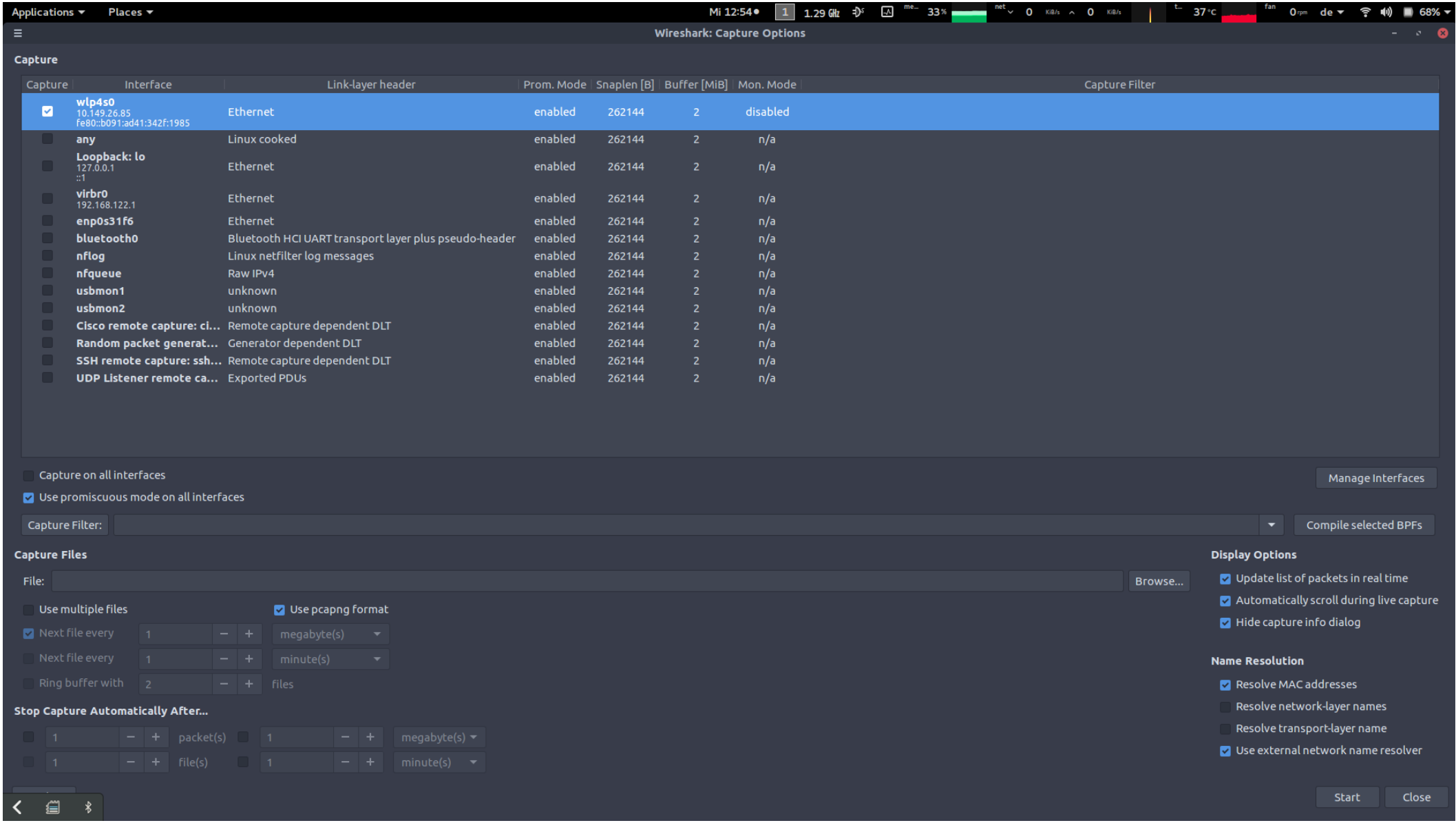
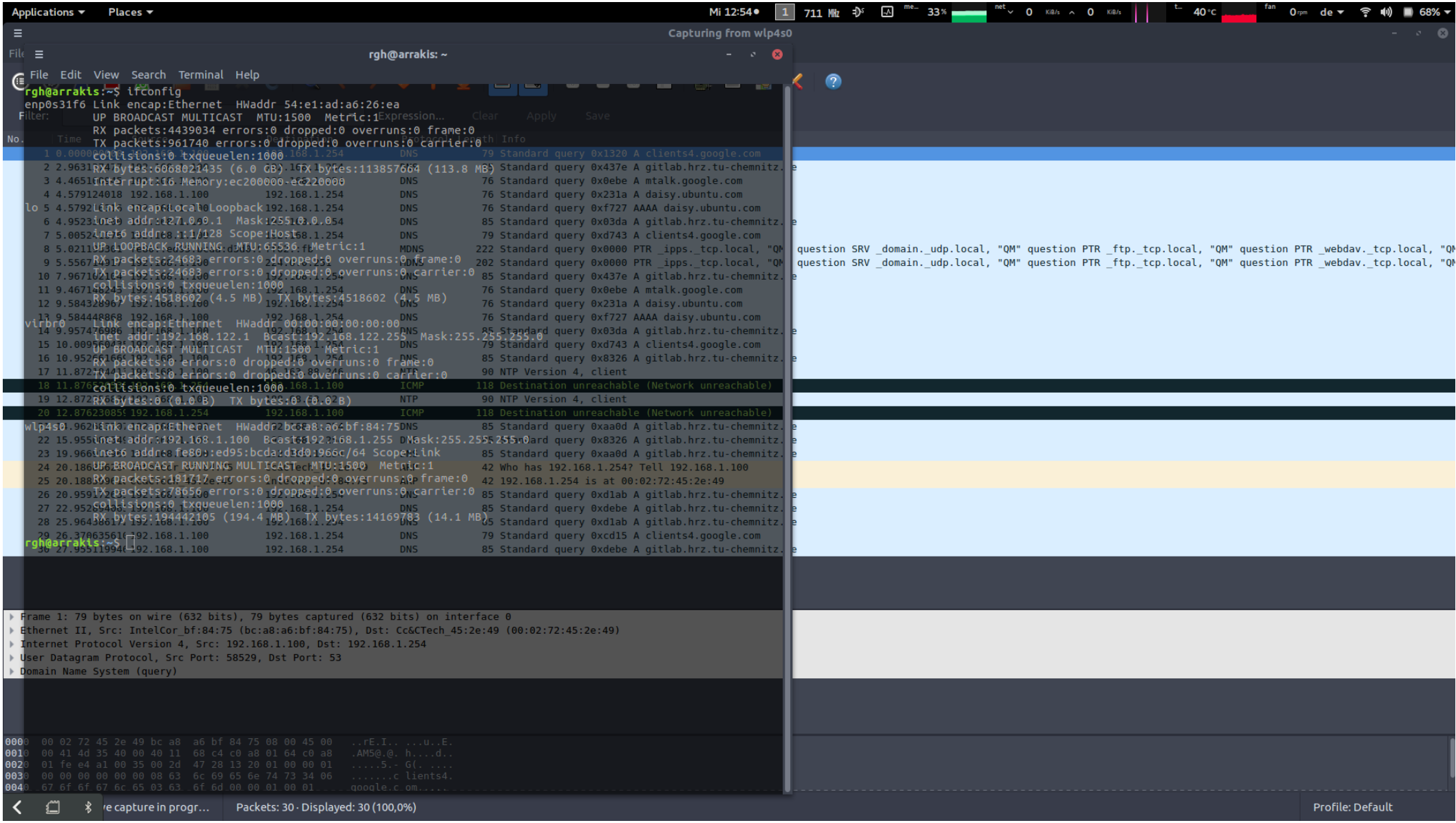
Promiscuous Mode Acer

