Chair of Distributed Systems and Security
School of Computation, Information and Technology
Technical University of Munich

Computer Networking and IT Security (INHN0012)

Tutorial 10

Problem 1 Network Address Translation

This task will look at the forwarding of IP packets (IPv4) when using a NAT-enabled router. For the mapping between public and private IP addresses, a NAT-enabled router has a mapping table that stores the relationship between the local and the global port. Many NAT-enabled devices also store additional information such as the remote IP address or the router's own global IP address (e.g., if the router has more than one global IP). We will refrain from doing this here.
Figure 1.1 shows the network topology. Router R1 has NAT enabled, using a private IP address on eth0 and a public IP address on wan0. Router R2 does not use NAT. PC2 had already communicated with Server 2, which created the entry in R1's NAT table (see Figure 1.1). Where you have the freedom, choose sensible values for the IP addresses and port numbers.

a)* Assign suitable IP addresses to PC1 and wan0 of R1. The subnet is 10.0.0.0/24.

Possible choices are

- PC1: 10.0.0.1
- R1.wan0: 10.0.0.254

b)* PC1 now establishes an HTTP connection to Server 2. Specify the source IP, destination IP, source port, destination port, and TTL fields of the IP or TCP header for the packets at the three highlighted locations in Figure 1.1. Also specify newly created entries in the NAT table of R1.
See Figure 1.1.

- **Between PC1 and R1:** TTL = 64
  It is important that the source port is larger than 1023 (values smaller than 1024 are *well-known ports* and cannot be used as source ports). Furthermore, it must not be larger than 65535, because port numbers are 16 bit long. The destination port is given as TCP 80 (HTTP).
- **R1 and R2** TTL = 63
  R1 replaces the private source IP with its own public IP address. The source port will usually remain unchanged (if it is not already in use). Otherwise, it is modified, for example by incrementing it. Otherwise, it is modified, for example by incrementing it. The choice of port numbers depends on the respective NAT type. If possible, the same port number is retained. At this point, a new entry in the NAT table is created: [10.0.0.1, 3627, 3627].
- **Between R2 and Server 2** TTL = 62
  No changes occur, because a normal router does not change IP addresses or port numbers. The TTL field is decremented as usual.

c) Server 2 now responds to PC1. In Figure 1.2, specify the header fields at the three named locations, as well as newly created entries in the NAT table of R1, analogous to the previous subtask.
We assume that the server sends out packets with a TTL = 64

- **Between Server 2 and R2** TTL = 64
  The server first addresses the response to R1 (where else?).
- **Between R2 and R1** TTL = 63
  R2 does not change anything (except TTL).
- **Between R1 and PC1:** TTL = 62
  R1 uses the entry in its own NAT table to determine the IP address of the real receiver. Afterwards the destination IP address and port number are replaced (if necessary) and the packet is forwarded.

