



Tracing



2N® Tracing & Wireshark

Quick guide

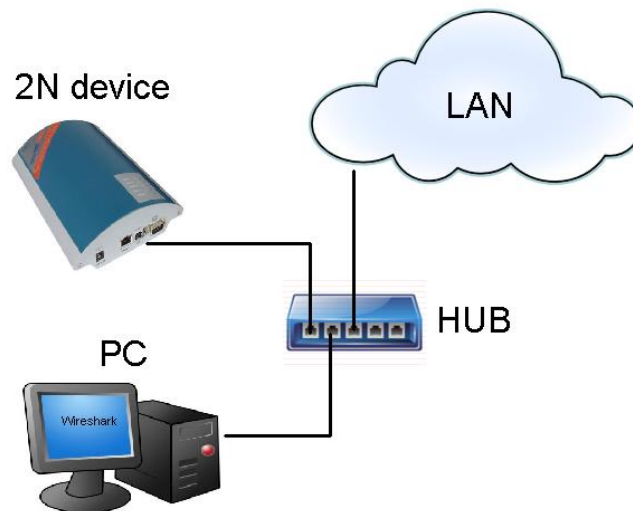
Version 1.00

www.2n.cz

Program Wireshark is used for tracing the communication between devices interconnected by LAN. The devices (e.g. PBX connected with a GSM gateway via LAN) send to each other packets which are captured by the above mentioned program. Wireshark is distributed under the Open source licence (<http://www.wireshark.org/download.html>).

