Technical Environment

BME Cloud Infrastructure: VIK cloud (https://cloud.bme.hu/ -> https://fured.cloud.bme.hu/)

Access: https://fured.cloud.bme.hu/accounts/login/ , login via BUTE Central Login Page

Template to create students' own Virtual Machine (VM): **VWS-AutNetCommSys2020-edu v3.1** (see details in a separate document: *Create_VM_in_FUred_Cloud_v1.pdf*)

Students VM: Win10, WireShark Protocol Analyzer, putty SSH and telnet client

Two common VMs

1. Public web server (Win10, hfs), IPv4: 10.9.1.97
1. Public telnet and ssh server (Ubuntu 18.04) with 4 users (user1 …user4, PassW1…PassW4), IPv4: 10.9.0.53

Tasks

There are guided demonstration tasks and individual tasks in the laboratory practice program:

- demonstration tasks are to illustrate some fundamentals of computer networking and to introduce some simple networking software tools, however
- individual tasks are to check basic the usage skills of introduced software tools.

A technical report should be submitted based on the results of individual tasks.

The submisson page is in KJK Moodle https://edu.kozlek.bme.hu/?lang=en in the course material, in section titled "Introduction to vehicle communication" under topic "Lab1 Technical Report Submission", a Report template is available there, as well.

1 Introduction to laboratory environment

a. Creation, deployment and access to a student VM
   i. Start a web browser
   ii. Open page: https://cloud.bme.hu/
   iii. Select Füred Cloud fured.cloud.bme.hu cloud infrastructure
   iv. Log in with your BUTE Central Login identifier
   v. Create a VM based on Template VWS-AutNetCommSys2020-edu v3.1
   vi. Connect to your VM using Windows Remote Desktop application
      - Host name, user name and password are available on the VM's dashboard
b. Launch Windows Command Prompt application and test some commands (dir, cd, cls, hostname, time, etc.)[1][2]

[1] https://www.thomas-krenn.com/en/wiki/Cmd_commands_under_Windows
[2] https://commandwindows.com/command3.htm

2 The big picture and some local details: Getting familiar with some basic network setting of a host

Usage and results of some console application (Use Windows Command Prompt application to execute the commands below).

a. Command: **`ipconfig`**[3]
   i. Switch: /all Results: IP address, MAC address, DNS server, etc.
   ii. Switch: /displaydns Results: local DNS cache
   iii. Switch: /flushdns (requires admin privileges, launch Command Prompt application as administrator) Result: local DNS cache content is deleted

c. Command: **`getmac`**[4] Result: the media access control (MAC) address and list of network protocols associated with each address for all network cards

d. Command: **`route`**[5]
   i. Switches: print -4 Result: the entries in the local IPv4 routing table

e. Command: **`arp`**[6]
   i. Switch: -a Result: entries in the Address Resolution Protocol (ARP) cache
   ii. Switches: -d * (requires admin privileges, launch Command Prompt application as administrator) Result: local ARP cache content is deleted

f. Command: **`ping`**[7] Result: IP level connectivity to a host, (RTT, packet loss rate)

g. Command: **`traceroute`**[8] Result: path taken to a destination (list of hops, RTT by hop)
   i. Switch: -d Result: list of hops without reverse DNS resolution (IP address only, no host name)

h. Command: **`netstat`**[9]
   i. Switch: -a Result: list of all active TCP connections and the TCP and UDP ports on which the computer is listening
   ii. Switch: -n Result: list of active TCP connections, however, addresses and port numbers are expressed numerically and no attempt is made to determine names
   iii. Switch: -e Result: Ethernet statistics, such as the number of bytes and packets sent and received
   iv. Switch: -s Result: statistics by protocol. By default: TCP, UDP, ICMP, and IP protocols
   v. Switches (combined) -e -s Result: see switches -e and -s above
   vi. Switch: -r Result: the contents of the IP routing table (equivalent to the route print command)

i. Command: **`nslookup`**[10] Name resolver and DNS diagnostic tool
   i. Switch: -type=ns Result: DNS name server for the named zone
   ii. Switch: -query=mx Result: mail server
   iii. nslookup 152.66.115.203 Result: host name (reverse DNS usage)