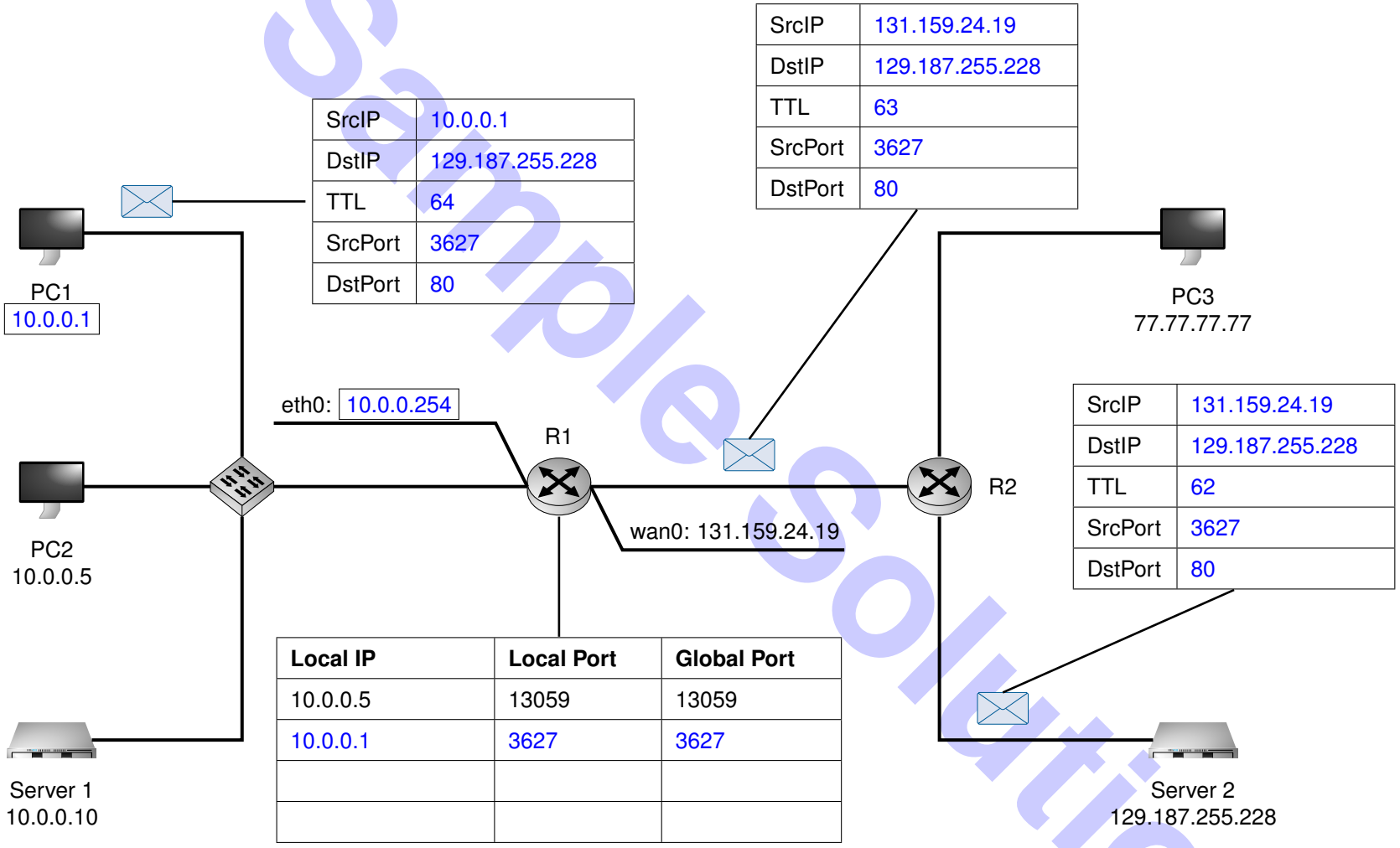

Figure 1.1: Solution sheet for problem 1a)/b)