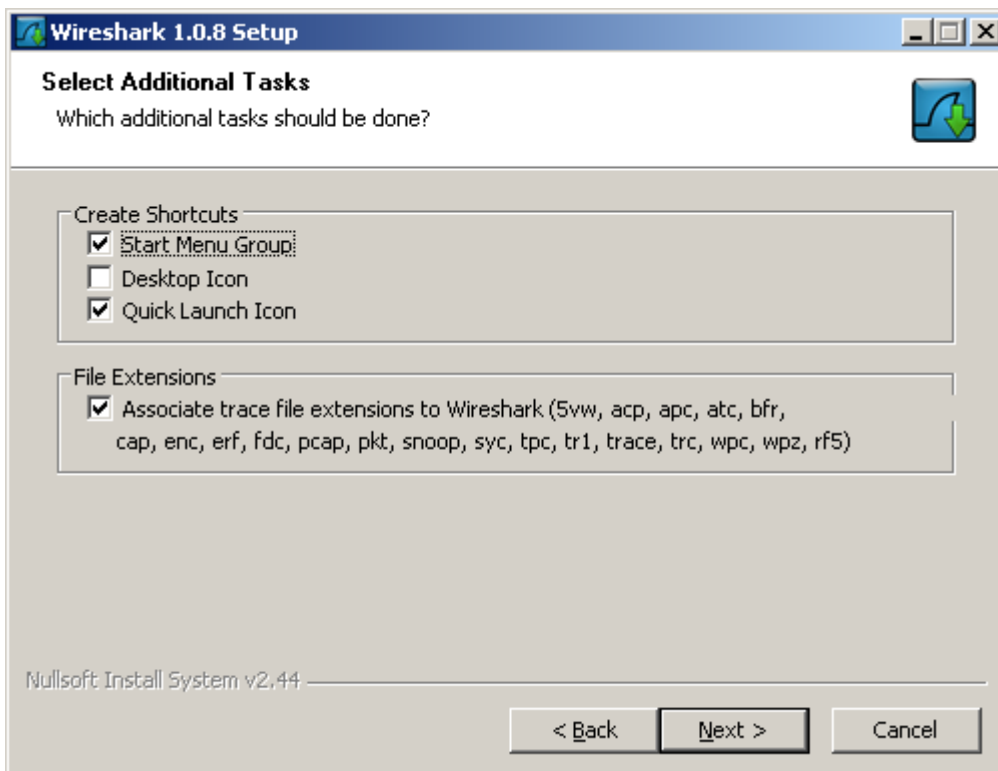
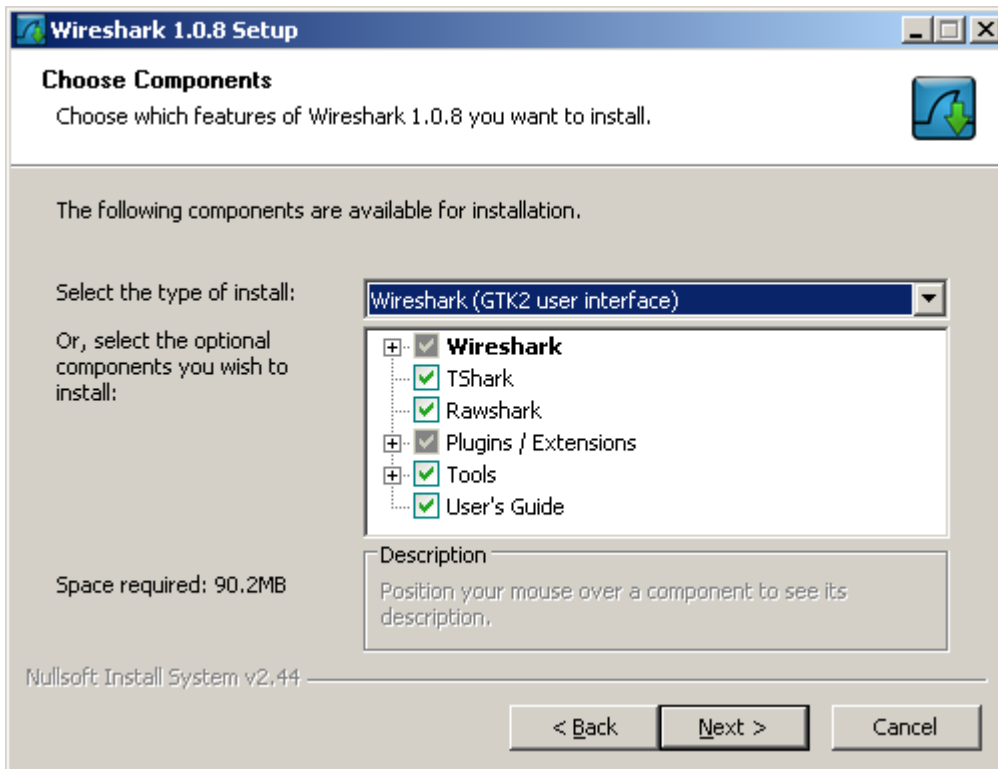
In order to capture all the packets which are sent via particular LAN the devices have to be part of the same segment (**they have to be connected via HUB**). If you do not have a hub you can use a switch which supports so called port mirroring (http://en.wikipedia.org/wiki/Port_mirroring).

