

LAB: Protocol & Packet Analysis with **wireshark** ... with some focus on IPv6



Sebastian Büttrich, NSRC

Last edit: March 2015

**ICTP Workshop on
Scientific Applications for the Internet of Things (IoT)**



These materials are licensed under the Creative Commons Attribution-NonCommercial 4.0 International license
(<http://creativecommons.org/licenses/by-nc/4.0/>)

Protocol & Packet Analysis

Motivation: why are we *sniffing packets*?

- Network analysis
- Problem identification
- Security auditing
- Statistics

Some terms

You will frequently find words like ...

- Dumping = capturing packets from interfaces and saving to file (called a dump)
- Sniffing = looking/listening for packets to dump

Packet dumping & sniffing – some tools

- **Tcpdump**, using libpcap (Linux)
- WinDump, using WinPcap (Windows)
- GUI interface: **wireshark** (Linux, Win, Mac)
- Wireshark without GUI: **TShark**
- Specifically wireless: **Kismet**

Where to capture packets

In order to capture relevant data,
we need to dump packets on network interfaces
that can see the traffic of interest -
e.g.

on a switch, a gateway, a router,
on wireless interfaces in monitor mode

What to capture

Your choice of interface and the software capabilities determine what you will be able to see:

- Don't expect to see wireless traffic on an ethernet (cable) interface
- Don't expect to see Bluetooth or 802.15.4 on an 802.11 interface, even though it is the same frequency

Some typical use cases

(from personal experience ... yours will be different)

- Finding Rogue Access Points (ARP, spoofing)
- Identifying sources of broadcast storms
(due to malware, virus)
- Troubleshooting 802.1x authentication problems,
radius, WPA2
- Identifying network device failure
- Identifying impact of unwanted usage
(e.g. Dropbox LAN Sync, Bittorrent, video streams)

wireshark



”Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

Originally named Ethereal, the project was renamed Wireshark in May 2006 due to trademark issues.” [wikipedia]

wireshark



wireshark_marconi_20150316-05.pcap [Wireshark 1.12.4 (Git Rev Unknown from unknown)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: **ipv6** Expression... Clear Apply Save

No.	vlan.id	Time	Source	Destination	Protocol	Length	Info
101916		174.837776	2a00:1450:4002:803::1	2001:760:2e0b:1728:51	TCP	86	80-56495 [FIN, ACK] Seq=15359 Ack=1355 Win=31360 Len=0 TSval=274402891 TSecr
101841		174.724264	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56492-80 [FIN, ACK] Seq=1354 Ack=15553 Win=41472 Len=0 TSval=1298862 TSecr
101840		174.724205	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56494-80 [FIN, ACK] Seq=1354 Ack=15032 Win=41472 Len=0 TSval=1298862 TSecr
101839		174.724165	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56491-80 [FIN, ACK] Seq=1354 Ack=13609 Win=38912 Len=0 TSval=1298862 TSecr
101838		174.724102	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56497-80 [FIN, ACK] Seq=1354 Ack=13596 Win=38912 Len=0 TSval=1298862 TSecr
101837		174.724002	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56495-80 [FIN, ACK] Seq=1354 Ack=15359 Win=38912 Len=0 TSval=1298862 TSecr
101785		174.635255	2607:f8b0:4002:c07::5	2001:760:2e0b:1728:51	TCP	86	[TCP Keep-Alive ACK] 80-50395 [ACK] Seq=330 Ack=511 Win=43776 Len=0 TSval=
101710		174.506137	2001:760:2e0b:1728:51	2607:f8b0:4002:c07::5	TCP	86	[TCP Keep-Alive] 50395-80 [ACK] Seq=510 Ack=330 Win=15872 Len=0 TSval=1298
101705		174.491058	fe80::d6ca:6dff:feaa::ff02::1	ff53:ffff::1	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:ffff from d4:c
101421		173.982128	2001:760:2e0b:1728:f5	ff02::1	ICMPv6	86	Neighbor Solicitation for fe80::d6ca:6dff:feaa:55d3 from 84:38:38:76:ec:05
101393		173.965850	2a01:111:f400:8000::2	2001:760:2e0b:1728:51	TCP	74	443-38253 [RST, ACK] Seq=4965 Ack=428 Win=155216 Len=0
101360		173.856052	2a00:1450:4002:803::1	2001:760:2e0b:1728:51	TCP	86	[TCP Keep-Alive ACK] 80-56494 [ACK] Seq=15032 Ack=1354 Win=31360 Len=0 TSv
101350		173.840765	2001:760:2e0b:1728:51	2a01:111:f400:8000::2	TCP	74	38253-443 [FIN, ACK] Seq=427 Ack=4965 Win=26112 Len=0
101348		173.840373	2a00:1450:4013:c00::b	2001:760:2e0b:1728:51	QUIC	104	CID: 0, Seq: 207
101317		173.789851	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	56499-80 [ACK] Seq=1355 Ack=11691 Win=29184 Len=0 TSval=1298628 TSecr=2743
101316		173.789802	2a00:1450:4002:803::1	2001:760:2e0b:1728:51	TCP	86	80-56499 [FIN, ACK] Seq=11690 Ack=1355 Win=31360 Len=0 TSval=274393338 TSe
101304		173.747298	2a00:1450:4016:804::2	2001:760:2e0b:1728:51	TCP	86	[TCP Keep-Alive ACK] 80-50179 [ACK] Seq=6374 Ack=2839 Win=34432 Len=0 TSva
101299		173.738115	2001:760:2e0b:1728:51	2a00:1450:4002:803::1	TCP	86	[TCP Keep-Alive] 56494-80 [ACK] Seq=1353 Ack=15032 Win=41472 Len=0 TSval=1
101297		173.731750	2a00:1450:4002:803::1	2001:760:2e0b:1728:51	TCP	86	[TCP Dup ACK 95571#1] 80-56499 [ACK] Seq=11690 Ack=1354 Win=31360 Len=0 TS

```

0000  d4 ca 6d aa 55 d3 18 3d  a2 56 18 d4 86 dd 60 00  ..m.U..= .V....
0010  00 00 00 20 06 40 20 01  07 60 2e 0b 17 28 05 15  ... .@. ....(
0020  2a a5 4b 06 66 13 2a 00  14 50 40 02 08 03 00 00  *.K.f.*. .P@....
0030  00 00 00 00 10 00 dc ae  00 50 c0 58 0d f3 40 24  .... .P.X..@$
0040  82 bd 80 11 00 51 c2 ea  00 00 01 01 08 0a 00 13  ....Q.. ....
0050  d1 ae 10 5b 7f 7a        ...[.z
  