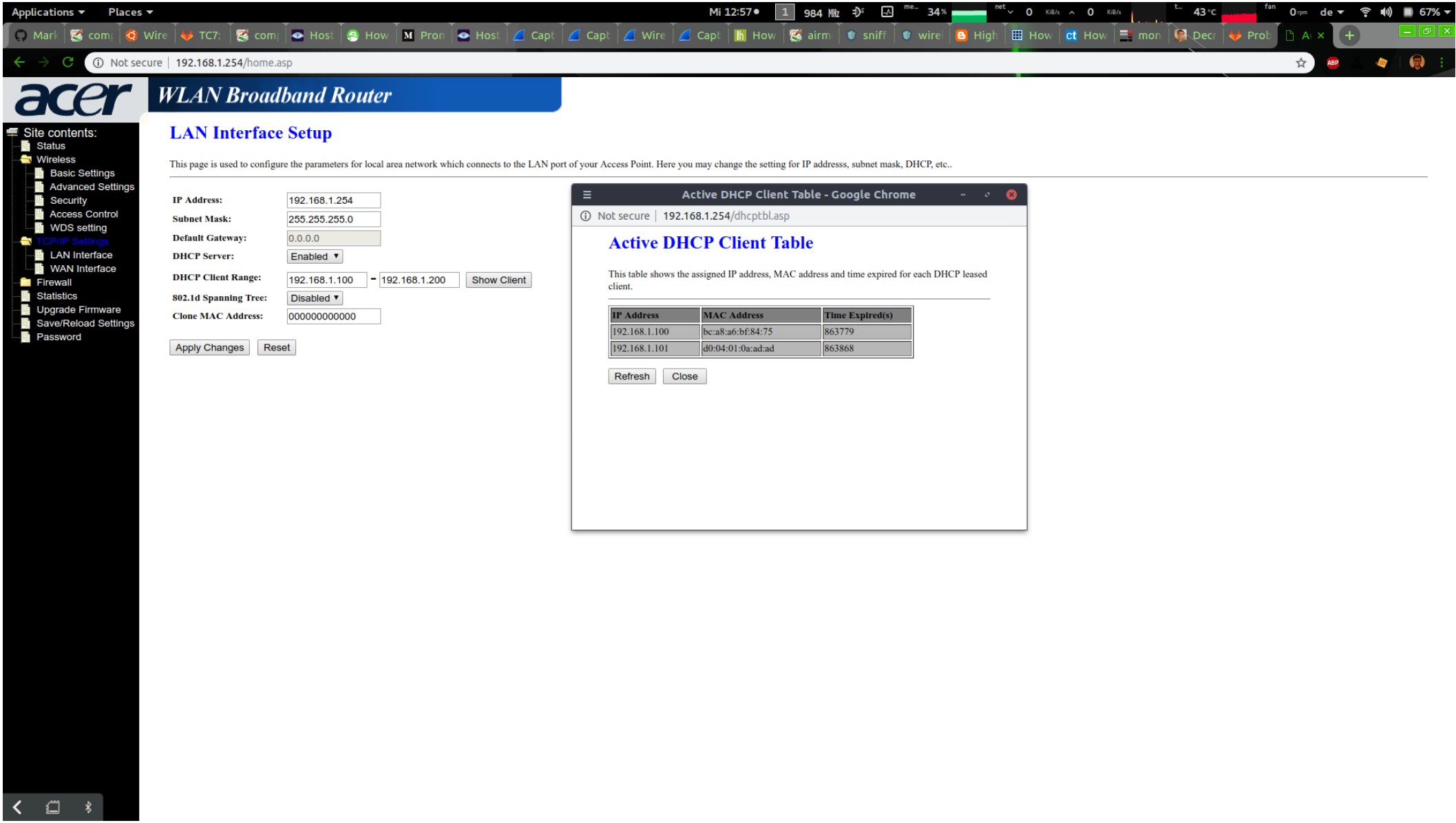
Settings


Applications
Places
Mi 12:54
Wireshark: Capture Options
Capture
Capture
Interface
Link-layer header
Prom. Mode
Snaplen [B]
Buffer [MiB]
Mon. Mode
Capture Filter
wlp4s0
10.149.26.85
fe80::b091:ad41:342f:1985
Ethernet
enabled
262144
2
disabled
any
Linux cooked
Loopback: lo
127.0.0.1
::1
virbr0
192.168.122.1
enp0s31f6
bluetooth0
Bluetooth HCI UART transport layer plus pseudo-header
nflog
Linux netfilter log messages
nfqueue
Raw IPv4
usbmon1
unknown
usbmon2
Cisco remote capture: ci...
Remote capture dependent DLT
Random packet generat...
Generator dependent DLT
SSH remote capture: ssh...
UDP Listener remote ca...