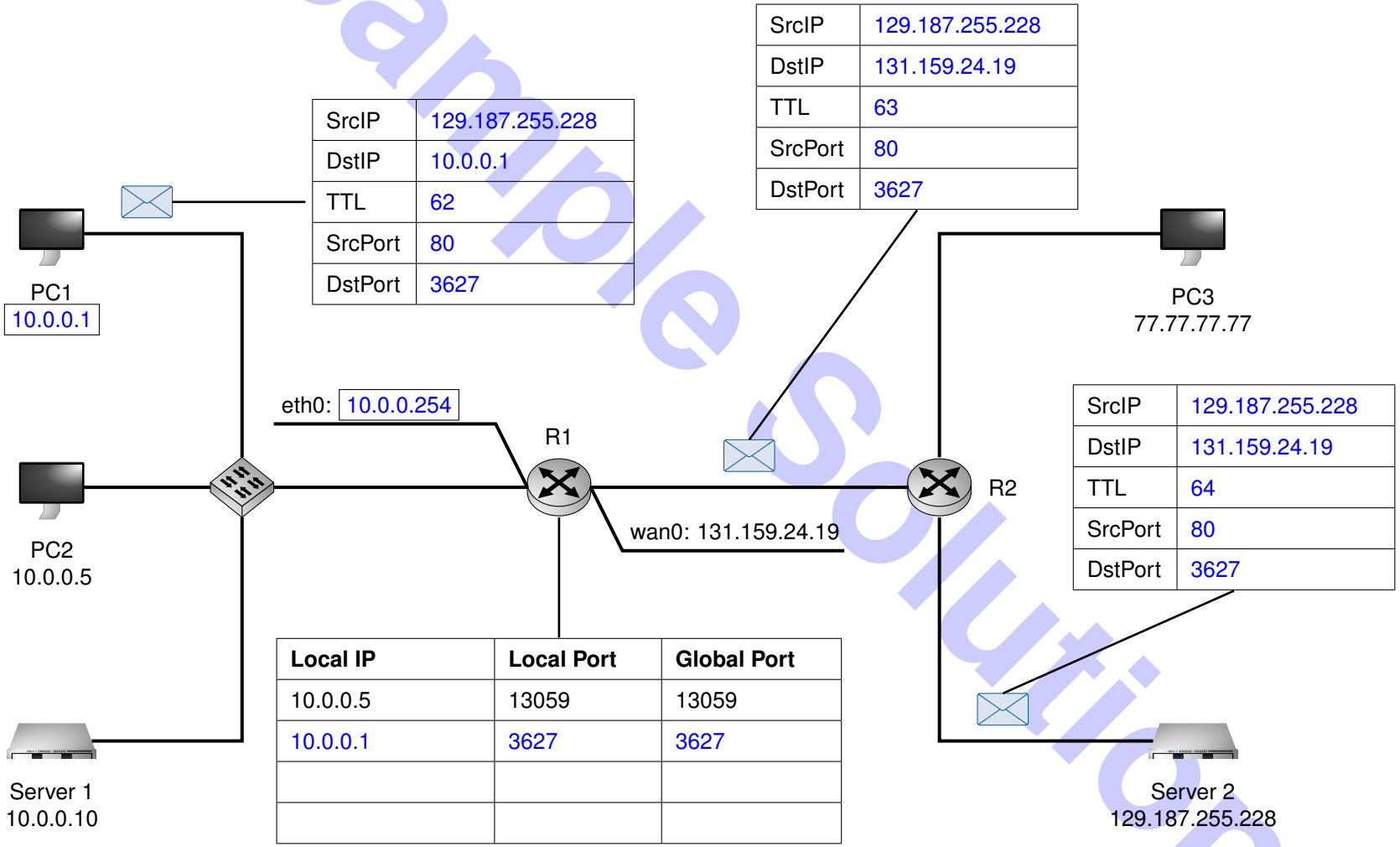

Figure 1.2: Solution sheet for problem 1c)

d)* Server 1 now also establishes a TCP connection to Server 2 on port 80. In doing so, it randomly chooses the source port 13059. Describe the problem that occurs at the NAT and how it is solved.

There is a collision with the first entry in the NAT table: The NAT router can no longer distinguish whether responses from Server 2 are destined for PC1 or Server 2, since the only distinguishing feature is the global port number.
The solution is that the NAT router checks whether the respective port is already in use before creating a new entry. If it is, the NAT chooses a random port number from the ephemeral port range (or increments the port number) and stores both the local and the new global port number. For incoming packets, the port number is translated back in the L4 PDUs.

e)* R1 receives a packet from PC3 addressed to `131.159.24.19:13059`. What will R1 do with this packet? What problems may arise from it?

R1 will translate the destination address of the packet according to the NAT table and forward it to PC2, even though the original entry was created for Server2.PC2 receives an „unexpected“ packet and must be able to handle it.The often erroneously assumed firewall function of NAT cannot be relied upon here.

f) Does a problem arise for PC2 when it receives a „random“ packet with TCP payload on a port with an existing connection?

The packet most likely has a different sender IP and source port and thus is therefore not associated with the existing connection.If the sender IP and source port „coincidentally“ match, the packet's sequence number will, with high probability, not fall within the currently valid receive window and will thus be discarded.

g)* What other distinguishing criteria could be used by a NAT router?

global IP (if the router has multiple interfaces/IP addresses configured),remote IP,remote portand the protocol number(TCP or UDP).

h)* What problem arises when PC1 sends an echo request to Server 2?

ICMP does not use port numbers therefore the NAT router can't create an entry.The response is thus discarded.

i) Describe a possible solution to the problem identified in the previous subproblem.

The NAT router could, in the case of ICMP packets, use the ICMP identifier in addition to the protocol number as a replacement for the missing port numbers.In this case, however, the NAT router must also differentiate between the various L3 SDU protocols (TCP, UDP, ICMP, etc.).

j) What problem arises if a NAT router receives ICMP TTL-exceeded messages and wants to forward them to the intended receiver (sender who caused the problem)? How can this problem be worked around?

TTL Exceeded messages are independent ICMP messages whose identifier was not entered in the NAT (messages are not generated in the local network, but from computers outside). An assignment to the original recipient is therefore not directly possible.ICMP TTL Exceeded messages contain, in addition to the ICMP header, also the IP header and the first 8 payload bytes of the triggering packet[1]. This allows the NAT to identify the triggering connection.For TCP and UDP, the port numbers can be found here; for ICMP messages, the original identifier.

k)* Now PC3 wants to establish an HTTP connection with Server 1. Can this work under the given circumstances? (Explain!)