```

Remote dumping & importing



Often, we would like to use wireshark GUI on dumps collected in remote places without GUI, e.g. via tcpdump:

```
$ tcpdump -i <interface> -s 65535 -w <some-file>
```

The output file can then be imported by wireshark.

wireshark and ...



- **IPv6 – yes (filter: ipv6)**
- **6Lowpan – yes (filter: 6lowpan)**
- **802.15.4 – yes (filter: wpan)**
- **Bluetooth - ...**

Bluetooth and wireshark



You can capture Bluetooth traffic to or from your machine on Linux in Wireshark with libpcap 0.9.6 and later, if the kernel includes the BlueZ Bluetooth stack; starting with the 2.4.6 kernel, the BlueZ stack was incorporated into the mainline kernel.

Note that Debian and Debian-derived derivatives call the libpcap package "libpcap-0.8"; this does *****NOT***** mean that all such systems use libpcap 0.8.

Debian and its derivatives continue to use the name "libpcap-0.8", even though newer versions' libpcap packages use newer versions of libpcap; for example, Wheezy's libpcap-0.8 package uses libpcap 1.3.0.

If it's supported, and if you have sufficient privileges to capture, there will be interfaces named bluetoothN for various values of N starting with 0. To passively capture Bluetooth traffic between other machines, you can use the Ubertooth USB device. There is currently no libpcap support for Ubertooth, so you can't capture with Wireshark.

However, there is a plugin for Kismet - look for "Kismet" on the "Getting Started" Ubertooth page - and it produces capture files that can be dissected with a Wireshark plugin.

wireshark LAB:



- **Installation**
- **Start**
- **Basic functionality**
- **Some exercises**
- **Your own use cases?**