[3] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/ipconfig
[4] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/getmac
[5] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/route_ws2008
[6] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/arp
[7] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/ping
[8] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/tracert
[9] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/netstat
[10] https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/nslookup

Note: A more complex and function reach DNS resolution tool (dig) is available on the VM, as well.
Usage: dig {switches} @<name server IP address> <host name> e.g. from Fured Cloud:
dig @152.66.54.121 www.polimi.it

Hosts for **`ping`** and **`tarcert`**: (Hint: start with tracert -d, and, based on the hop RTT-s guess long links, e.g. submarine cable, than remove switch -d, and check your guesses according to the provided names)

www.szotar.sztaki.hu
www.polimi.it (Milan, Italy) //no echo reply for ping,
//check the web server availability with a browser
www.uni-ulm.de (Ulm, Germany)
bangaloreuniversity.ac.in (India)
www.clemson.edu (USA)
www.cisco.com (USA?)
www.uba.ar (Argentina)
sydney.edu.au (Sydney, Australia)
www.kobe-u.ac.jp (Kobe, Japan)
www.knu.ac.kr (Degu, South Korea)
www.uva.edu (Charlottesville, USA)
www.arstechnica.com in MS cloud Azur- (cloudapp.net)
http://www.manoa.hawaii.edu/
http://www5.usp.br/
https://www.knu.ac.kr/wbbs/
http://www.copa2014.gov.br/

3 Some Web pages with extended services

Some public web pages provide interesting and useful extension of ping and traceroute tools. Repeat some already performed **`ping`** and **`tracert`** tests!

a. Ping extensions.
   i. Fast Remote Ping Tool: https://www.wormly.com/test-remote-ping
   ii. Global Ping Statistics, ping times between WonderNetwork servers around the world: https://wondernetwork.com/pings
   iii. Geolocation: https://www.ipfingerprints.com/
   iv. Access to a given host from different locations all around the word: https://wheresitup.com/demo
b. Traceroute extension.
   i. Visual Traceroute: http://en.dnstools.ch/visual-traceroute.html

4 A locally (on your VM) installed graphic tool: PingPlotter

Repeat some already performed ping tests for distant server locations! Analyze and comment the RTT variations.

5 Protocol layers, and demonstrations for encapsulation/decapsulation

a. Application layer: **`http`** messages
   i. A web server is available (in Fured Cloud), a simple text file and a directory are published
      - VWS hfs web server IP address: **10.9.1.97**
      - published file: **test1.txt**

- published directory: **Pub_WebSever/**

ii. Start a Firefox browser on your VM, and connect to the web server.
- How to turn to a web server, if it does not have a name? (Why is there no name for our web server?)
- Connect to the web server by its IP address: **http://10.9.1.97**

Comment: There is no DNS server inside the cloud network, therefore, we connect the web server directly using its IP address.

- What has been downloaded?

Comment: There is no index.html available the hfs webserver sends something instead)

iii. Use the Firefox development support (F12): to see http messages
- download the available file: **http://10.9.1.97/test1.txt**

Hint: First clear the already downloaded file from the cache of the browser.

- http messages (network)
- message header (select message, and select Header tab)
- generate some response status codes
  a. Reload the file: 304 not modified (displayed from local cache of the browser)
  b. Modify the file name (e.g. test**2**.txt , and try to download the file: 404 file not found

b. Application and Transport layers: `http tcp` transport (using protocol analyzer WireShark)
i. Start WireShark on your VM (DO NOT UPDATE!)
ii. Select the Ethernet interface
iii. Start capture
iv. Generate http traffic by downloading the file (not from local cache!)
v. Stop capture
vi. Set display filter: http
vii. Analyze the captured packets
viii. Delete the captured PDU-s

c. Application and transport layer : `dns udp` transport (using protocol analyzer WireShark)
i. Start capture (WireShark already running, select *Continue without Saving* option)
ii. Generate udp traffic with nslookup (dns query udp)
iii. Stop capture
iv. Set display filter: dns
v. Analyze the captured packets
vi. Delate the captured PDU-s

d. Full protocol stack: http tcp, ip, Ethernet
i. Repeat the above task b.
ii. Extend the packet analysis for the entire protocol stack.