PC3 cannot address packets directly to the address 10.0.0.10 because it is a private IP address which is not routed on the public Internet. If PC3 knows the public IP of R1, behind which Server 1 is located, it can address the packet to the IP address of R1 and TCP 80; but, according to the problem statement, R1 does not have a suitable entry in its NAT table and therefore can't identify the correct receiver of the packet in the local network.

l) How can this problem be avoided while maintaining a NAT?

A static mapping (so called portforwarding) can be configured in the NAT.
Example: 10.0.0.10 80 80 With this static entry, Server 1 can be reached with the IP address of R1 via the router R1 on port 80 from the outside.

Problem 2 Re-wired shark

Given is the hexdump shown in Figure 2.1 in network byte order of an Ethernet frame, which is to be analyzed in the following.
Note: To solve this task, you will need information from the cheatsheet.
Important: All following subproblems require justification through reasoning or through marking the corresponding header field or fields clearly in the hexdump. In the latter case, note the the subproblems character (e.g. *a)*) next to the marking.

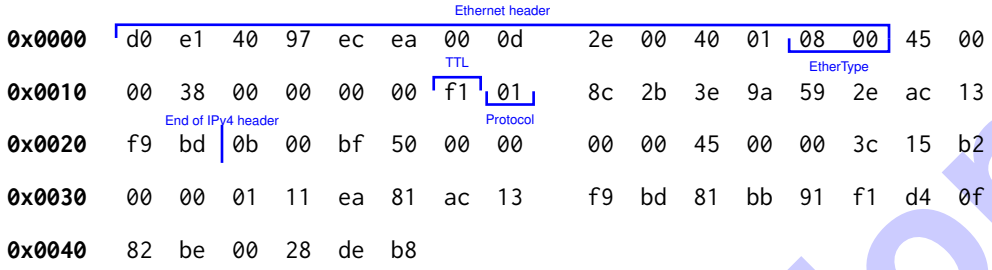

Figure 2.1: Hexdump of an Ethernet frame in network byte order

a)* Mark the beginning and end of the Ethernet header in Figure 2.1.

b) Which protocol is used at layer 3?

Ethertype `0x0800` stands for IPv4.

c) Determine the length of the layer 3 header and mark its end in Figure 2.1.

For IPv4, the header length is encoded as a multiple of 4 B in the lower nibble of the first byte (IHL). Specifically: `0x45` → `0x5` = 20 B.

d) State – if contained in the packet – the TTL or HopCount in decimal **and** hexadecimal notation.

Since it is IPv4, we look for the **TTL** which is `0xf1` = 241

e) Justify to which protocol the L3 SDU belongs.

The protocol number is `0x01` = ICMPv4.

Given is the SDU of layer 3 shown in Figure 2.2 **of another packet**. It is known that this is ICMPv4.

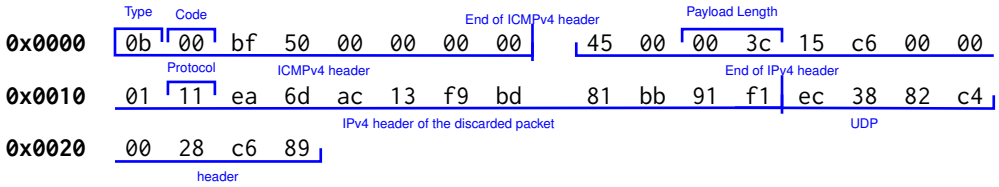

Figure 2.2: ICMP message including ICMP header in network byte order

f)* Determine the type and code of the ICMP message.

Time Exceeded / TTL expired in transit

g) What causes such a message?

If a router receives a packet with TTL=1 (or, for IPv6, Hop Limit 1) that should be forwarded, the packet is discarded and instead a Time Exceeded / TTL expired in transit message is sent back to the original sender of the discarded packet.

h)* Mark the end of the ICMP header in Figure 2.2.

i) Explain what the payload of such a message fundamentally contains.

The payload contains the IP header as well as at least the first 8 B of the L3 SDU of the packet that triggered the ICMP TTL Exceeded message.

j) Was IPv4 or IPv6 used to transmit the original message that triggered this error?

IPv4, because we know the error message is ICMPv4. If it were IPv6, we would have received an ICMPv6 error message.

k) What is the length of the original IP packet?

Total Length in the encapsulated IP packet is `0x003c` = 60 Bytes.
Note that the ICMP error does not contain the whole original IP message.

l) Mark the end of the original messages IP header in Figure 2.2.

m) What layer 4 protocol was used in the original message?

Protocol in the encapsulated IP packet is `0x11` = UDP.