wireshark - installation



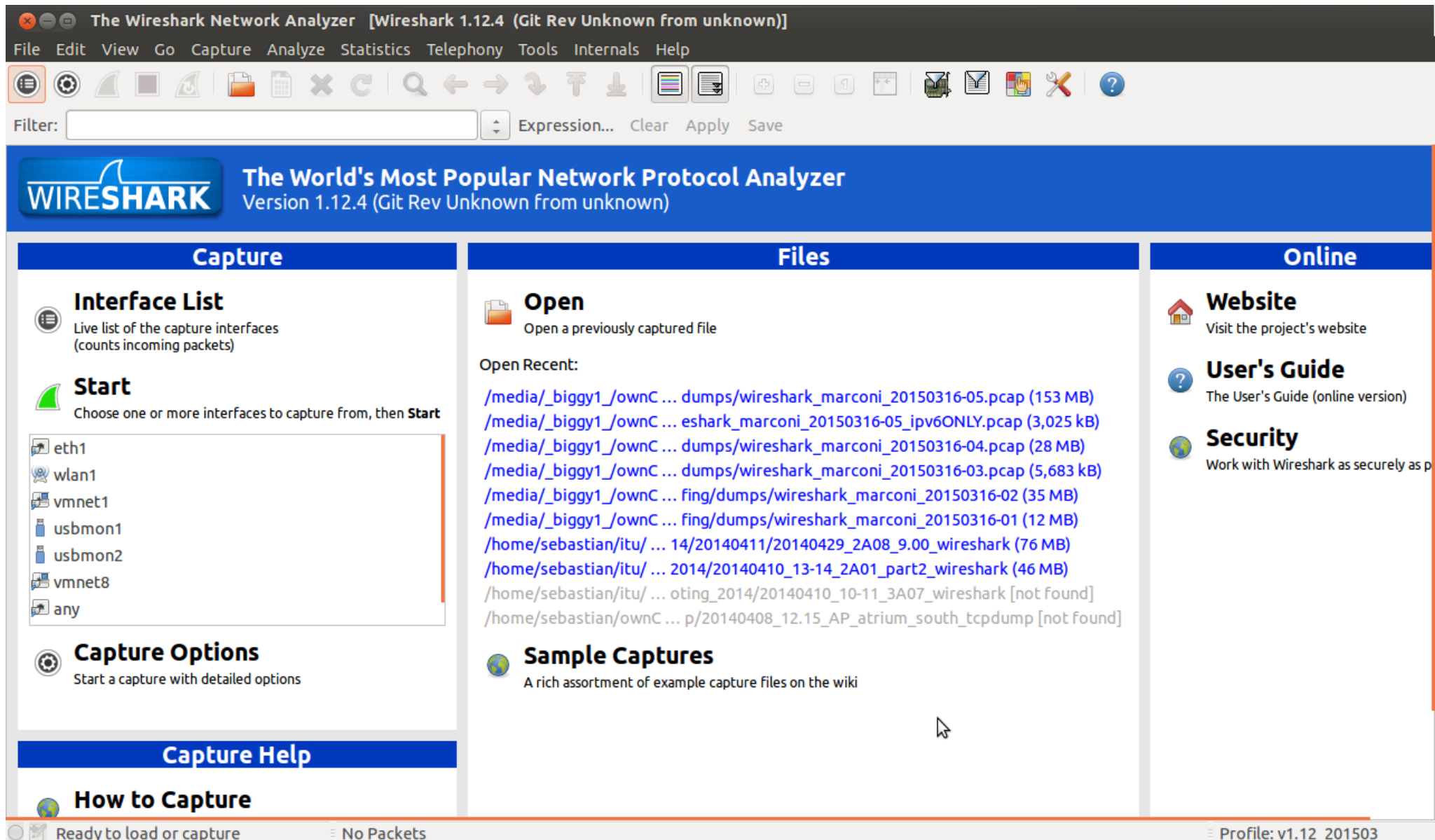
- <https://www.wireshark.org/download.html>
- **Linux: via repositories or build**

on Ubuntu: `$sudo apt-get install wireshark`

- **Windows: binary**
- **Mac: binary**

(note: needs X11, which is no longer part of OSX -
installation process will point you at XQuartz,
<http://xquartz.macosforge.org>)

wireshark - start



The screenshot shows the Wireshark 1.12.4 interface. The title bar reads "The Wireshark Network Analyzer [Wireshark 1.12.4 (Git Rev Unknown from unknown)]". The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, Internals, and Help. The toolbar contains icons for file operations, capture, and analysis. Below the toolbar is a filter bar with a text input and buttons for Expression..., Clear, Apply, and Save. The main interface is divided into three panes: Capture, Files, and Online. The Capture pane on the left includes an "Interface List" showing a live list of capture interfaces (counts incoming packets), a "Start" button with a description "Choose one or more interfaces to capture from, then Start", and "Capture Options" for starting a capture with detailed options. The Files pane in the center has an "Open" button to open a previously captured file, a list of "Open Recent" files with their paths and sizes, and "Sample Captures" linking to example capture files on the wiki. The Online pane on the right provides links to the "Website", "User's Guide" (online version), and "Security" information. At the bottom, a status bar shows "Ready to load or capture", "No Packets", and "Profile: v1.12_201503".

The Wireshark Network Analyzer [Wireshark 1.12.4 (Git Rev Unknown from unknown)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

WIRESHARK The World's Most Popular Network Protocol Analyzer
Version 1.12.4 (Git Rev Unknown from unknown)

Capture

Interface List
Live list of the capture interfaces (counts incoming packets)

Start
Choose one or more interfaces to capture from, then **Start**

- eth1
- wlan1
- vmnet1
- usbmon1
- usbmon2
- vmnet8
- any

Capture Options
Start a capture with detailed options

Capture Help

How to Capture

Files

Open
Open a previously captured file

Open Recent:

- /media/_biggy1/_ownC ... dumps/wireshark_marconi_20150316-05.pcap (153 MB)
- /media/_biggy1/_ownC ... eshark_marconi_20150316-05_ipv6ONLY.pcap (3,025 kB)
- /media/_biggy1/_ownC ... dumps/wireshark_marconi_20150316-04.pcap (28 MB)
- /media/_biggy1/_ownC ... dumps/wireshark_marconi_20150316-03.pcap (5,683 kB)
- /media/_biggy1/_ownC ... fing/dumps/wireshark_marconi_20150316-02 (35 MB)
- /media/_biggy1/_ownC ... fing/dumps/wireshark_marconi_20150316-01 (12 MB)
- /home/sebastian/itu/ ... 14/20140411/20140429_2A08_9.00_wireshark (76 MB)
- /home/sebastian/itu/ ... 2014/20140410_13-14_2A01_part2_wireshark (46 MB)
- /home/sebastian/itu/ ... oting_2014/20140410_10-11_3A07_wireshark [not found]
- /home/sebastian/ownC ... p/20140408_12.15_AP_atrium_south_tcpdump [not found]

Sample Captures
A rich assortment of example capture files on the wiki

Online

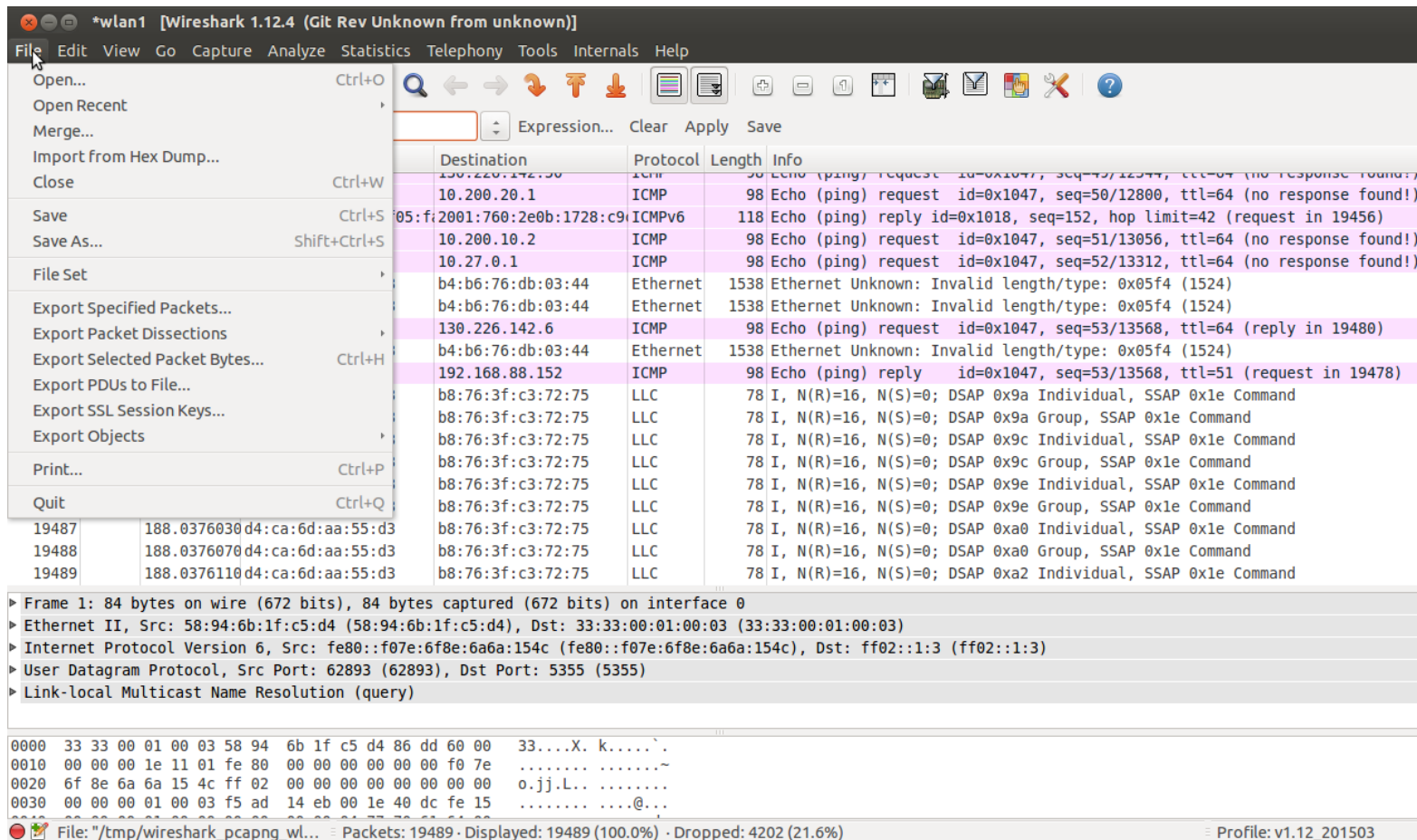
Website
Visit the project's website

User's Guide
The User's Guide (online version)

