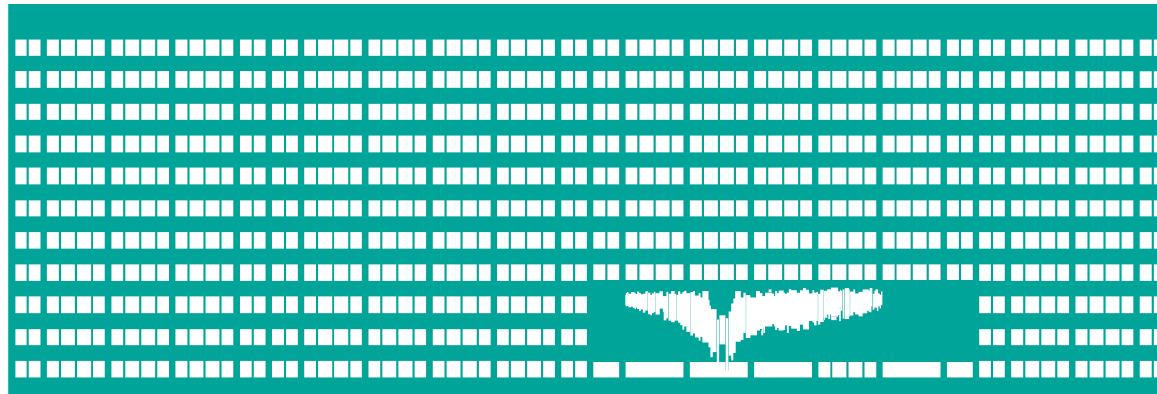


# Getting known the laboratory



## Computer networks Seminar 1

# Structured cabling

- Network sockets:
  - School network – TUN sockets
  - Laboratory – sockets R1-4 (leading to the corresponding rack)
- Computer network cards
  - Integrated on the motherboard (by default school network) – eth1 / eth0 for our tasks
- A label with MAC addresses, school IP address, computer name and domain name for DNS

# Operating systems

- Ubuntu
  - Common user: **cnap** (password: **cisco**)
  - Superuser: change with command **su** (password: **cisco**)
- MS Windows
  - Only common user – is allowed to change the IP address

# Ubuntu 18.04

- Internet browser **Firefox**
- File manager **mc**
- Editors
  - **mcedit** – part of mc
  - nano
  - **vim (gvim)**
  - **gedit**
- Serial communication – **minicom**
- **telnet, ssh, ftp, ...**

# Ubuntu

- **All locally saved files are temporary, they will be lost after restart!**
- Possible to connect USB drive
  - Automatic connection
  - manual: `mount -t <fs> <device> <location>`
- Remote copying with **scp**:
  - **scp** *<source>* *<target>*
  - *<source/target>*: local or remote file  
`user@machine:path`
    - for example:  
`scp ./test.txt abc123@home1:~`  
`scp abc123@home1:~/test.txt .`

# Manual settings of network parameters

- Linux (**requires superuser - root**)
  - **ifconfig** [*ethX* [*<address>* **netmask** *<mask>*]]
  - **ip addr add** *<ip address/mask>* **dev** *ethX*
  - **route add default gw** *<gateway>*
  - DNS servers and default domain set in **/etc/resolv.conf**
    - Commands: **nameserver A.B.C.D ; search <domain>**
- *Windows*
  - Settings → Control Panel → Networks connection → *eth0* → Properties → Protocol TCP/IP → Properties
    - Choose use the following IP address, fill **address**, **mask**, **gateway**, addresses of **DNS** servers
  - To display the settings: **ipconfig** (or **ipconfig /all**), or Settings → Control Panel → Networks connection → *eth0* → Status → Support (+Details)

# Connectivity

- **ping** – check the accessibility of target device
  - **Linux:**  
ping [-c <count>] [-i <pause>] [-f] [-s <size>] <target>
  - **Windows:**  
ping [-t] [-n <count>] [-l <size>] [-i <ttl>] <target>
- **traceroute** – path to target device
  - **Linux:** traceroute [-m <max. of hops>] <target>
  - **Windows:** tracert [-h <max. of hops>] <target>
- **wireshark (ethereal)**
  - List of received packets on network interface
- **dig/nslookup** – DNS lookup utilities
- **dhclient** – acquire IP address from DHCP serv.
  - **Linux:** dhclient <ethX>

