3a 31 0d 0a 4e 54 53  onManage r:1..NTS
0090 3a 73 73 64 70 3a 61 6c 69 76 65 0d 0a 4c 6f 63  :ssdp:al ive..Loc
00a0 61 74 69 6f 6e 3a 68 74 74 70 3a 2f 2f 39 31 2e  ation:ht tp://91.
00b0 39 30 2e 31 33 30 2e 37 32 3a 32 38 36 39 2f 75  90.130.7 2:2869/u
00c0 70 60 70 60 6f 73 74 76 7e 64 60 60 73 61 70 60  rroduct / udh3200f
```

Generating traffic - netcat

- Use netcat on both ends. One listens on a port, one initiates the conversation.
- Then clear text (from stdin or a file) can be transferred both ways.
- Wireshark can record the traffic on each of these machines.



Generating traffic – hping3

- Generates traffic in one direction.
- Doesn't care if anyone listens on the other side.
- Can send packets in any protocol, and even raw packets.



Virtual Networking

1

Virtual Machines & Networks

2

Network Tools

3

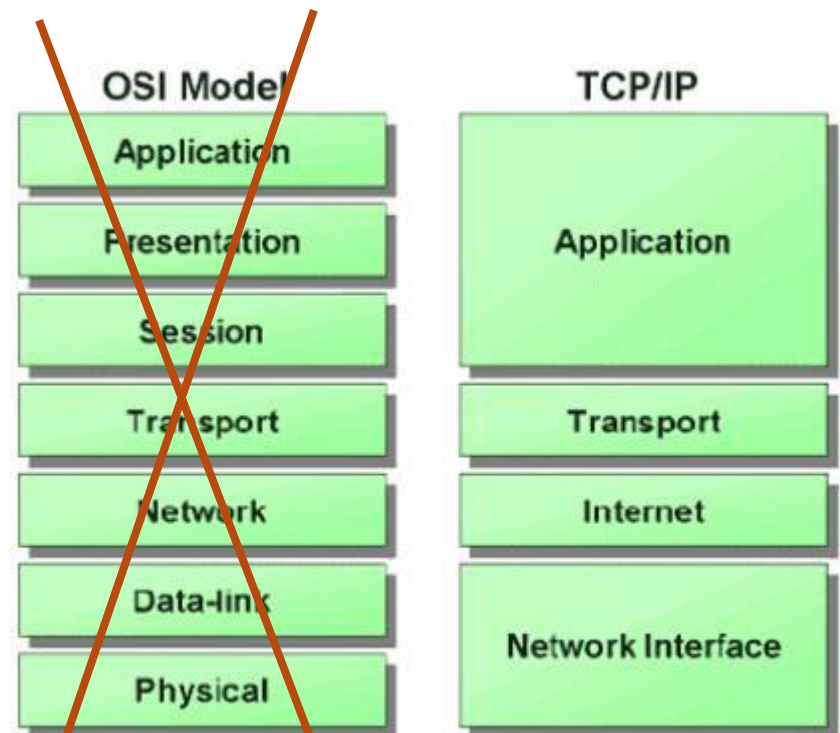
TCP/IP

4

Firewall Functionality

OSI Model

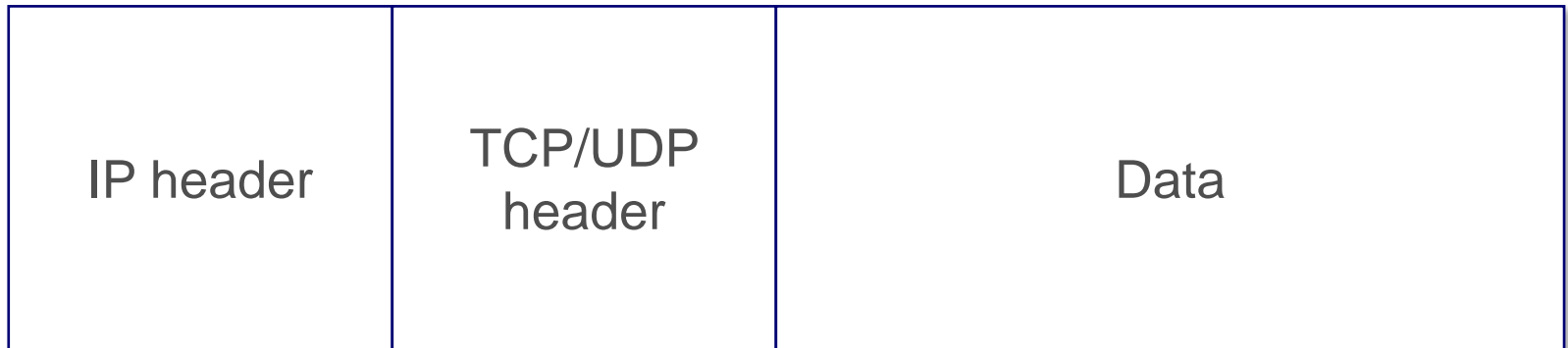
- A layer serves the layer above it and is served by the layer below it.
- Application – the actual content of the conversation.
- Transport – making sure the communication arrives to its destination, and in the right order.
- Network – makes sure two hosts on different networks can communicate.