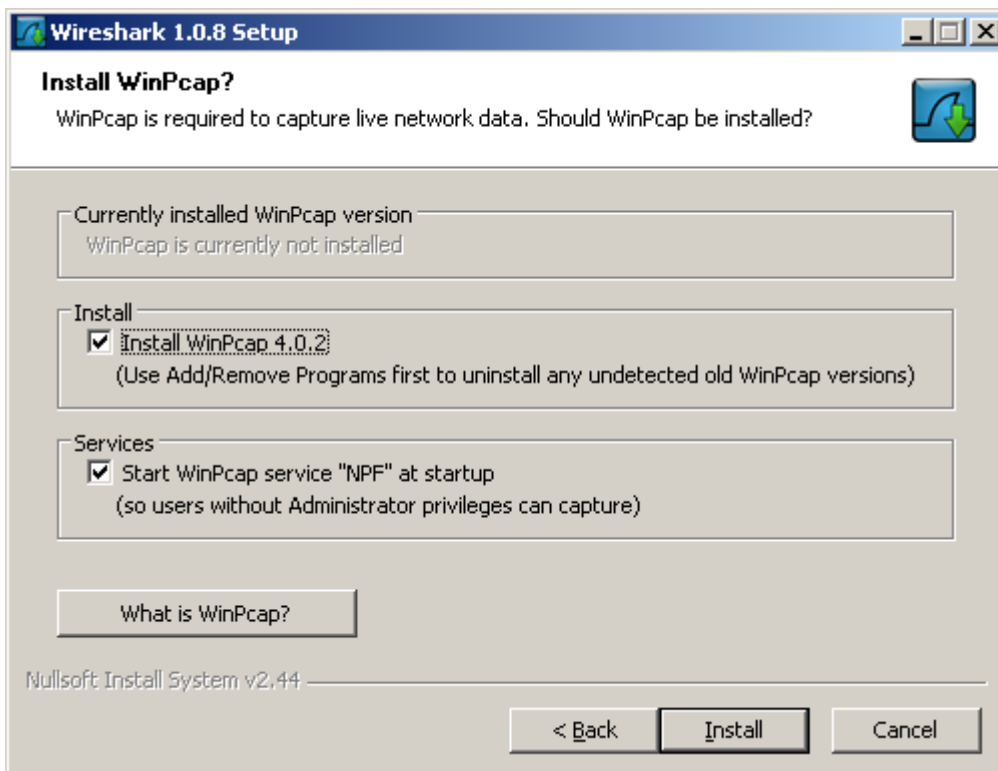
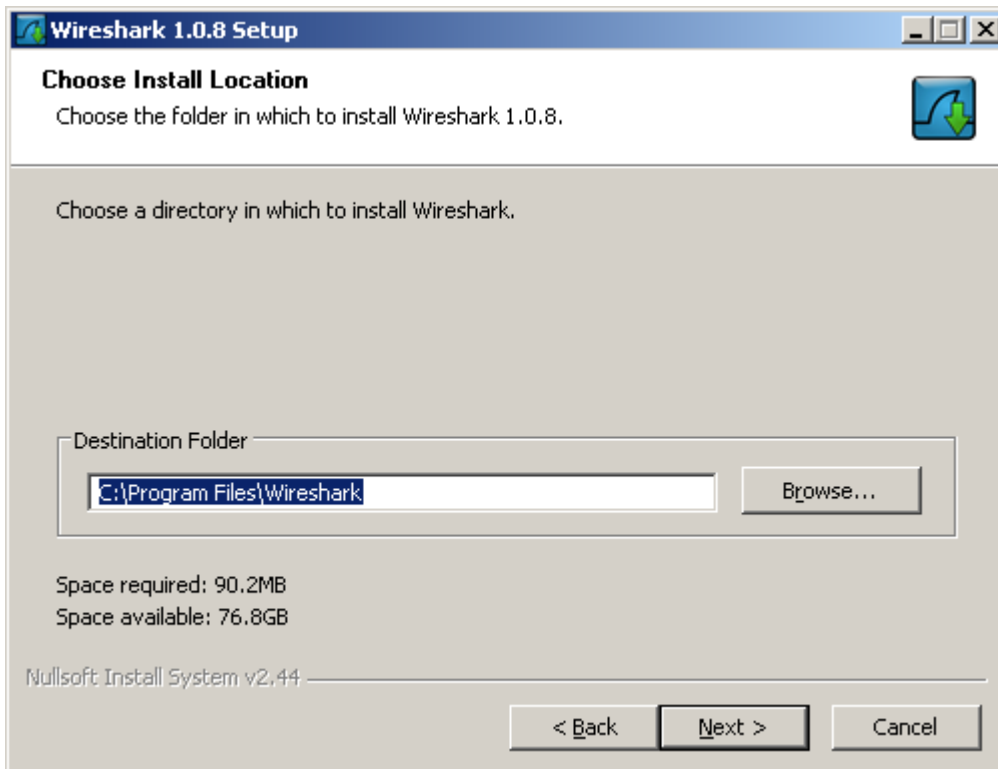
Here is the required scheme:

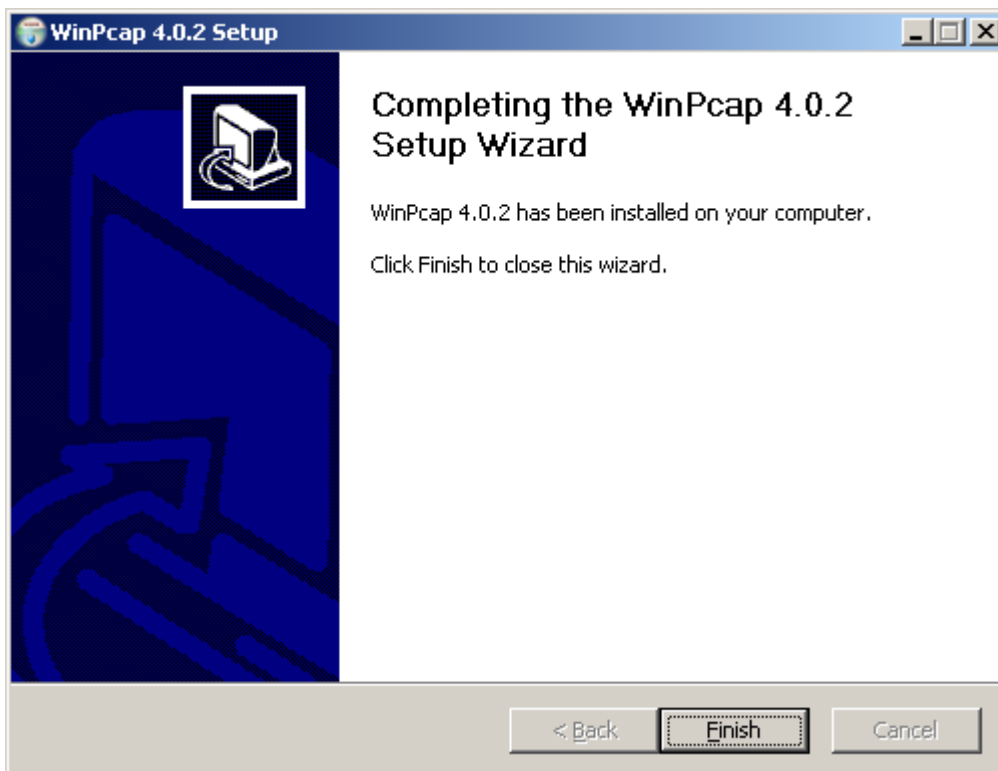
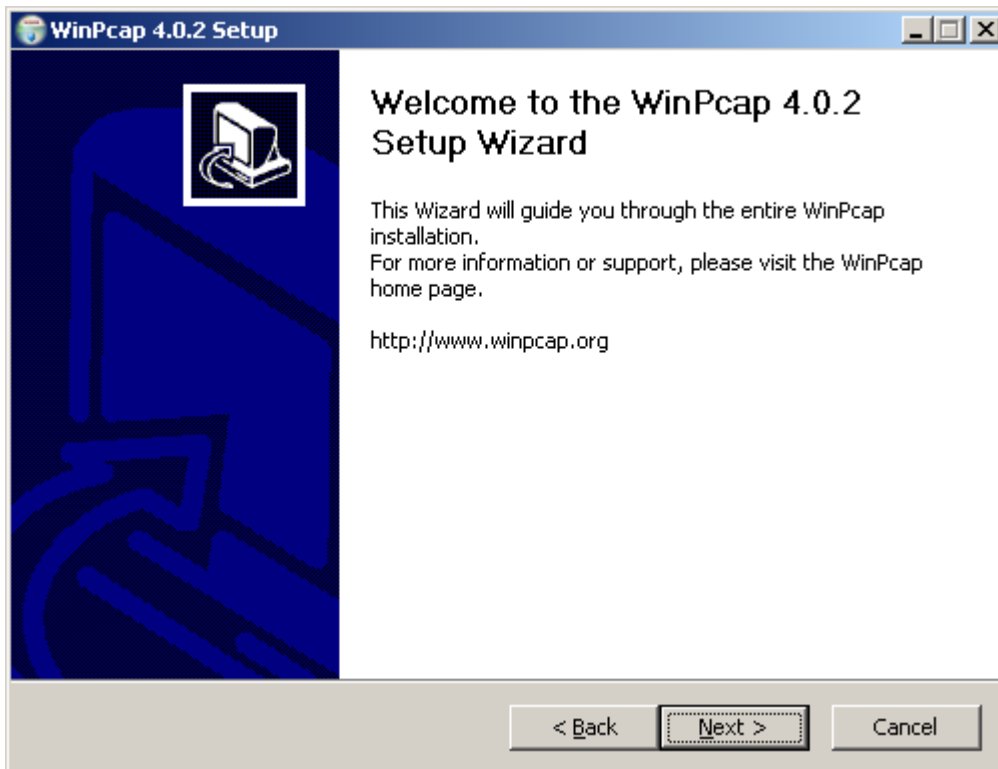