Exported PDUs
n/a
Capture on all interfaces
Use promiscuous mode on all interfaces
Manage Interfaces
Capture Filter:
Compile selected BPFs
Capture Files
File:
Browse...
Use multiple files
Use pcapng format
Next file every
megabyte(s)
minute(s)
Ring buffer with
files
Stop Capture Automatically After...
packet(s)
file(s)
Display Options
Update list of packets in real time
Automatically scroll during live capture
Hide capture info dialog
Name Resolution
Resolve MAC addresses
Resolve network-layer names
Resolve transport-layer name
Use external network name resolver
Start
Close

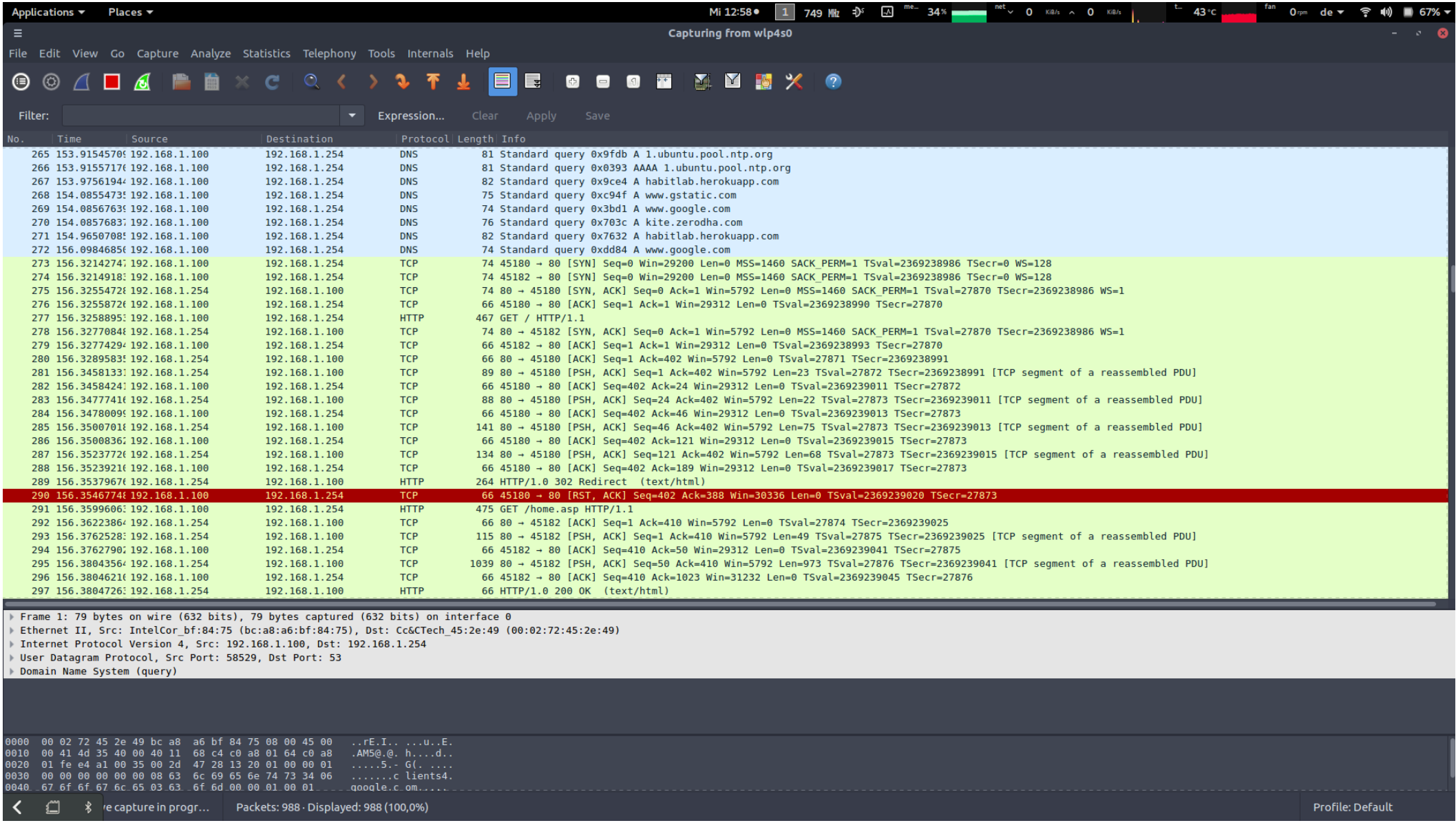
IP Address of computer.