Security
Work with Wireshark as securely as possible

Ready to load or capture No Packets Profile: v1.12_201503

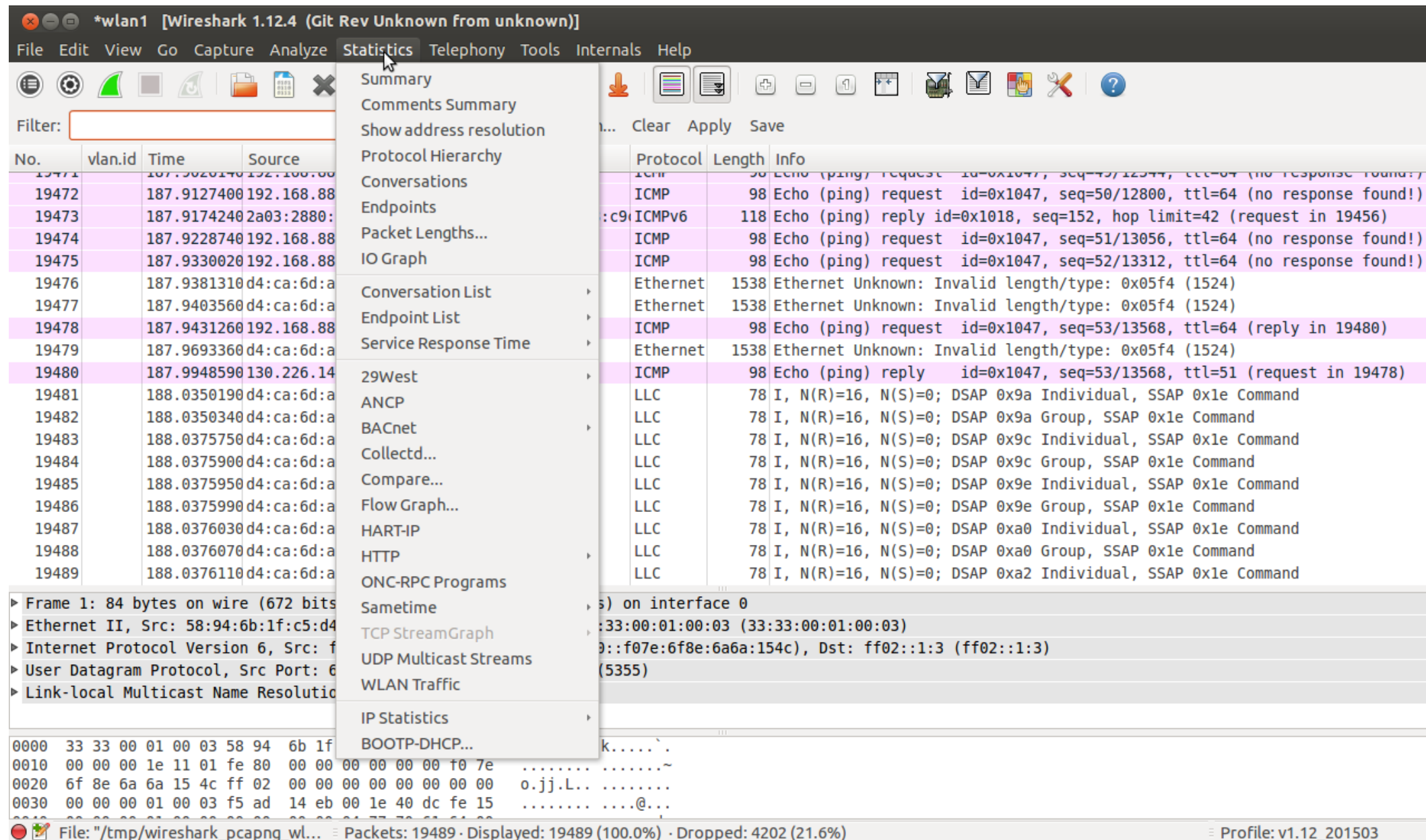
wireshark – file menu



- **Exercise 2: save your capture to file, and then open it again**
- **Exercise 3: find a 6Lowpan capture on**

<https://wiki.wireshark.org> and open it

wireshark – statistics



The image shows the Wireshark 1.12.4 interface with the Statistics menu open. The menu options are:

- Summary
- Comments Summary
- Show address resolution
- Protocol Hierarchy
- Conversations
- Endpoints
- Packet Lengths...
- IO Graph
- Conversation List
- Endpoint List
- Service Response Time
- 29West
- ANCP
- BACnet
- Collectd...
- Compare...
- Flow Graph...
- HART-IP
- HTTP
- ONC-RPC Programs
- Sametime
- TCP StreamGraph
- UDP Multicast Streams
- WLAN Traffic
- IP Statistics
- BOOTP-DHCP...

