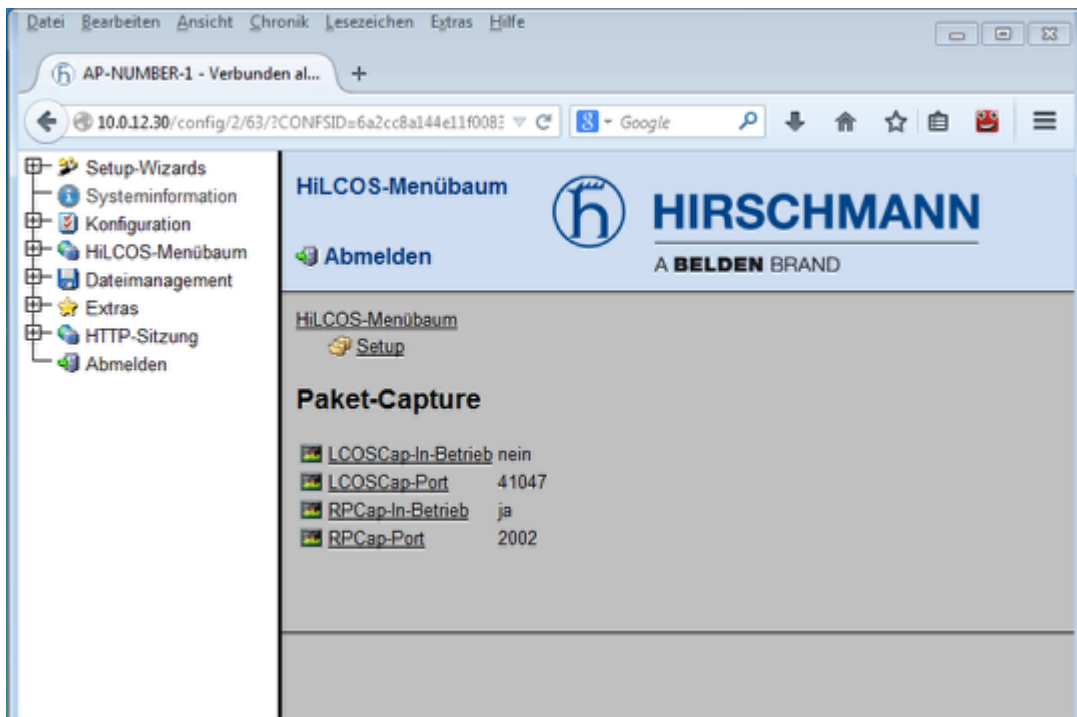


How to remotely capture the traffic of an Open BAT interface with RPCap function and Wireshark

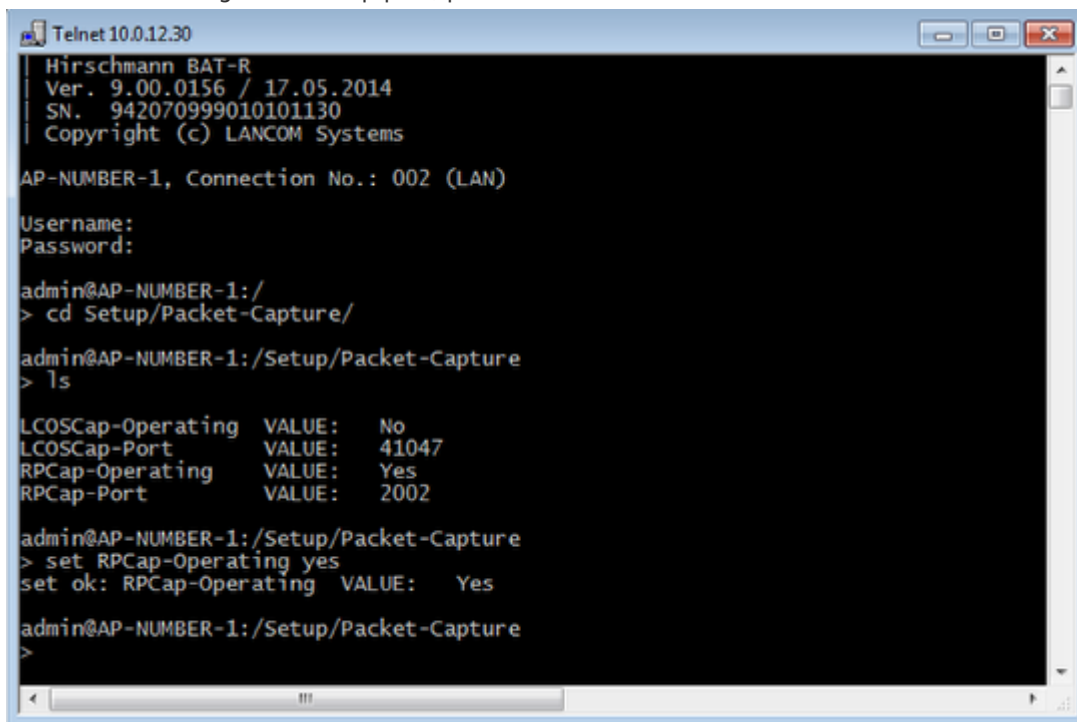
- 2018-02-21 - BAT, WLC (HiLCOS)