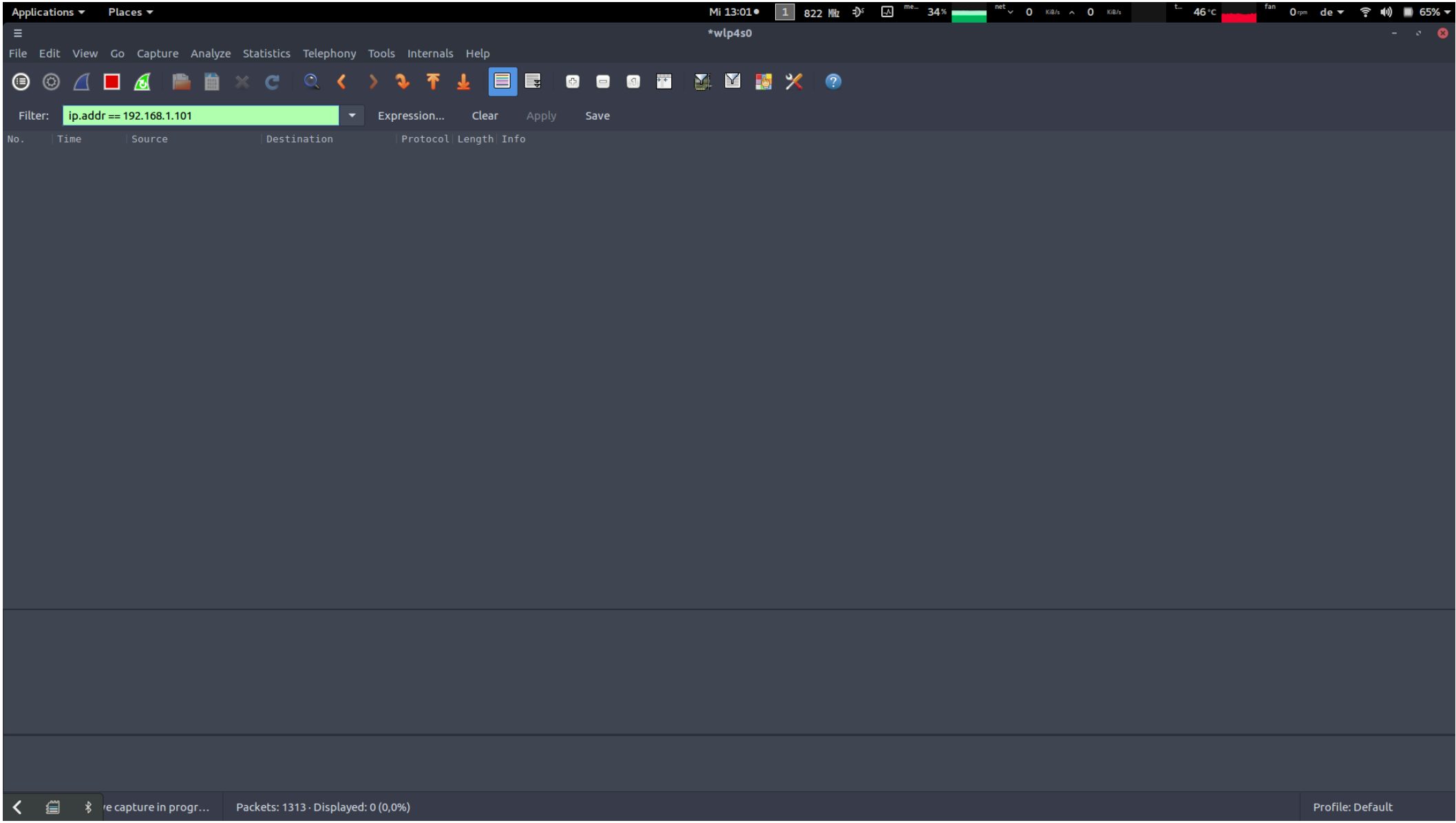
IP Address of phone



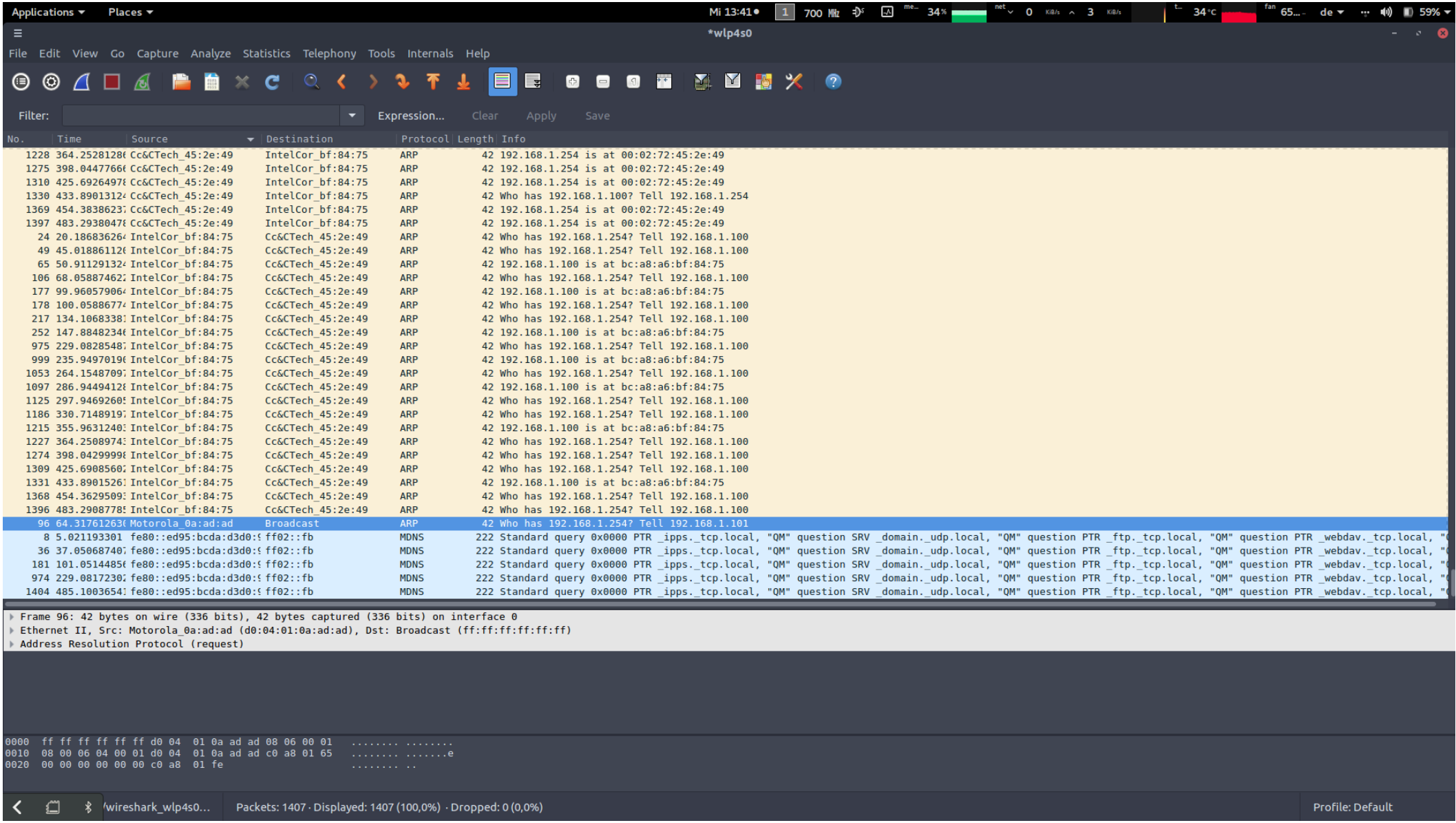
Sniffing begins – mostly packets between router and computer.



The source address is not shown properly for the phone



Not sure why only that ARP packet is visible via the promiscuous mode sniffing.



Conclusion.