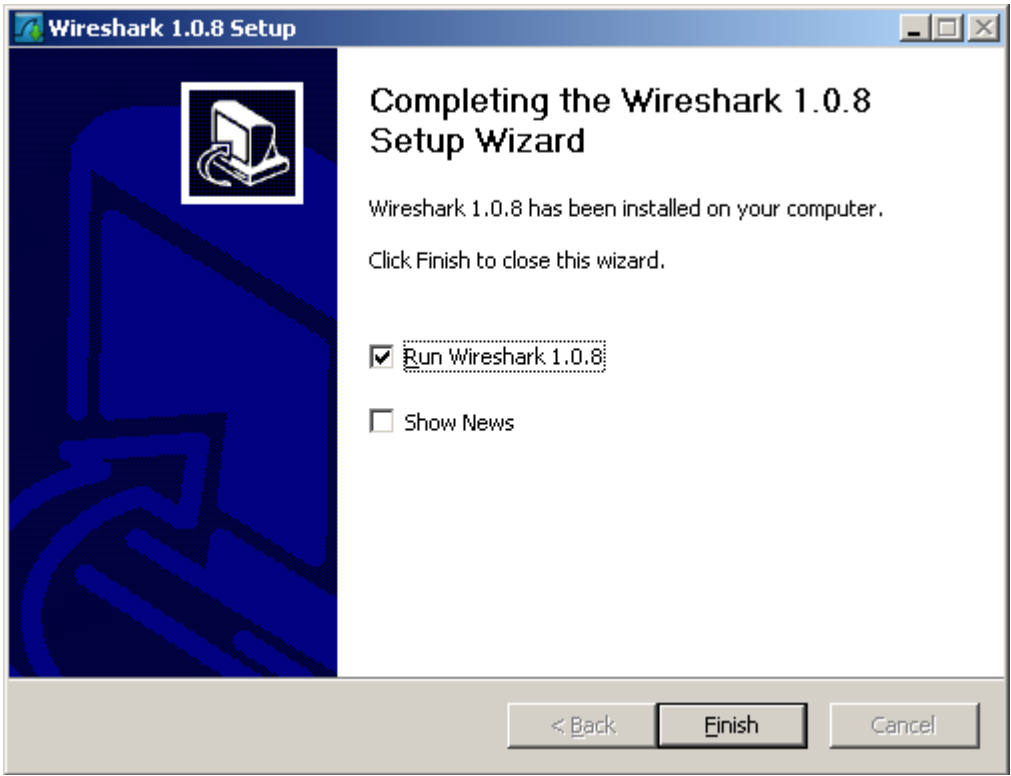
Installation: Download the program and appropriate installer (e.g. Windows installer).