This lesson explains via a few steps how to use the RPCap function to capture traffic remotely on specific interface(s) of the BAT devices (rel 8.90)

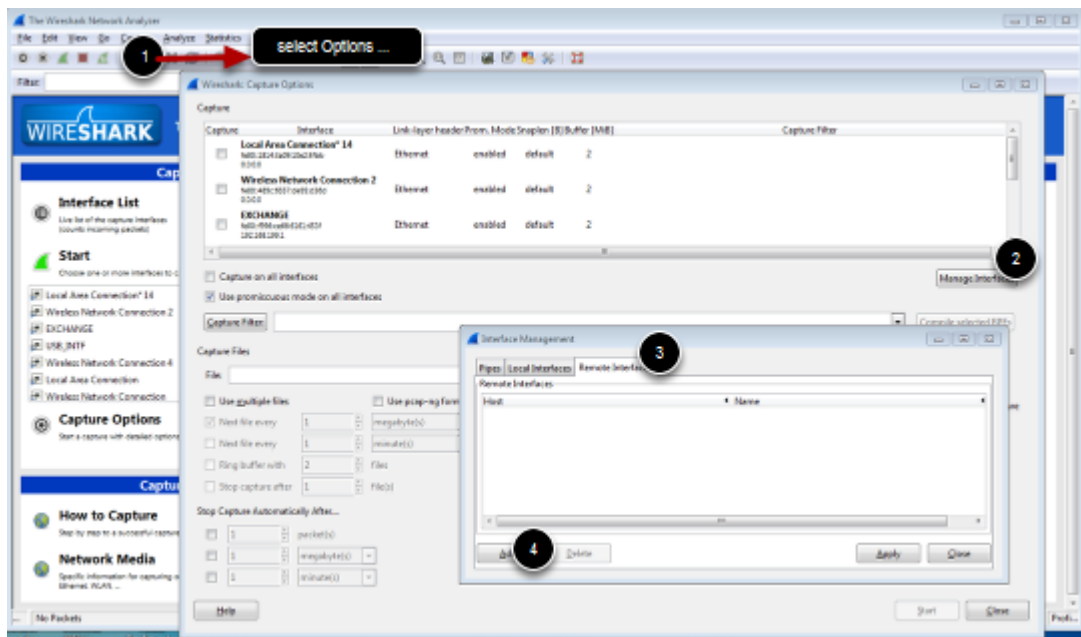
Enable RPCap on the BAT using the web interface or per CLI



You can also change the RPCap port, per default it's 2002



Add remote interfaces in wireshark options



From Wireshark main Windows, open the Capture Options window (Capture/Options...). Click on manage Interface and select the tab Remote Interfaces and click on Add

Configure the BAT as remote device



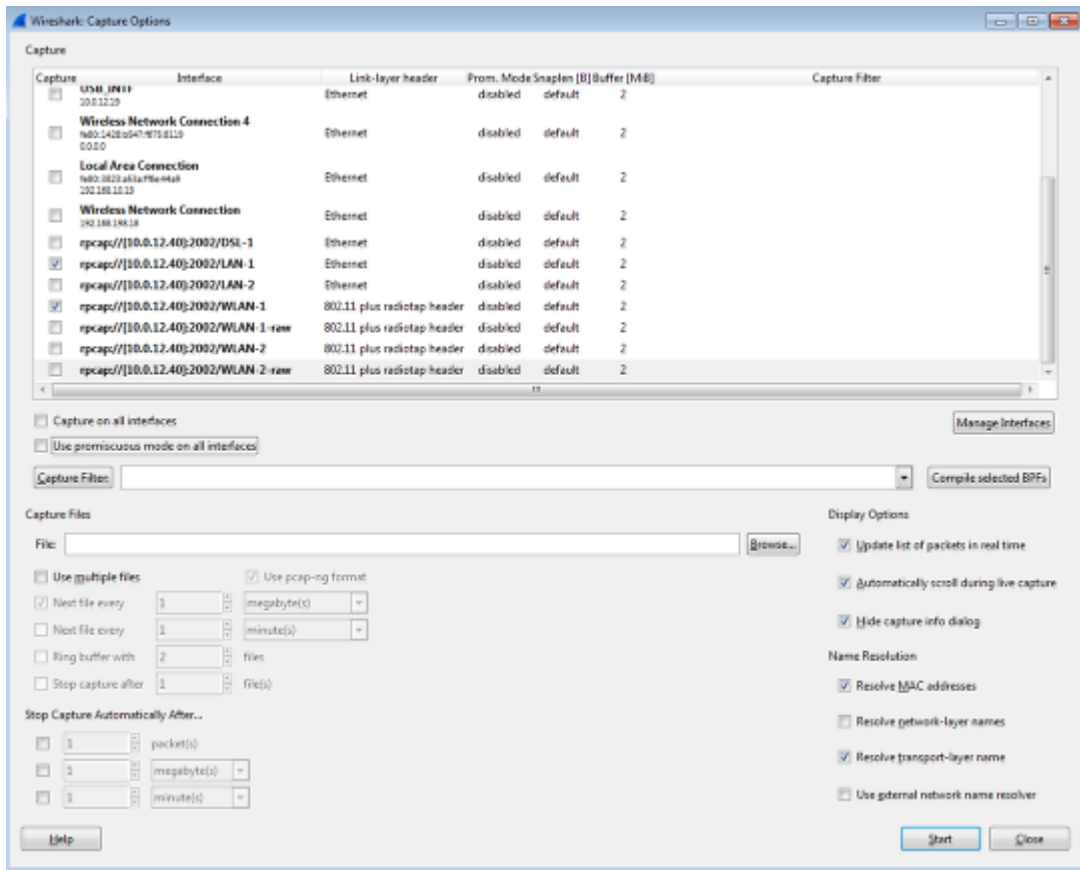
Give the IP address of the BAT, the RCap port relevant username and password to access the device then click ok