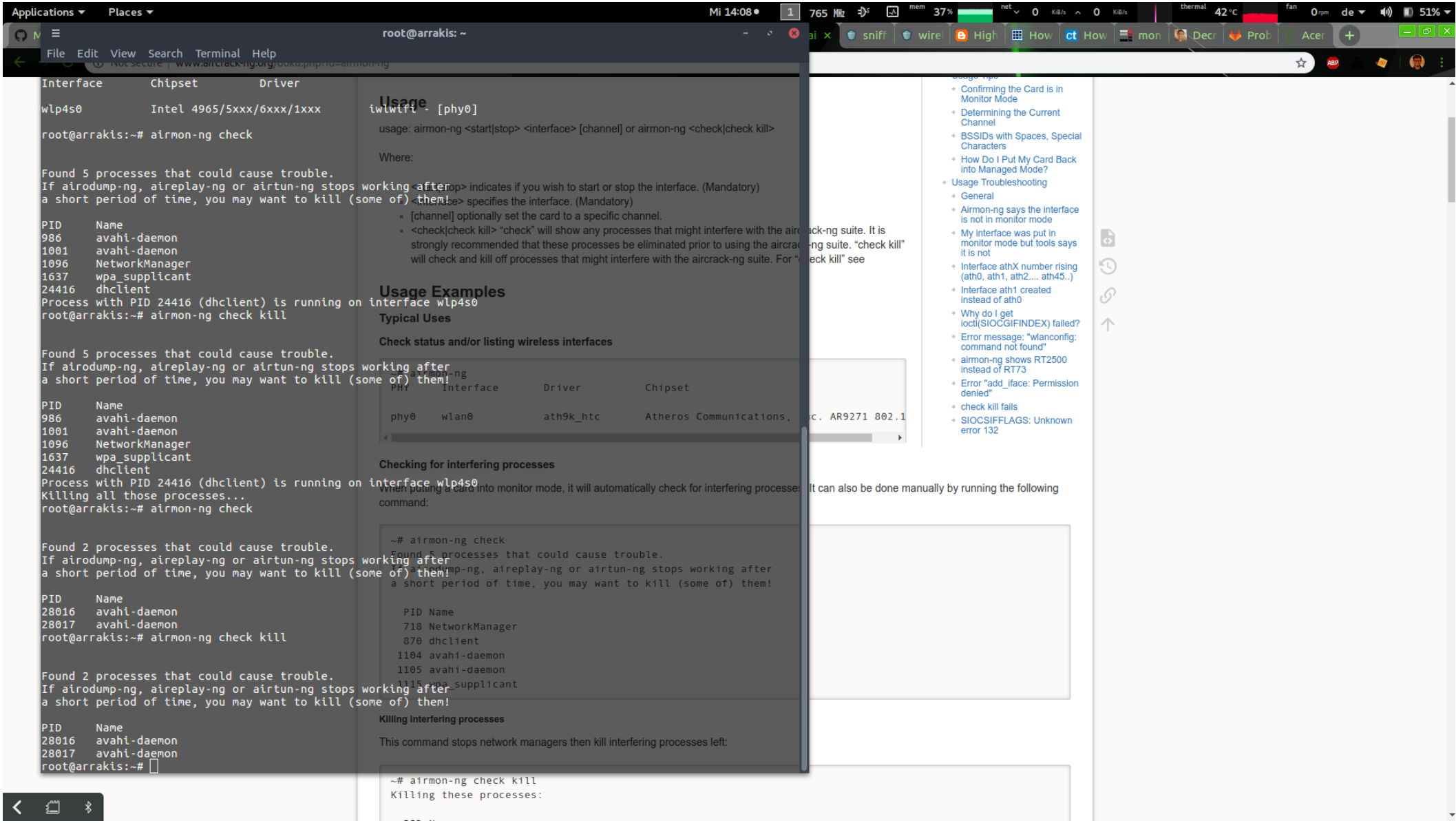
- I expected to see the same volume of message flow from the router to the phone as I saw between router and computer.
- But that didn’t happen. I am not sure if this is because one device was a Linux based computer and the other an Android based phone.
- The same experiment I repeated with the “simple network” below:

ThinkPad 1	AP (Acer router)
ThinkPad 2 (Attacker)	ThinkPad 3

- Note that all the ThinkPads are on the same network which is created by the AP and I use Wireshark on ThinkPad 2 in promiscuous mode to track messages between ThinkPad 1 and ThinkPad 3. In this case also, I could not see the messages.

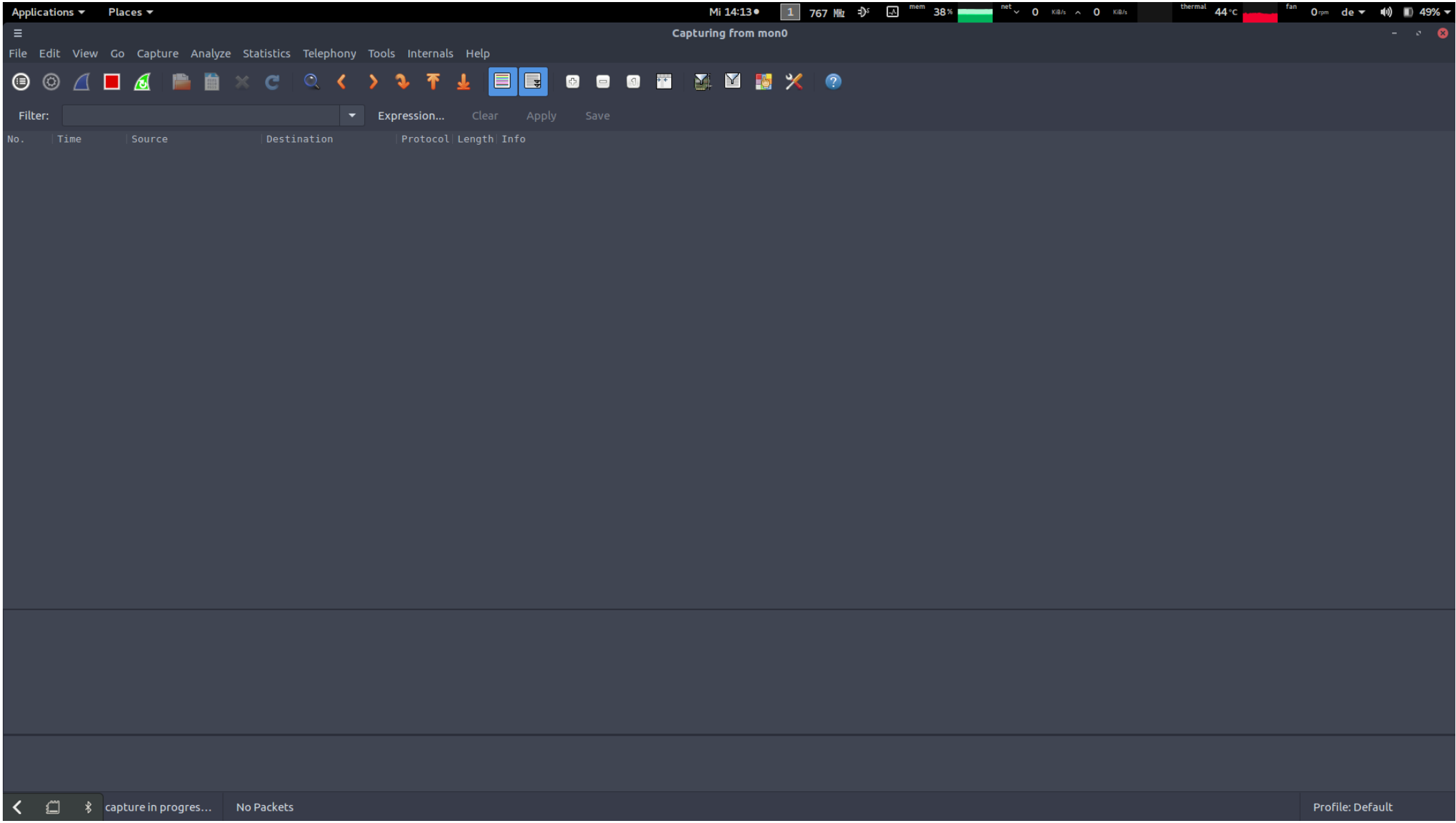
Monitoring mode Acer.