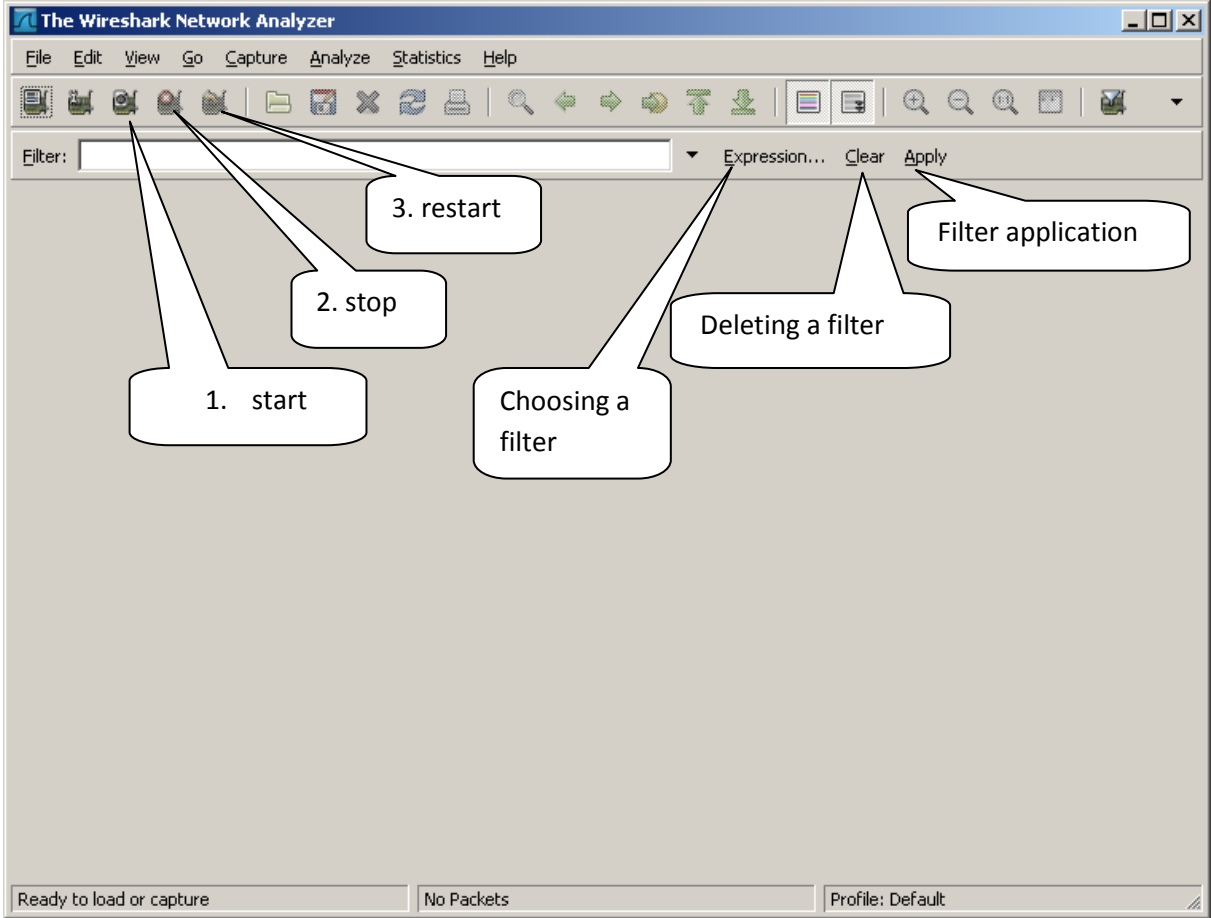








Wireshark is then initiated:



Start capturing: „Capture->Start” or use icon #1 End capturing: „Capture->Stop” or use icon #2
Restart capturing with deletion of previously captured packets: „Capture->Restart” or use icon #3

Save results: „File->Save As->Wireshark/tcpdump/... -libpcap(*.pcap;*.cap) Open previously saved trace: „File->Open” Basics of filter application:

- 1) I want to filter the packets according to source IP– eg: ip.src_host=="192.168.5.7"
- 2) I want to filter the packets according to destination IP – eg: ip.dst_host=="192.168.5.7"
- 3) Protocol type: tcp, udp, sip, ...
- 4) The terms can be combined via logical expressions : and, or, not, ...
- 5) Applying a filter – Apply, Deleting a filter – Clear

Example of captured communication:

The screenshot shows the Wireshark interface with a list of captured packets and a detailed view of a selected packet. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Info
57	5.773219	192.168.2.56	255.255.255.255	UDP	Source port: distinct
58	6.055340	192.168.2.178	239.255.255.250	SSDP	M-SEARCH * HTTP/1.
59	6.056268	192.168.2.128	239.255.255.250	SSDP	M-SEARCH * HTTP/1.
60	6.079625	HewlettP_4e:82:26	Broadcast	ARP	who has 192.168.1.
61	6.098099	192.168.2.139	192.168.3.255	NBNS	Name query NB SERV
62	6.098102	Kollmorg_f1:29:50	Broadcast	ARP	who has 192.168.2.
63	6.279981	192.168.1.22	192.168.3.255	NBNS	Name query NB DC2N

The packet details pane for the selected packet (No. 61) shows:

- Frame 61 (956 bytes on wire, 956 bytes captured)
- Ethernet II, Src: HwServer_00:bb:97 (00:0a:59:00:bb:97), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Internet Protocol, Src: 192.168.2.56 (192.168.2.56), Dst: 255.255.255.255 (255.255.255.255)
- User Datagram Protocol, Src Port: distinct (9999), Dst Port: distinct (9999)
- Data (956 bytes)

The hex dump pane shows the raw data of the packet, with a callout pointing to the hexadecimal value '00 0a 59 00 bb 97 00 45 00'.

Example of captured communication after applying a filter:

The screenshot shows the Wireshark interface with the following details:

- Filter: `ip.src_host=="192.168.2.56" and udp`
- Packet List Table:

No.	Time	Source	Destination	Protocol	Info
27	2.634493	192.168.2.56	255.255.255.255	UDP	Source port: distinct (9999)
36	3.703857	192.168.2.56	255.255.255.255	UDP	Source port: distinct (9999)
57	5.773219	192.168.2.56	255.255.255.255	UDP	Source port: distinct (9999)
- Packet Details (Frame 57):
 - Ethernet II, Src: Hwserver_00:bb:97 (00:0a:59:00:bb:97), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 - Internet Protocol, Src: 192.168.2.56 (192.168.2.56), Dst: 255.255.255.255 (255.255.255.255)
 - User Datagram Protocol, Src Port: distinct (9999), Dst Port: distinct (9999)
 - Data (914 bytes)
- Packet Bytes:

```
0000 ff ff ff ff ff ff 00 0a 59 00 bb 97 08 00 45 00 .....Y.....E.
0010 03 ae df e8 00 00 40 11 d4 76 c0 a8 02 38 ff ff .....@..v...8..
0020 ff ff 27 0f 27 0f 03 9a 22 fe 3c 3f 78 6d 6c 20 .....<?xml
0030 76 65 72 73 69 6f 6e 3d 22 31 2e 30 22 20 65 6e version="1.0" en
0040 63 6f 64 69 6e 67 3d 22 55 54 46 2d 38 22 3f 3e coding="UTF-8"?>
0050 0d 00 2c 64 65 76 60 62 65 20 0d 00 00 2c 61 70 </doc>
```

Important: In order to provide us with detailed information about the possible problem with the device please do not apply any filter and simply capture all the communication during the test call. Save it via **Save as>Wireshark/tcpdump/... -libpcap(*.pcap;*.cap)** and send it to us.



2N TELEKOMUNIKACE a.s.

Modřanská 621, 143 01 Praha 4
tel.: 261 301 111, fax: 261 301 999,
e-mail: sales@2n.cz
www.2n.cz