

Lab Exercise Sheet 2

Deadline: 7th June 2020

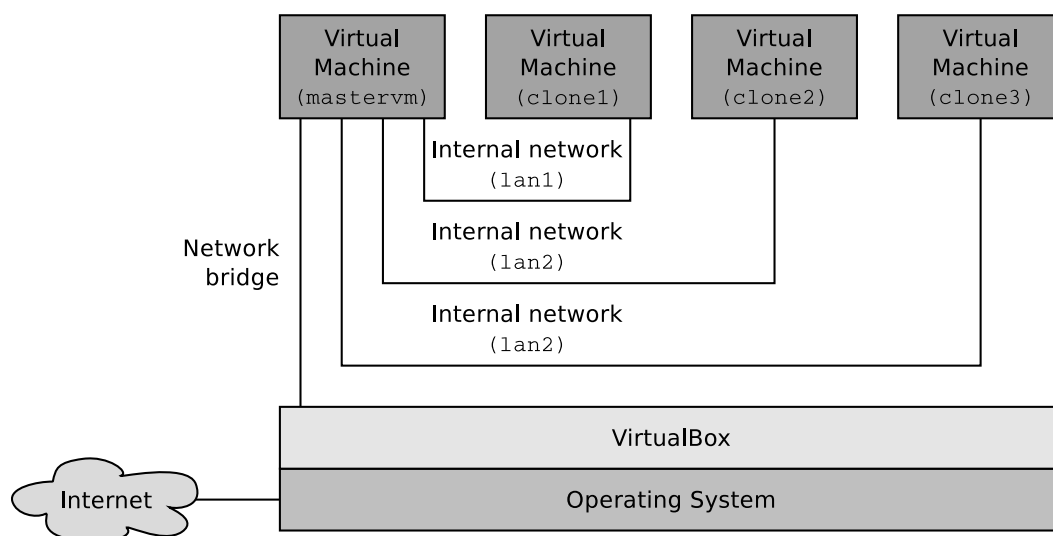
Document and analyze your experimental procedures by using your Wireshark and terminal recordings. Note all relevant intermediate steps. Mark and explain all relevant information, such as protocol header fields, MAC addresses, IP addresses, port numbers. If you have little experience with Linux, you may need to do some research. **Upload your self prepared experiment documentation in the PDF file format on Moodle until 7th June 2020! Alternatively, fill out the document, print it out, and submit it during one of the exercise sessions.**

First name:

Last name:

Student number:

1. Clone the VM from the 1st lab exercise (from now on we will call this VM **mastervm**) three times with the VirtualBox user interface.
 - Specify for each new VM a unique name (e.g. **clonevm[1-3]**) and specify that each VM will get a new MAC address.
 - Add three new virtual interfaces to the **mastervm** via the VirtualBox user interface. Each one of these new network interfaces must be attached to different *internal networks*. The name of each internal network must be unique. e.g. **lan[1-3]**.
 - The network interface of each one of **clonevm[1-3]** also need to be attached to an *Internal Network* in the VirtualBox user interface. Connect **clonevm1** to the first internal network (e.g. **lan1**), **clonevm2** to the second internal network (e.g. **lan2**), and so on.



Check the relevant MAC addresses and write them into this table:

Your local Router to the internet:
Physical network interface of your host:
 `mastervm` (bridged interface):
 `mastervm` (internal interface 1):
 `mastervm` (internal interface 2):
 `mastervm` (internal interface 3):
 `clonevm1` (internal interface):
 `clonevm2` (internal interface):
 `clonevm3` (internal interface):

The `mastervm` should operate as a Bridge/Switch between the bridged network interface and the three new network interfaces for `clonevm[1-3]`, which are attached to the internal networks `lan[1-3]`.

- Install the command line tools for bridging Ethernet connections (package `bridge-utils`) on the `mastervm`.
- You have several options to implement the IP forwarding.
 - Option 1: Create¹ a new logical Bridge with the command line tool `brctl` on the `mastervm`. Add the four virtual network interfaces of the `mastervm` to the logical Bridge.
 - Option 2: Specify for `lan[1-3]` three independent address spaces (e.g. `192.168.10.0/24`, `172.22.0.0/16` and `192.168.60.0/24`). Assign² valid IP addresses and further network configuration parameters to the virtual network devices inside the `mastervm` and `clonevm[1-3]`. Implement IP package forwarding (NAT-Masquerading)³ **Hint:** *This solution is recommended for the course of the semester!*

General Hints:

- The naming convention of the interfaces is the following: **Adapter 1:**`enp0s3`, **Adapter 2:**`enp0s8`, **Adapter 3:**`enp0s9`, **Adapter 4:**`enp0s10`
- The configuration made on the command-line for `bridge-utils` and `iptables` are not permanent! They are reverted after a reboot! To make the changes permanent you need to make the configuration inside the file `/etc/network/interfaces!`
- If you have problems with DNS name resolution you should disable IPv6 inside your virtual machine!

¹<http://www.tldp.org/HOWTO/BRIDGE-STP-HOWTO/set-up-the-bridge.html>

²This can be done with command line tools like `ip` or `ifconfig` or inside the file `/etc/network/interfaces`.

³This can be done with command line tools like `ip` or `iptables` or inside the file `/etc/network/interfaces`.

Copy the content of the IP routing table of the `mastervm` into this field:

Copy the relevant content of the file `/etc/network/interfaces` of the `mastervm` into this field:

2. Check the content of the ARP cache on the `mastervm`. Copy the content of the ARP cache of the `mastervm` into this field:

Send some ping requests between the `mastervm` and `clonevm[1-3]`. Copy the content of the ARP cache of the `mastervm` into this field:

3. Do a ping operation from `clonevm1` to the address `debian.org`. The ping operation will cause the transmission of an ARP request and an ARP reply. Monitor these transmissions with Wireshark from the `mastervm`. Copy the relevant information (MAC addresses and IP addresses of sender and target) of the ARP request into this field:

Copy the relevant information (MAC addresses and IP addresses of sender and target) of the ARP reply into this field:

Which network protocols are involved in the transmission of the ARP messages? Assign them to the protocol stack.

Layer 7:

Layer 6:

Layer 5:

Layer 4:

Layer 3:

Layer 2:

Layer 1:

What is the destination address of the frame, that is used to transmit the ARP request?

What is the value of the `type` header field inside the frame, that is used to transmit the ARP request?

Which IP address belongs to the sender MAC address in the header of the frame, that is used to transmit the ARP request?

What is the destination IP address, to which the matching MAC address is searched inside the ARP request?

What is the name of the header field, that is used to store the destination IP address inside the ARP request?

What is the value of the header field, that is used to store the destination MAC address inside the ARP request?

Expand the protocol window of the first Layer of protocols inside Wireshark, that are involved in the first echo request/reply message pair and copy the relevant information into this field:

Which network protocols are involved in the transmission of the echo request/reply message pair? Assign them to the protocol stack.

Layer 7:

Layer 6:

Layer 5:

Layer 4:

Layer 3:

Layer 2:

Layer 1:

What is the length of the IP header in bytes?

What is the length of the ICMP payload in bytes?

To which destination IP address is the ICMP request sent?

What is the matching MAC address?

What is the value of the `type` header field inside the frame, that is used to transmit the ICMP request?

What is the purpose of the sequence number inside the ICMP header?