6 Simple security demonstrations

a. Start putty application on your VM

b. `telnet` to remote Linux host
   - host IP address: **10.9.0.53**
   - use one of the accounts below:

     Usr: user1 Pwd: PassW1
     Usr: user2 Pwd: PassW2
     Usr: user3 Pwd: PassW3
     Usr: user4 Pwd: PassW4
   - execute some basic commands (e.g. ls, pwd)
   - close the connection

c. `ssh` to remote Linux host
   - host IP address: **10.9.0.53**
   - use one of the accounts below:

     Usr: user1 Pwd: PassW1
     Usr: user2 Pwd: PassW2
     Usr: user3 Pwd: PassW3
     Usr: user4 Pwd: PassW4
   - execute some basic commands (e.g. ls, pwd)
   - close the connection

d. Encryption - cleartext: Analyse `telnet`[11] RFC854[12] protocol traffic with Wireshark
   i. Start capture (WireShark already running, select Continue without Saving option)
   ii. Generate telnet protocol traffic connecting to the remote host (above subtask b.)
       Hint: paste your password instead of typing
   iii. Stop capture
   iv. Set display filtere to telnet (or tcp.port==23)
   v. Find your password int he captured packets.

e. Encryption - cyphertext: Analyse `ssh`[13] RFC4251[14] protocol traffic with Wireshark
   i. Start capture (WireShark already running, select Continue without Saving option)
   ii. Generate ssh protocol traffic connecting to the remote host (above subtask c.)
       Hint: paste your password instead of typing
   iii. Stop capture
   iv. Set display filter to ssh (or tcp.port==22)
   v. Find differences comparing the connection initiation with telnet (Guess: e.g. Key exchange)
   vi. Find the last cleartext message (Guess: Ubuntu banner from server.)

f. Encryption: `telnet` and `ssh` once more: executing a command on remote host, resulting some readable text result (e.g. ls, or less <prepared text file>, exit less with q)

   i. Prepare one telnet and one ssh connection to the remote Linux server (with two putty applications) in parallel
   ii. Start capture (WireShark already running, select Continue without Saving option)
   iii. Execute a simple command both in telnet and ssh terminal windows
   iv. Stop capture
   v. First analyze telnet protocol traffic: find the command and the result
   vi. Then analyze ssh protocol traffic
   vii. Close Wireshark

g. `Ping` your neighbor's VM

[11] https://regi.tankonyvtar.hu/hu/tartalom/tkt/operacios-rendszerek/ch06s03.html
[12] https://tools.ietf.org/html/rfc854
[13] http://support.suso.com/supki/SSH_Tutorial_for_Linux
[14] https://tools.ietf.org/html/rfc4251

iv. No answer, all packets lost
v. Default setup of Win10 Firewall filters ICMP traffic
vi. Modify Firewall settings:
    - Settings/Update and Security/Windows Security/Firewall and Network Security, -> Advanced settings.
    - Find Inbound Rules, and enable (right click) File and Printer Sharing (Echo Request ICMPv4-In) rule both for private and public networks. Set the corresponding rule if you prefer to ping IPv6 address, as well.
vii. Repeat the ping.

7 WiFi demonstration (projected from instructor's laptop only)

Unfortunately, for security reasons WiFi connections are disabled on lab laptops, therefore the wifi demonstration can be performed on the instructor's laptop only.

8 Animations illustrating fundamental concepts of computer networking[15]

Public java applet available on web:

- Transmission versus Propagation Delay Applet
- Queuing and Loss Applet
- Message Segmentation
- HTTP Delay Estimation
- Recursive/Iterative Queries in DNS
- Go-Back-N Protocol
- Selective Repeat Protocol
- Flow Control
- TCP Congestion Control - Fairenss
- CSMA/CD
- 802.11 CSMA/CA WITHOUT Hidden Terminals
- 802.11 CSMA/CA WITH Hidden Terminals

9 Summary

Please, provide a Technical Report on the performed Tasks (see template separately).

10 Quiz - Lab1 closing test

The purpose of the quiz is to check the learning outcomes of the guided laboratory practice. Some bonus points can be collected to improve the Interim Exam 1 mark (details are specified soon).

Available in KJK Moodle: https://edu.kozlek.bme.hu/?lang=en in the course material, under topic "Introduction to vehicle communication", open from 18:45 (password required, and will be provided in time).

[15] Kurose, Ross Computer Networking, 8th Edition, Student resources