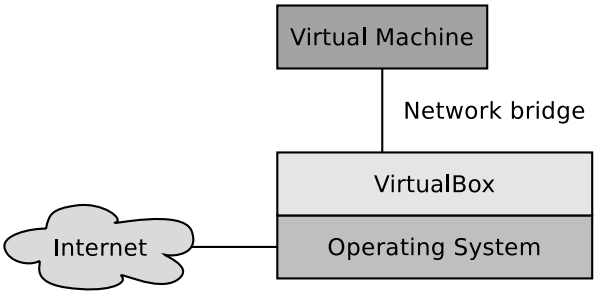
Lab Exercise Sheet 1 – (Sample Solution)

Document and analyze your experimental procedures by using your Wireshark and terminal recordings. Note all relevant intermediate steps. Mark and explain all relevant information, such as protocol header fields, MAC addresses, IP addresses, port numbers. If you have little experience with Linux, you may need to do some research. **Send your self prepared experiment documentation in the PDF file format to cocos@stud.fra-uas.de and christianbaun@fb2.fra-uas.de. Alternatively, fill out the document, print it out, and submit it during one of the exercise sessions.**

Sample Solution (No Guarantee !!!)

1. Install the virtualization software VirtualBox[1] on your personal computer. VirtualBox is available for the operating systems Linux, Windows and Mac OS X.



2. Create a virtual machine with the operating system Ubuntu Server[2] and configure the virtual network interface to be a bridged network interface to the network interface of your node (personal computer), which has a working internet connection. Do not use Network Address Translation (NAT).

   - Install[3] the package `xfce4` to get a graphical user interface. The download and installation of the packages requires some time. After the installation has finished, you can start the GUI with the command `startx`.
   - Install the VirtualBox Guest Additions[4].
   - Install the network protocol analyzer Wireshark (package `wireshark`). Configure it in a way that it can be used without root privileges.
   - Install the network interface of the virtual machine in a way, that the IPv4 address will be fetched via DHCP (this is done per default).
   - Install the text-based web browser Lynx (package `lynx`).
   - Install the network diagnostic tool `traceroute` (package `traceroute`).

[1] `http://www.virtualbox.org`
[2] `http://releases.ubuntu.com/16.04/ubuntu-16.04.3-server-amd64.iso`
[3] `http://www.google.de/search?q=Install+packages+ubuntu+command+line`
[4] `http://www.google.de/search?q=install+virtualbox+guest+additions+Ubuntu+server`

3. Renew the IPv4 address of the guest operating system inside the virtual machine via DHCP and monitor this procedure via Wireshark (hint: set the filter inside Wireshark to value `bootp`). Expand only the first layer of the DHCP protocol inside the protocol window of Wireshark and copy the content of all DHCP messages into this field:

```
dhclient -r && dhclient
_____________________________________________________________
DHCP Release - Transaction ID 0xd218ab6a
Frame3:342bytesonwire(2736bits),342bytescaptured(2736bits)
   oninterface0
EthernetII,Src:PcsCompu_52:da:cd(08:00:27:52:da:cd),
Dst:Avm_74:70:92(34:31:c4:74:
70:92)
InternetProtocolVersion4,Src:192.168.178.57,Dst:192.168.178.1
UserDatagramProtocol,SrcPort:68,DstPort:67
BootstrapProtocol(Release)
_____________________________________________________________
DHCP Discover - Transaction ID 0x1d1f3874
Frame4:342bytesonwire(2736bits),342bytescaptured(2736bits)
   oninterface0
EthernetII,Src:PcsCompu_52:da:cd(08:00:27:52:da:cd),
Dst:Broadcast(ff:ff:ff:ff:ff:ff)
InternetProtocolVersion4,Src:0.0.0.0,Dst:255.255.255.255
UserDatagramProtocol,SrcPort:68,DstPort:67
BootstrapProtocol(Discover)
_____________________________________________________________
DHCP Offer- Transaction ID 0x1d1f3874
Frame5:590bytesonwire(4720bits),590bytescaptured(4720bits)
   oninterface0
EthernetII,Src:Avm_74:70:92(34:31:c4:74:70:92),
Dst:Broadcast(ff:ff:ff:ff:ff:ff)
InternetProtocolVersion4,Src:192.168.178.1,Dst:255.255.255.255
UserDatagramProtocol,SrcPort:67,DstPort:68
BootstrapProtocol(Offer)
_____________________________________________________________
DHCP Request - Transaction ID 0x1d1f3874
Frame6:342bytesonwire(2736bits),342bytescaptured(2736bits)
   oninterface0
EthernetII,Src:PcsCompu_52:da:cd(08:00:27:52:da:cd),
Dst:Broadcast(ff:ff:ff:ff:ff:ff)
InternetProtocolVersion4,Src:0.0.0.0,Dst:255.255.255.255
UserDatagramProtocol,SrcPort:68,DstPort:67
BootstrapProtocol(Request)
_____________________________________________________________
DHCP ACK - Transaction ID 0x1d1f3874
Frame7:590bytesonwire(4720bits),590bytescaptured(4720bits)
   oninterface0
EthernetII,Src:Avm_74:70:92(34:31:c4:74:70:92),
Dst:Broadcast(ff:ff:ff:ff:ff:ff)
InternetProtocolVersion4,Src:192.168.178.1,Dst:255.255.255.255
UserDatagramProtocol,SrcPort:67,DstPort:68
BootstrapProtocol(ACK)
_____________________________________________________________
Source: DHCP.pdf
```