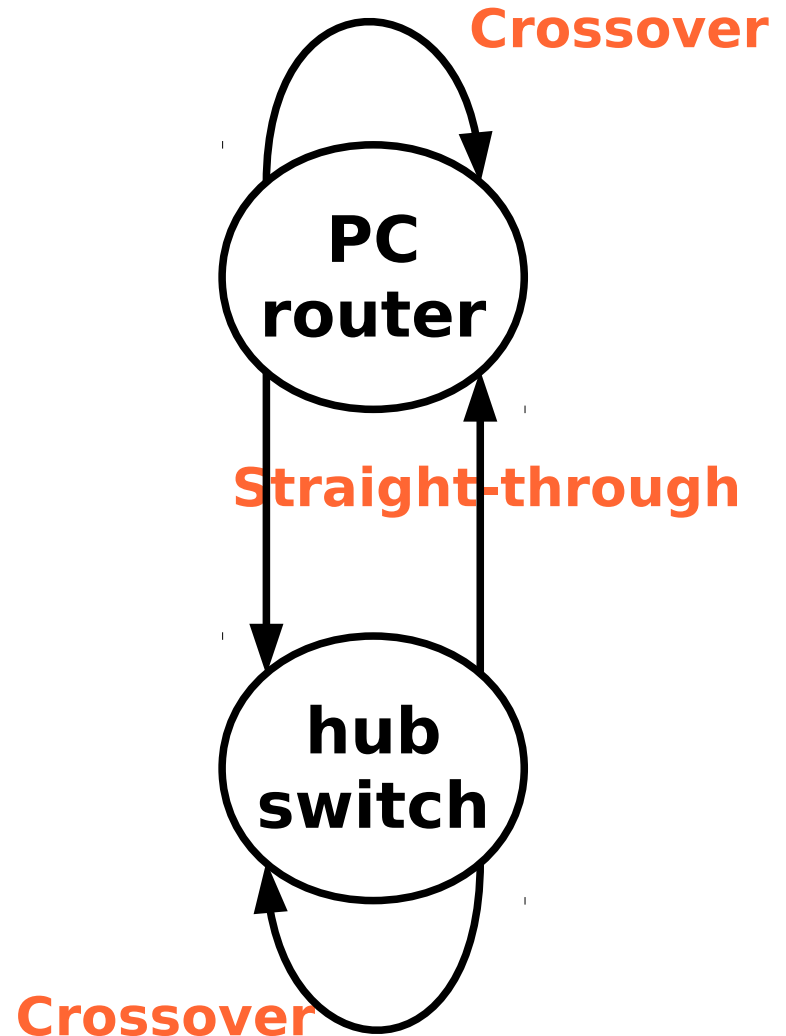
# Building the network with the hub

- Network card
  - Network PCMCIA card
- **Hub** (rozbočovač) – serves to distribute – broadcast the signal in network but it doesn't care about the transmitted data



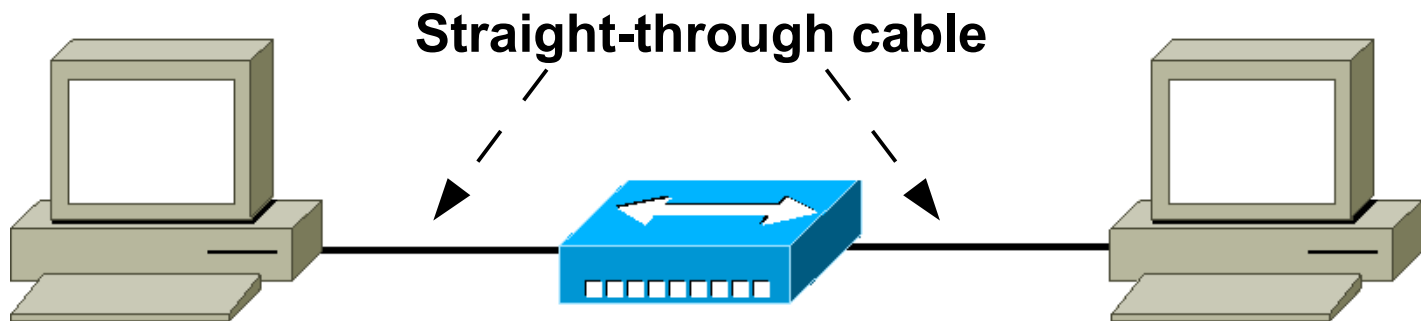
# Cables usage

- Crossover
  - PC - PC
  - Switch - Switch
  - Hub - Hub
  - Router - Router
  - Router - PC
- Straight-through
  - PC - Switch (Hub)
  - Router - Switch (Hub)