Setting up monitor mode for wireless adapter on the computer.

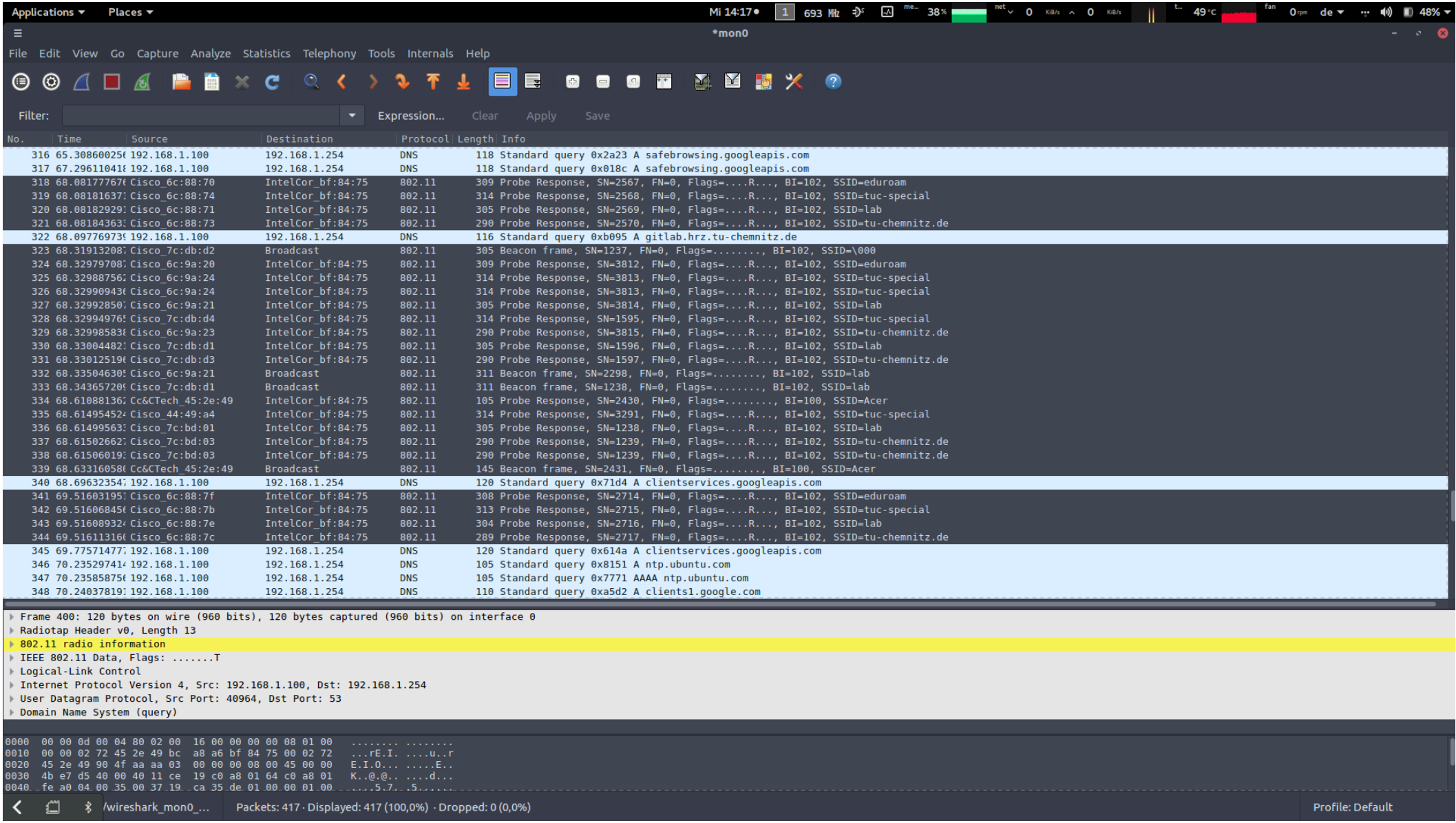


Mon0 is the monitor mode adapter.