What is the sender address of the DHCP client?

0.0.0.0

Why uses the DHCP client use his sender address?

because the Client has no IP-Address

To which destination IP address does the DHCP client send messages?

255.255.255.255

To which destination MAC address sends the DHCP client messages?

ff:ff:ff:ff:ff:ff

To which destination IP address are messages send by the DHCP server?

255.255.255.255

To which destination MAC address are messages send by the DHCP server?

ff:ff:ff:ff:ff:ff

Which IP address has been offered to the DHCP client by the DHCP server?

192.168.178.57

Which lease time was offered by the DHCP server?

(864000s) 10 days

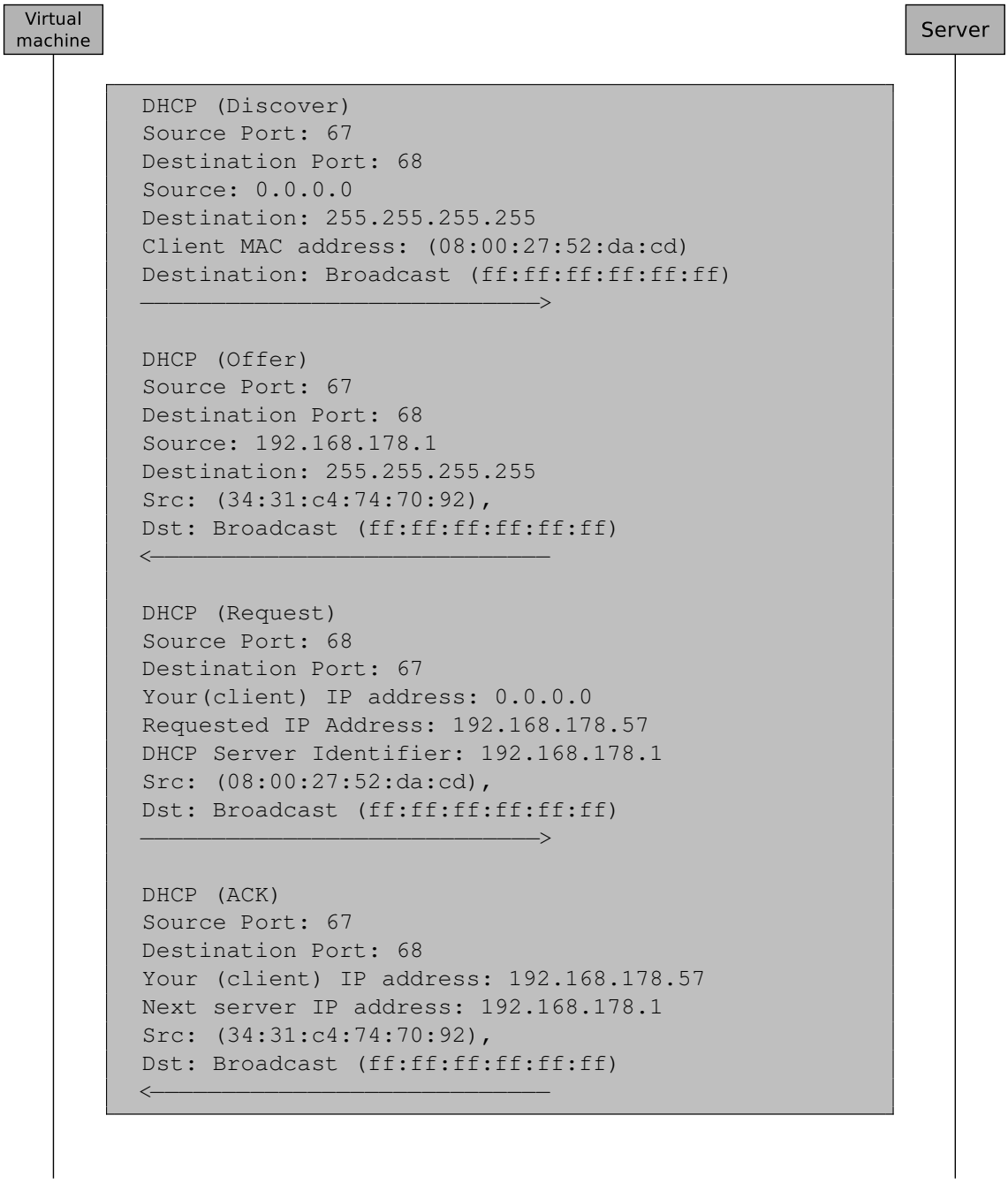
Which IP address did the DHCP client select and request in the reply to the DHCP server?