# Practical task

- Description: Connecting two computers using hub or switch



# Traffic analysis

- Wireshark
  - <http://www.wireshark.org>
  - Full graphic interface
  - Cross-platform program (Linux, Windows, Solaris, ...)
- Tcpdump
  - <http://www.tcpdump.org>

# Wireshark

The screenshot displays the Wireshark interface with a list of 18 captured packets. The first packet is selected, and its details are shown in the lower pane. The packet list includes various protocols such as TCP, DNS, ARP, and HTTP. The details pane for the first packet shows the Ethernet II, Internet Protocol, and Transmission Control Protocol layers.

No. -	Time	Source	Destination	Protocol	Info
1	0.000000	158.196.68.123	212.65.244.178	TCP	1204 > http [FIN, ACK] Seq=0 Ack=0 win=65075 Len=0
2	0.013700	212.65.244.178	158.196.68.123	TCP	http > 1204 [ACK] Seq=0 Ack=1 win=6432 Len=0
3	12.938110	158.196.68.123	158.196.149.9	DNS	Standard query A www.avu.cz
4	12.970543	158.196.149.9	158.196.68.123	DNS	Standard query response CNAME apache.avu.cz A 195.113.80.126
5	12.971962	158.196.68.123	Broadcast	ARP	who has 195.113.80.126? Tell 158.196.68.123
6	12.972028	apache.avu.cz	158.196.68.123	ARP	195.113.80.126 is at 00:00:d1:1f:c4:0c
7	12.972043	158.196.68.123	195.113.80.126	TCP	1210 > http [SYN] Seq=0 Ack=0 win=65535 Len=0 MSS=1260
8	12.981896	195.113.80.126	158.196.68.123	TCP	http > 1210 [SYN, ACK] Seq=0 Ack=1 win=5840 Len=0 MSS=1460
9	12.981987	158.196.68.123	195.113.80.126	TCP	1210 > http [ACK] Seq=1 Ack=1 win=65535 Len=0
10	12.982428	158.196.68.123	195.113.80.126	HTTP	GET /technPrac/digLab.htm HTTP/1.1
11	12.995066	195.113.80.126	158.196.68.123	TCP	http > 1210 [ACK] Seq=1 Ack=594 win=6523 Len=0
12	13.039489	195.113.80.126	158.196.68.123	TCP	[TCP segment of a reassembled PDU]
13	13.041779	195.113.80.126	158.196.68.123	TCP	[TCP segment of a reassembled PDU]
14	13.041848	158.196.68.123	195.113.80.126	TCP	1210 > http [ACK] Seq=594 Ack=2521 win=65535 Len=0
15	13.059920	195.113.80.126	158.196.68.123	TCP	[TCP segment of a reassembled PDU]
16	13.060029	158.196.68.123	195.113.80.126	TCP	1210 > http [ACK] Seq=594 Ack=3781 win=65535 Len=0
17	13.062094	195.113.80.126	158.196.68.123	TCP	[TCP segment of a reassembled PDU]
18	13.064331	195.113.80.126	158.196.68.123	TCP	[TCP segment of a reassembled PDU]

Frame 1 (54 bytes on wire (54 bytes captured) on interface eth0):

- Ethernet II, Src: 158.196.68.123 (00:05:9a:3c:78:00), Dst: apache.avu.cz (00:00:d1:1f:c4:0c)
- Internet Protocol, Src: 158.196.68.123 (158.196.68.123), Dst: 212.65.244.178 (212.65.244.178)
- Transmission Control Protocol, Src Port: 1204 (1204), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0

```
0000  00 00 d1 1f c4 0c 00 05 9a 3c 78 00 08 00 45 00  .....<X...E.
0010  00 28 53 2a 40 00 80 06 fb 71 9e c4 44 7b d4 41  .(S*@...q.D{.A
0020  f4 b2 04 b4 00 50 30 9e ad 60 dc eb 18 ee 50 11  ....P0.....P.
0030  fe 33 2c 8f 00 00                                     .3,...
```

Frame (frame), 54 bytes | P: 101 D: 101 M: 0 Drops: 0