The packet list shows the following details:

No.	vlan.id	Time	Source
19471		187.9020140	192.168.88
19472		187.9127400	192.168.88
19473		187.9174240	2a03:2880:
19474		187.9228740	192.168.88
19475		187.9330020	192.168.88
19476		187.9381310	d4:ca:6d:a
19477		187.9403560	d4:ca:6d:a
19478		187.9431260	192.168.88
19479		187.9693360	d4:ca:6d:a
19480		187.9948590	130.226.14
19481		188.0350190	d4:ca:6d:a
19482		188.0350340	d4:ca:6d:a
19483		188.0375750	d4:ca:6d:a
19484		188.0375900	d4:ca:6d:a
19485		188.0375950	d4:ca:6d:a
19486		188.0375990	d4:ca:6d:a
19487		188.0376030	d4:ca:6d:a
19488		188.0376070	d4:ca:6d:a
19489		188.0376110	d4:ca:6d:a

The packet details pane shows the following information:

- Frame 1: 84 bytes on wire (672 bits)
- Ethernet II, Src: 58:94:6b:1f:c5:d4
- Internet Protocol Version 6, Src: f
- User Datagram Protocol, Src Port: 6
- Link-local Multicast Name Resolution

The packet bytes pane shows the following hex data:

```
0000 33 33 00 01 00 03 58 94 6b 1f
0010 00 00 00 1e 11 01 fe 80 00 00 00 00 00 f0 7e
0020 6f 8e 6a 6a 15 4c ff 02 00 00 00 00 00 00 00
0030 00 00 00 01 00 03 f5 ad 14 eb 00 1e 40 dc fe 15
```

The status bar shows: File: "/tmp/wireshark_pcapng_wl..." Packets: 19489 · Displayed: 19489 (100.0%) · Dropped: 4202 (21.6%) Profile: v1.12_201503

wireshark – statistics

Wireshark: Protocol Hierarchy Statistics

Display filter: none

Protocol	% Packets	Packets	% Bytes	Bytes	Mbit/s	End Packets	End Bytes
▼ Frame	100.00 %	19489	100.00 %	15980391	0.680	0	0
▼ Ethernet	100.00 %	19489	100.00 %	15980391	0.680	0	0
▼ Internet Protocol Version 6	41.05 %	8001	36.98 %	5910233	0.251	0	0
▼ User Datagram Protocol	1.96 %	382	0.59 %	94876	0.004	0	0
Domain Name Service	1.53 %	298	0.45 %	72074	0.003	298	0
Hypertext Transfer Protocol	0.38 %	74	0.13 %	21174	0.001	74	0
Mikrotik Neighbor Discovery Protocol	0.02 %	3	0.00 %	564	0.000	3	0
DHCPv6	0.04 %	7	0.01 %	1064	0.000	7	0
Internet Control Message Protocol v6	2.34 %	456	0.31 %	48980	0.002	456	0
▼ Transmission Control Protocol	36.75 %	7163	36.08 %	5766377	0.245	2854	10
Internet Message Access Protocol	0.06 %	11	0.01 %	1721	0.000	11	0
▼ Secure Sockets Layer	22.03 %	4294	29.47 %	4709332	0.200	4267	46
Secure Sockets Layer	0.14 %	27	0.16 %	26057	0.001	27	0
▼ Hypertext Transfer Protocol	0.02 %	4	0.02 %	3321	0.000	2	0
Online Certificate Status Protocol	0.01 %	2	0.01 %	1396	0.000	2	0
▼ Internet Protocol Version 4	23.33 %	4546	16.31 %	2606490	0.111	0	0
▼ User Datagram Protocol	5.07 %	989	1.16 %	185225	0.008	0	0
Domain Name Service	2.31 %	451	0.56 %	90032	0.004	451	0
Hypertext Transfer Protocol	0.61 %	119	0.20 %	31456	0.001	119	0
NetBIOS Name Service	1.14 %	222	0.13 %	20640	0.001	222	0
▼ NetBIOS Datagram Service	0.04 %	7	0.01 %	1623	0.000	0	0
▼ SMB (Server Message Block Protocol)	0.04 %	7	0.01 %	1623	0.000	0	0
▼ SMB MailSlot Protocol	0.04 %	7	0.01 %	1623	0.000	0	0

Help Close

- **Exercise 4: do a new capture on your interface, go to menu > statistics > protocol hierarchy – what is the ratio between IPv4 and IPv6 on your interface?**

.... % IPv4 <==> % IPv6

wireshark – filters & expressions

wireshark_marconi_20150316-05.pcap [Wireshark 1.12.4 (Git Rev Unknown from unknown)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: **ipv6** Expression... Clear Apply Save