192.168.178.57

Which IP address did the DHCP server acknowledge to the DHCP client?

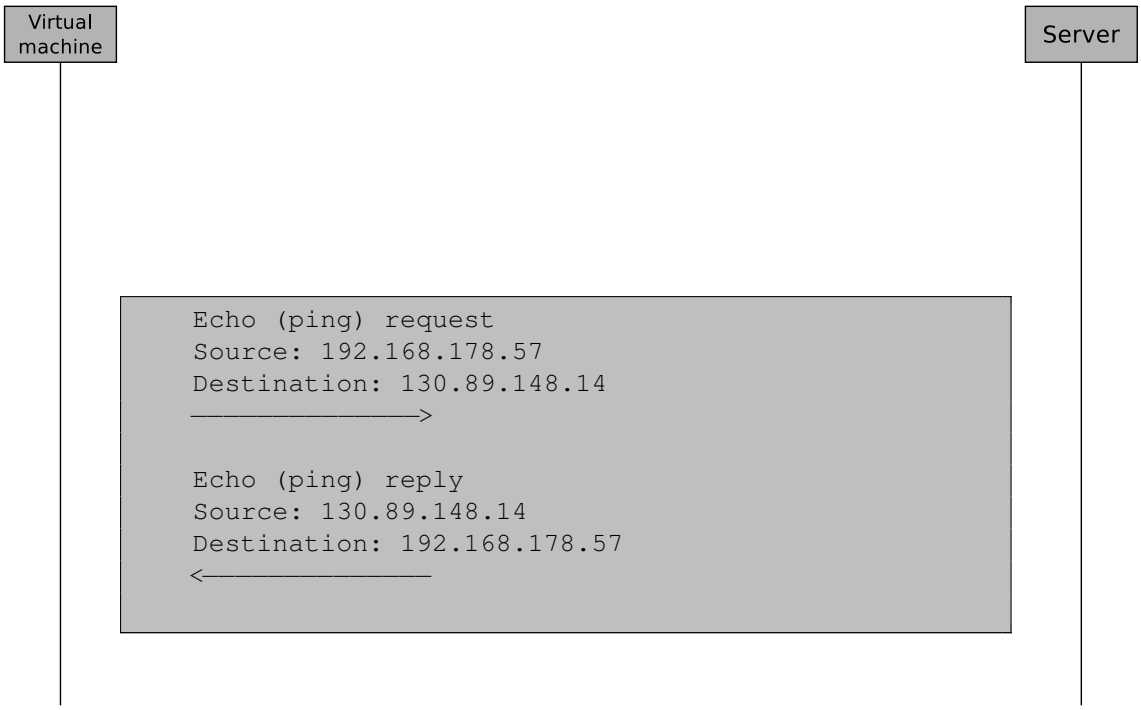
192.168.178.57

Sketch inside the Message Sequence Chart (MSC) the sequence of the IPv4 address assignment by using DHCP. Specify for each transmitted message the transmission direction, the MAC addresses and IP addresses, as well as the port numbers and DHCP message name.



4. Send a ping request from inside the guest operating system via the bridged network adapter to the address `debian.org`. Monitor the Ethernet frames and IPv4 packages of the ping operation (hint: set the filter inside Wireshark to value `icmp`).

   Sketch inside the Message Sequence Chart (MSC) the sequence of the ICMP transmissions that was caused by the ping operation.