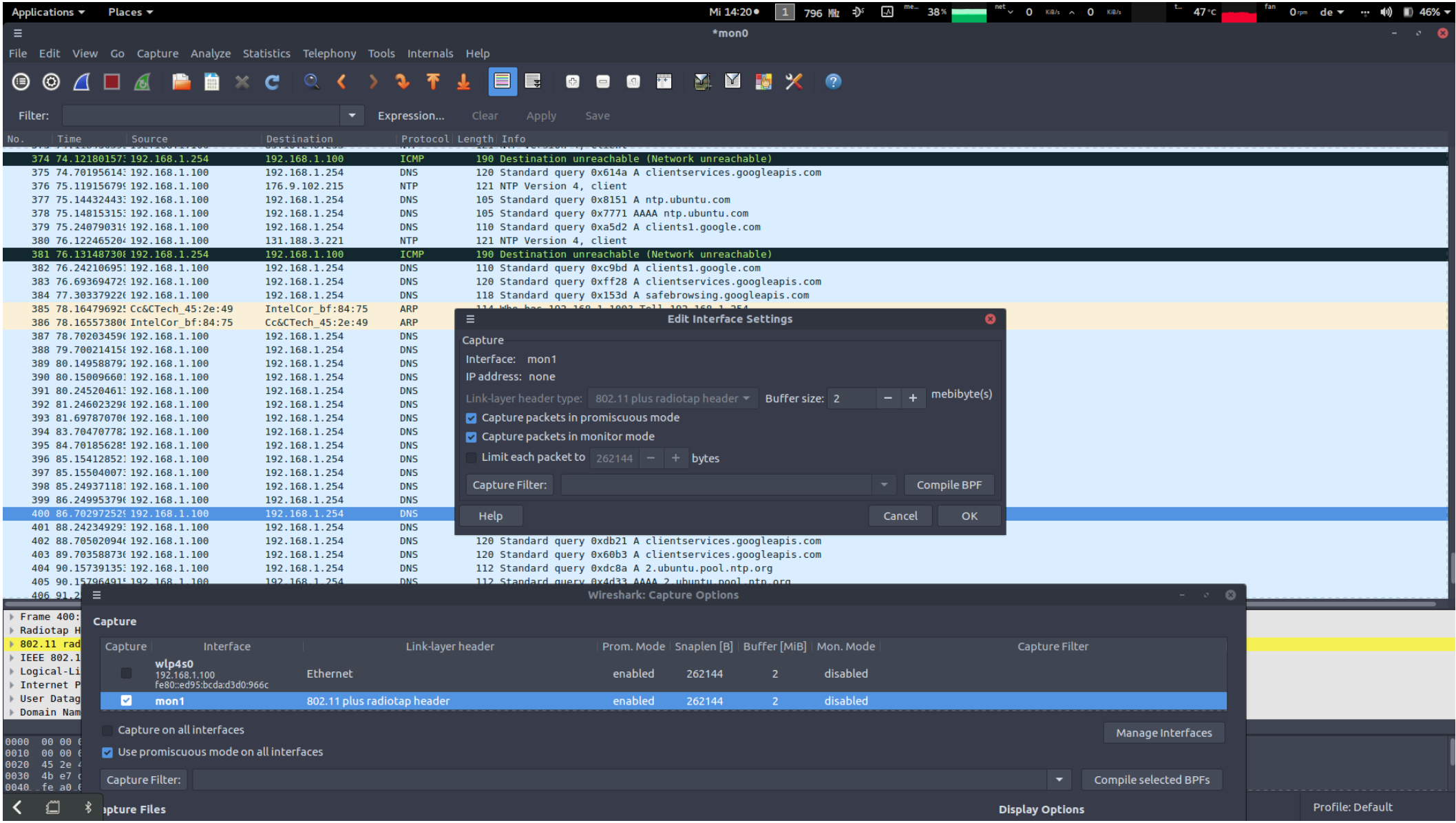
When all the steps from the airmon guide are followed (which includes switching off the network-manager) there is no data captured.



But if I am also connected to the Acer router, then I can see following data but this kind of defeats the purpose of monitoring mode. More importantly, even in this case I cannot actually see the data which is sent from router to phone.



In my view the settings are correct...



Conclusion.

I think the thinkpad computer is capable of wireless monitoring mode – because I read in aircrack-ng user guide that otherwise I would have seen errors in the airmon-ng steps. I didn't see them.

- I went through quite a few "tutorials" and "walkthroughs" for monitor mode packet capture using wireshark and no one actually mentions whether the network-manager should be on or off.
- The airmon-ng guide says that it should be off but that is not aircrack-ng toolset. Not sure how that mixes with wireshark.
- For now I feel that if I am connected to the same network which I am trying to "monitor", then it doesn't really mean I am "monitoring". Defeats the definition.
- Also, Wireshark actually offers an automatic "monitor mode". When using this mode, after trying several times, I could not actually see any data between the phone and the router. I expected to see at least DNS protocol packets and some TCP packets when connection is made.

Overall Conclusion

- This is very similar to what I observed earlier with the GoPros.
- Not able to see data communication between other devices (in this case, the phone and in GoPro's case, the communication between Smart Remote and GoPros) in Promiscuous mode.
- Monitor mode doesn't produce any results when I am not connected to the network using wireshark. Even if I am – and I believe this is the intended way of using it, it doesn't actually show any useful data.
- The experiment was also repeated using the "simple network" as follows:

ThinkPad 1 (connected to Acer AP)	Acer AP
ThinkPad 2 (not connected to Acer AP)	ThinkPad 3 (connected to Acer AP)

- Results were exactly same in this network as well.