No.	vlan.id	Time	Source	Destination	Protocol	Length	Info
12098		13.729092	fe80::d6ca:6d11:7eaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
12099		13.729433	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
12100		13.737089	2a00:1450:4013:c00::b	2001:760:2e0b:1728:51:1	TCP	86	[TCP ACKed unseen segment] 5228-35511 [ACK] Seq=1 Ack=2 Win=358 Len=0 TSval=1259761 TSecr=22172
12488		14.722550	2a00:1450:4013:c00::b	2001:760:2e0b:1728:51:1	QUIC	98	CID: 0, Seq: 183
12491		14.745627	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
12492		14.745943	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
12762		15.668528	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
12975		15.974368	fe80::cced:e193:1345::1	ff02::1:3	LLMNR	84	Standard query 0xecd0 A wpad
13000		16.079626	fe80::cced:e193:1345::1	ff02::1:3	LLMNR	84	Standard query 0xecd0 A wpad
14116		17.718492	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
14715		18.125208	2001:760:2e0b:1728:51:1	2607:f8b0:4000:80a::2	TLSv1.2	132	Application Data
14966		18.320158	2001:760:2e0b:1728:51:1	2607:f8b0:4000:80a::2	TLSv1.2	117	[TCP Previous segment not captured] Encrypted Alert
14967		18.320216	2001:760:2e0b:1728:51:1	2607:f8b0:4000:80a::2	TCP	86	51471-443 [FIN, ACK] Seq=124 Ack=62 Win=62 Len=0 TSval=1259761 TSecr=22172
15206		18.526702	2607:f8b0:4000:80a::2	2001:760:2e0b:1728:51:1	TCP	74	443-51471 [RST] Seq=62 Win=0 Len=0
15432		18.763560	fe80::d6ca:6dff:feaa::1	ff02::1:ff53:fffc	ICMPv6	86	Neighbor Solicitation for 2001:760:2e0b:1728:b4ae:ba57:c553:fffc from d4:ca:6d:aa:55:d3
15779		19.125893	2001:760:2e0b:1728:51:1	2a00:1450:4002:804::1	TLSv1.2	132	Application Data
15908		19.330630	2a00:1450:4002:804::1	2001:760:2e0b:1728:51:1	TCP	74	443-57099 [RST] Seq=62 Win=0 Len=0
15914		19.333073	2a00:1450:4002:804::1	2001:760:2e0b:1728:51:1	TCP	74	443-57099 [RST] Seq=62 Win=0 Len=0
15915		19.336892	2a00:1450:4002:804::1	2001:760:2e0b:1728:51:1	TCP	74	443-57099 [RST] Seq=62 Win=0 Len=0

▶ Frame 2: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)

▶ Ethernet II, Src: d4:ca:6d:aa:55:d3 (d4:ca:6d:aa:55:d3), Dst: 33:33:ff:53:ff:fc (33:33:ff:53:ff:fc)

▶ Internet Protocol Version 6, Src: fe80::d6ca:6dff:feaa::1 (fe80::d6ca:6dff:feaa::1), Dst: ff02::1:ff53:fffc (ff02::1:ff53:fffc)

▶ Internet Control Message Protocol v6

0000 33 33 ff 53 ff fc d4 ca 6d aa 55 d3 86 dd 60 00 33.S.... m.U...`.

0010 00 00 00 20 3a ff fe 80 00 00 00 00 00 d6 ca ... :...