   The `ping` command has triggered a DNS resolution because the domain name needed to be resolved into the IP address of the web server. Monitor the Ethernet frames and IPv4 packages of the DNS resolution (hint: set the filter inside Wireshark to value `dns`).

   Which port number is used by the DNS server per default?

   53

   What is the destination IP address of the DNS query sent (hint: set the filter inside Wireshark to value `dns`)?

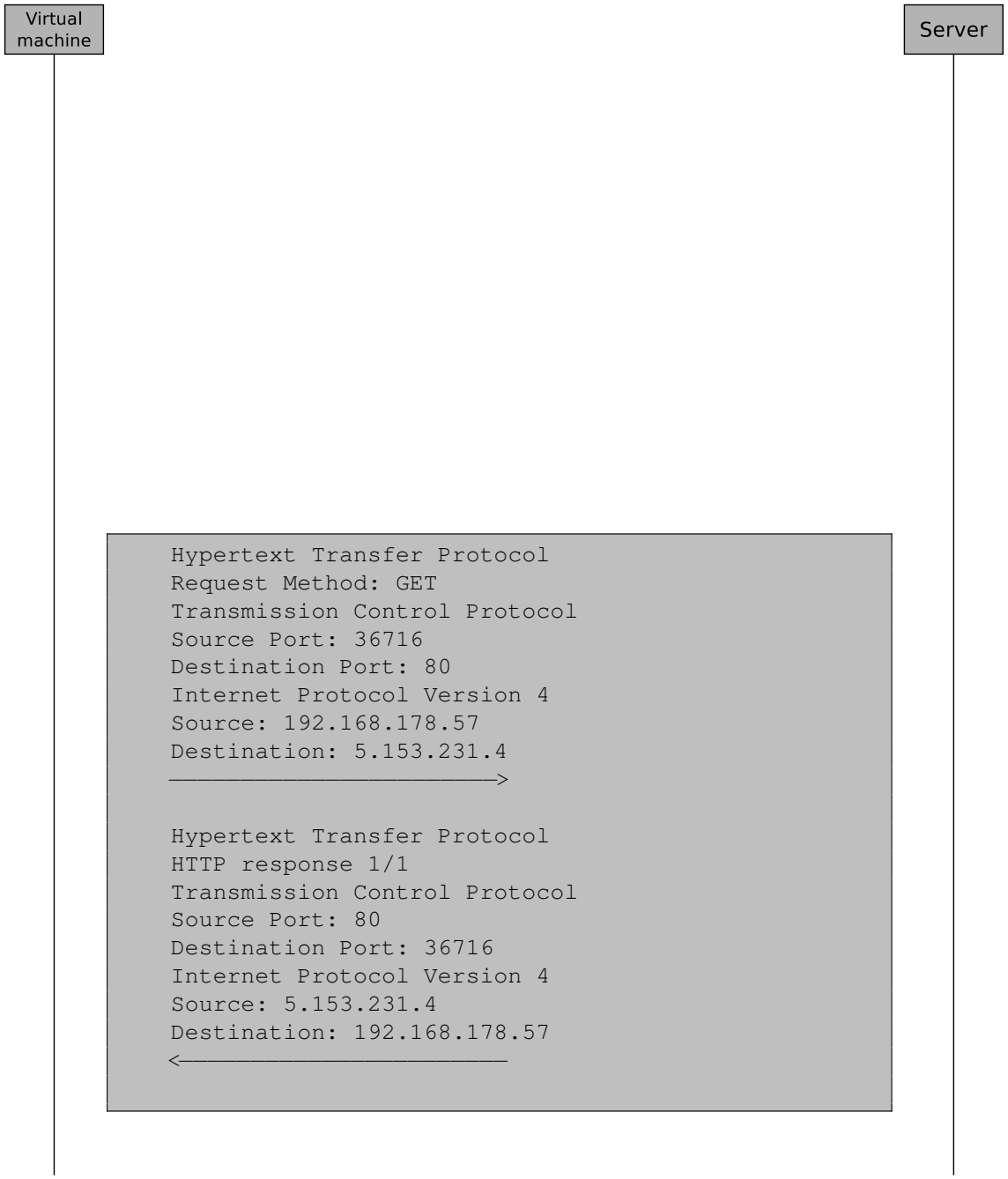
   192.33.4.12

   Which IP address is transmitted as a reply inside the DNS response?