RCap gives all the available interfaces on the remote device



click on Apply and Close

From the Capture option Window, the remote interfaces are now available, select the one(s) you want to capture the traffic on.



In this example traffic going through LAN-1 and WLAN-1 will be captured. Then just clic on start

Result view

The screenshot displays the Wireshark network protocol analyzer interface. The top pane shows a list of captured packets with columns for No., Time, Date, Source, Destination, Protocol, and Info. The middle pane shows the details of the selected packet (No. 6), identifying it as an IEEE 802.11 Beacon frame. The bottom pane shows the raw packet data in hexadecimal and ASCII.

No.	Time	Date	Source	Destination	Protocol	Info
1	0.000000000	2016-02-07 07:34:42.124453000	10.0.12.40	10.0.12.19	TCP	mailbox > tdsos190 [ACK] Seq=1 Ack=1
2	0.000477000	2016-02-07 07:34:42.125120000	10.0.12.19	10.0.12.40	TCP	mosaicysysvcl > globe [ACK] Seq=1 Ack=1
3	0.001054000	2016-02-07 07:34:42.125707000	10.0.12.40	10.0.12.19	TCP	globe > mosaicysysvcl [ACK] Seq=1 Ack=1
4	0.002523000	2016-02-07 07:34:42.127174000	10.0.12.19	10.0.12.40	TCP	[TCP Previous segment not captured]
5	0.037883000	2016-02-07 07:34:42.162536000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=618, F=0, Flags=..
6	0.040730000	2016-02-07 07:34:42.165383000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=620, F=0, Flags=..
7	0.047590000	2016-02-07 07:34:42.172243000	Sensoint_87:26:8a	Broadcast	802.11	Beacon frame, S=1027, F=0, Flags=..
8	0.052454000	2016-02-07 07:34:42.177107000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=1067, F=0, Flags=..
9	0.053866000	2016-02-07 07:34:42.178529000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=1068, F=0, Flags=..
10	0.063072000	2016-02-07 07:34:42.187221000	10.0.12.40	10.0.12.19	TCP	[TCP ACKed unseen segment] globe > m
11	0.003442000	2016-02-07 07:34:42.128095000	10.0.12.40	10.0.12.19	RfCAP	update filter reply
12	0.003595000	2016-02-07 07:34:42.128248000	10.0.12.40	10.0.12.19	TCP	[TCP window update] globe > mosaicysy
13	0.038126000	2016-02-07 07:34:42.162779000	10.0.12.40	10.0.12.19	RfCAP	Packet
14	0.055322000	2016-02-07 07:34:42.179975000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=2069, F=0, Flags=..
15	0.055614000	2016-02-07 07:34:42.180267000	10.0.12.40	10.0.12.19	RfCAP	Packet
16	0.057542000	2016-02-07 07:34:42.182193000	10.0.12.19	10.0.12.40	TCP	brcontrol > brutus [ACK] Seq=1 Ack=1
17	0.057810000	2016-02-07 07:34:42.183463000	10.0.12.40	10.0.12.19	RfCAP	Packet
18	0.059263000	2016-02-07 07:34:42.185916000	Hirschma_ff:d2:f3	Broadcast	802.11	Beacon frame, S=1092, F=0, Flags=..
19	0.089998000	2016-02-07 07:34:42.214451000	Hirschma_ff:d2:f3	Broadcast	802.11	Beacon frame, S=1404, F=0, Flags=..
20	0.112718000	2016-02-07 07:34:42.237371000	Hirschma_ff:d5:64	Broadcast	802.11	Beacon frame, S=2869, F=0, Flags=..
21	0.141242000	2016-02-07 07:34:42.265795000	Juniper_N_72:9b:00	Broadcast	802.11	Beacon frame, S=824, F=0, Flags=..

Frame 6: 338 bytes on wire (2704 bits), 338 bytes captured (2704 bits) on interface 1
 IEEE 802.11 Beacon frame, Flags:

RPCap tunnels the traffic between the BAT and the capturing station. Packets from WLAN-1 with radio header and packets from LAN-1 are in the same capture but can be read separately filtering the interface id.