TCP/IP and the OSI model

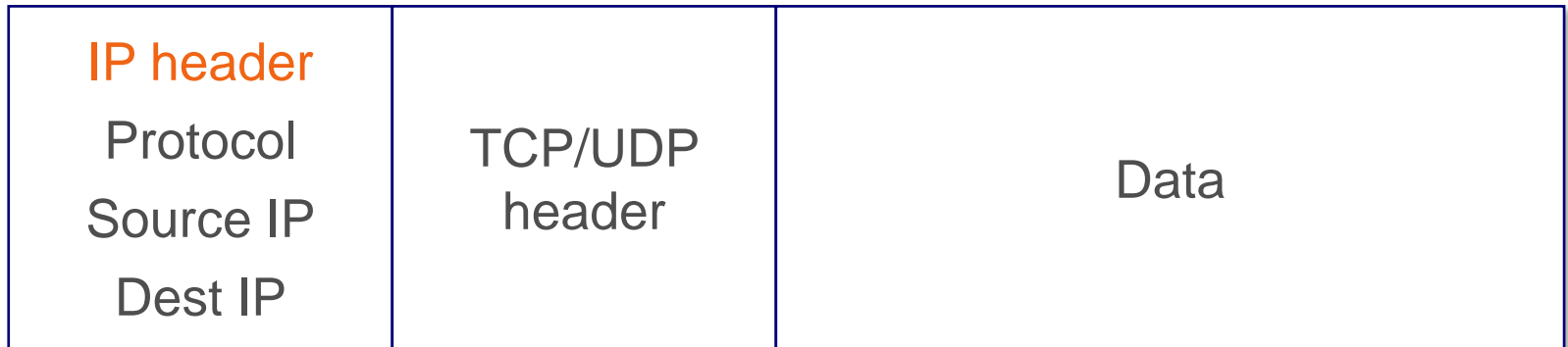
TCP/IP

- A protocol suite named after 2 very important components:
 - TCP – a very important protocol in the transport layer.
 - IP – a very important protocol in the network layer.
- Used everywhere. The standard of today's internet.
- A large portion of the chunks of data we will encounter will look like this:



TCP/IP - Network

- The **IP header** contains information like source and destination IP addresses, the protocol of the encapsulated transport header etc.
- IP allows one host in one network to connect to another host, on a different network.



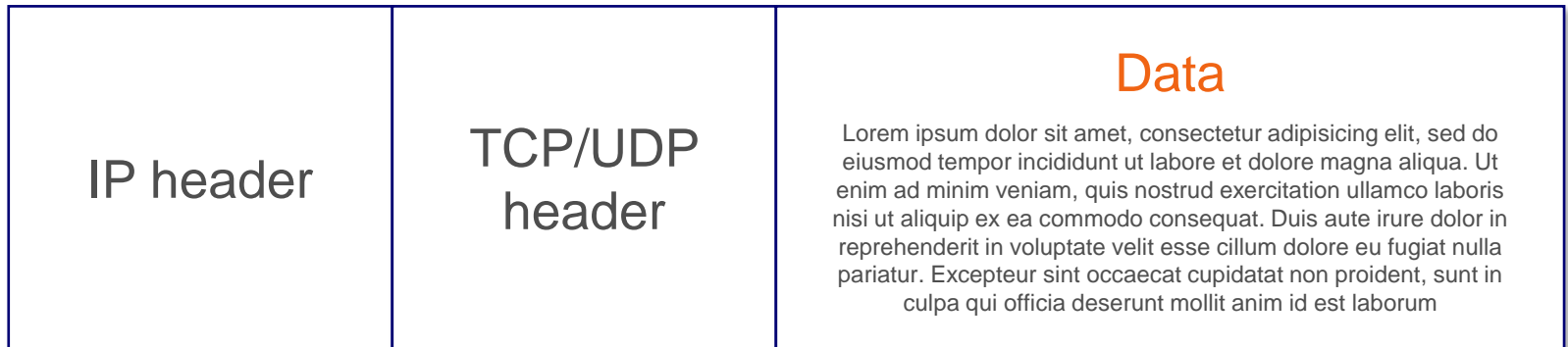
TCP/IP - Transport

- UDP header contains only source and destination ports, because it is a stateless protocol.
- TCP contain additional information:
 - Seq#, Ack#, 9 control bits, checksum, etc.
- The transport layer enables **different applications** on the host to communicate with the outer world, without disturbing each other.



TCP/IP - Payload

- The Data (called payload too), is the information passed between the two applications. It can be a web page, a file transferred through FTP, etc.



Virtual Networking

1

Virtual Machines & Networks

2

Network Tools

3

TCP/IP

4

Firewall Functionality

Example: TCP Handshake

- To initiate a TCP connections, we have a **3-way handshake**:
 - SYN + seq(X)
 - “Hello, I’m here. I chose X.”
 - SYN + seq(Y) + ACK + ack(X+1)
 - “I heard your X. here’s your X+1. I’m here too, I chose Y.”
 - ACK + ack(Y+1)
 - “I heard your Y, here’s your Y+1.”
- Now a connection has been established.
- SYN, ACK, seq#, ack# are all fields in the TCP header.

Connection Tracking

- A firewall which tracks connections will:
 - Open a new connection table entry when inspecting a packet with SYN flag on. The sender is the client.
 - Update the entry when the SYN+ACK is replied. The host that replied is the server.
 - Update the entry to active connection when the ACK is sent. Usually set a long timeout to the entry.
 - Close the entry with:
 - FIN → FIN+ACK → ACK
 - RST
 - Timeout expired
- Stricter tracking includes also seq# and ack# enforcement.