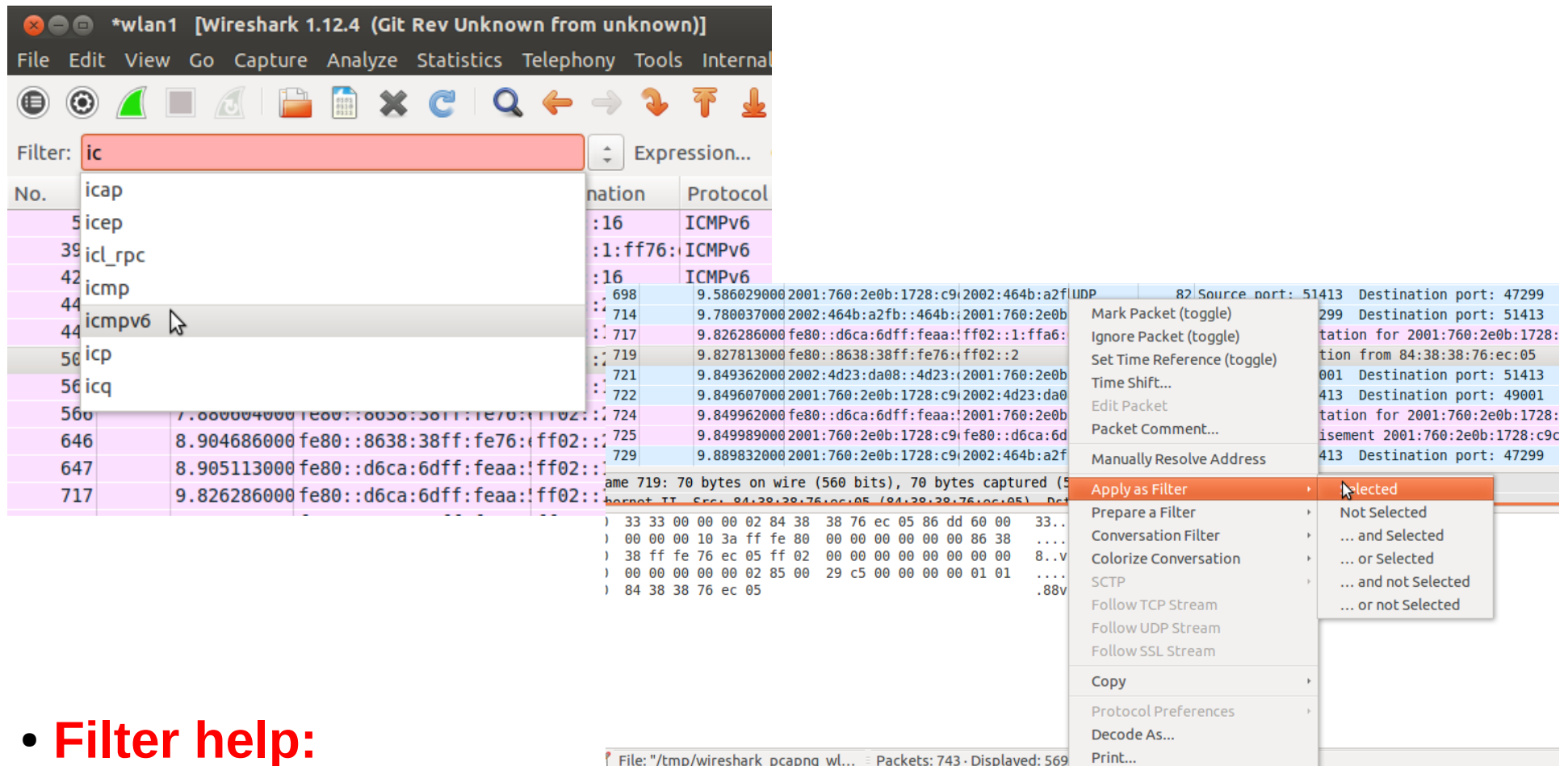
0020 6d ff fe aa 55 d3 ff 02 00 00 00 00 00 00 00 m...U...

0030 00 01 ff 53 ff fc 87 00 a8 51 00 00 00 00 20 01 ...S.... .Q....

File: "/media/_biggy1/_ownCloud/... Packets: 152751 · Displayed: 5788 (3.8%) · Load time: 0:02.794 Profile: v1.12_201503

- **Exercise 5: use Filter to show all IPv6 traffic**

wireshark – filters & expressions



The image shows the Wireshark 1.12.4 interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, and Internal. Below the menu bar is a toolbar with various icons. The Filter field is set to 'ic', and a dropdown menu is open showing a list of protocols: icap, icep, icl_rpc, icmp, icmpv6, icp, and icq. The packet list shows several packets, with packet 719 selected. A right-click context menu is open for packet 719, showing options like Mark Packet (toggle), Ignore Packet (toggle), Set Time Reference (toggle), Time Shift..., Edit Packet, Packet Comment..., Manually Resolve Address, Apply as Filter, Prepare a Filter, Conversation Filter, Colorize Conversation, SCTP, Follow TCP Stream, Follow UDP Stream, Follow SSL Stream, Copy, Protocol Preferences, Decode As..., and Print... The 'Apply as Filter' option is highlighted, and a sub-menu is open showing filter expressions: Selected, Not Selected, ... and Selected, ... or Selected, ... and not Selected, and ... or not Selected.

- **Filter help:**
 - there is autocompletion in the filter field,
 - right-clicking a packet gives you context filters

wireshark – filters & expressions

- **Exercise 6:**

leave wireshark for a moment and find facebook's ipv6 address – if they have one?

- ping6 or produce some other traffic to that address, capture and then find it in your wireshark capture

- Can you find any broadcast traffic in IPv6 packets?

- Move over to IPv4 – can you see broadcast traffic there?

Look at your protocols – what protocols create broadcast traffic?

wireshark – filters & expressions

- **Exercise 7:**

start a new dump and filter for ICMPv6 – what kinds of packets do you see?

.....
.....

- **Try to find a Router Advertisement – and look into it.**

What prefix is being advertised?

.....

Who sent this Advertisement?

.....

wireshark – additonal fun

Additonal exercises:

- Explore statistics possibilities -
what are your most active protocols, IPs, streams?
- Do a tcpdump on a (remote) command line and analyze it with wireshark
- Consider if and how wireshark would be useful in
 - our sensor deployment?
 - your own work?

Questions?



Thanks!

sebastian@nsrc.org

This image was originally posted to Flickr by hermanusbackpackers at <http://flickr.com/photos/36084059@N08/3343254977>.
It was reviewed on 25 September 2009 by the FlickreviewR robot and was confirmed to be licensed under the terms of the cc-by-2.0.