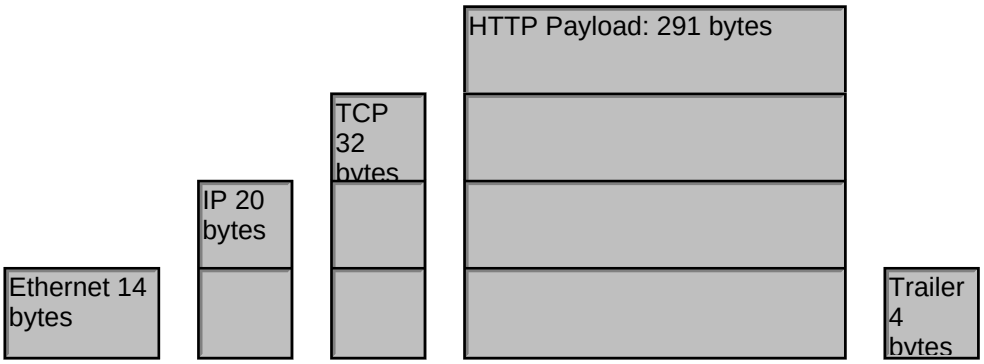
   130.89.148.14

5. Request the web page behind the address `debian.org` by using the text-based web browser Lynx from inside the guest operating system via the bridged network adapter. Monitor the Ethernet frames and IPv4 packages.

   Sketch inside the Message Sequence Chart (MSC) the sequence of the HTTP- and TCP transmissions that was caused by requesting the web page.



Show the protocol stack of the first HTTP response (starting with OSI layer 2). Fill in the correct number of Bytes of the headers, trailer and payloads. Also name the protocols used inside the single layers.



Calculate the protocol overhead in Bytes for the transmission of the HTTP response?

(Ethernet 14 bytes + Trailer 4 bytes) + (IP 20 bytes) + (TCP 32 bytes) = 70 bytes

Calculate the protocol overhead ratio in % for the transmission of the HTTP response. Possible OSI layer 1 overhead should be ignored.

(Payload 291 bytes) + (70 bytes Header Information) =

361 bytes transmission size

70 bytes Overhead divided by 361 bytes transmission size

70 / 361 = 0,1939 * 100% = 19,39%

6. Use the command line tool `traceroute` to print out the routers on the network connection between your local site and the web page behind the address `debian.org` (hint: set the filter inside Wireshark to value `icmp`). Copy the output of the `traceroute` command into this field:

```
traceroute to debian.org (128.31.0.62), 30 hops max, 60 byte
   packets
1 192.168.178.1 (192.168.178.1) 1.207 ms 1.319 ms 3.127 ms
2 192.168.0.1 (192.168.0.1) 6.140 ms 8.200 ms 11.168 ms
3 * * *
4 de-fra01b-rc1-ae28.fra.unity-media.net (81.210.141.33) 29.344 ms
    29.463 ms
30.285 ms
5 de-fra01b-ri2-ae30-0.aorta.net (84.116.134.166) 30.260 ms 30.443
    ms 30.393 ms
6 213.46.177.134 (213.46.177.134) 30.341 ms 34.562 ms 34.752 ms
7 be2256.ccr42.fra03.atlas.cogentco.com (154.54.36.250) 29.652 ms
   29.322 ms
27.648 ms
8 be12194.ccr41.lon13.atlas.cogentco.com (154.54.56.93) 39.839 ms
be2249.ccr42.ams03.atlas.cogentco.com (154.54.36.214) 24.013 ms
be12194.ccr41.lon13.atlas.cogentco.com (154.54.56.93) 33.107 ms
9 be12488.ccr42.lon13.atlas.cogentco.com (130.117.51.41) 31.247 ms
    36.086 ms
38.341 ms
10 be2983.ccr32.bos01.atlas.cogentco.com (154.54.1.178) 125.305 ms
be3255.ccr32.bos01.atlas.cogentco.com (66.28.4.30) 104.966 ms
be2983.ccr32.bos01.atlas.cogentco.com (154.54.1.178) 123.590 ms
11 38.104.186.186 (38.104.186.186) 113.315 ms 113.299 ms 111.267
   ms
12 dmz-rtr-1-external-rtr-3.mit.edu (18.69.7.1) 122.534 ms 115.259
    ms 120.922 ms
13 dmz-rtr-2-dmz-rtr-1-2.mit.edu (18.69.4.2) 104.037 ms dmz-rtr-2-
   dmz-rtr-1-1.mit.edu
(18.69.3.2) 105.879 ms 108.771 ms
14 * * *
15 mirror-csail.debian.org (128.31.0.62) 114.365 ms 113.351 ms
   112.442 ms
```

How many routers are on the network connection between your local site and the web server?

15