Your First Wide-band Capture

Introduction

A wide-band sniffer is very easy to use. Start the capture, connect your devices and all traffic will immediately be captured. There are a few things that need to be known however, in order to get a perfect capture. This document will guide the reader through some simple steps required to ensure an optimal capture.

Typical Capture Process

The capture process with a wide-band sniffer is as follows:

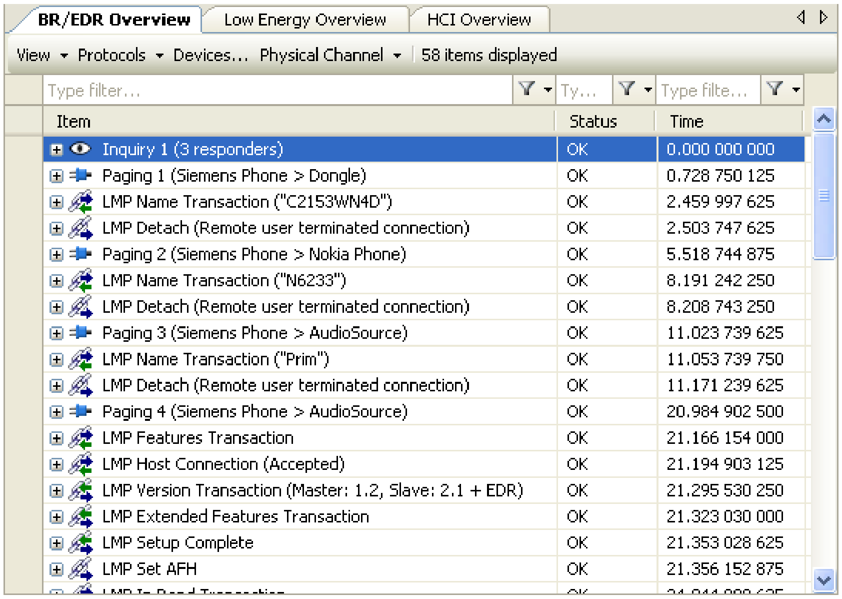
1. Position the analyzer and the devices
2. Start the analyzer
3. Connect and use the desired *Bluetooth* devices
4. Stop the analyzer and save the trace

Too good to be true? Well, a bit. The Ellisys wide-band sniffer is designed to learn device information from the captured information. This is great for usability, but some care needs to be taken when capturing a device for the first time.

Information such as BD_ADDR, friendly name, SDP parameters, L2CAP channels, link key, etc., needs to be known in order to display information successfully.

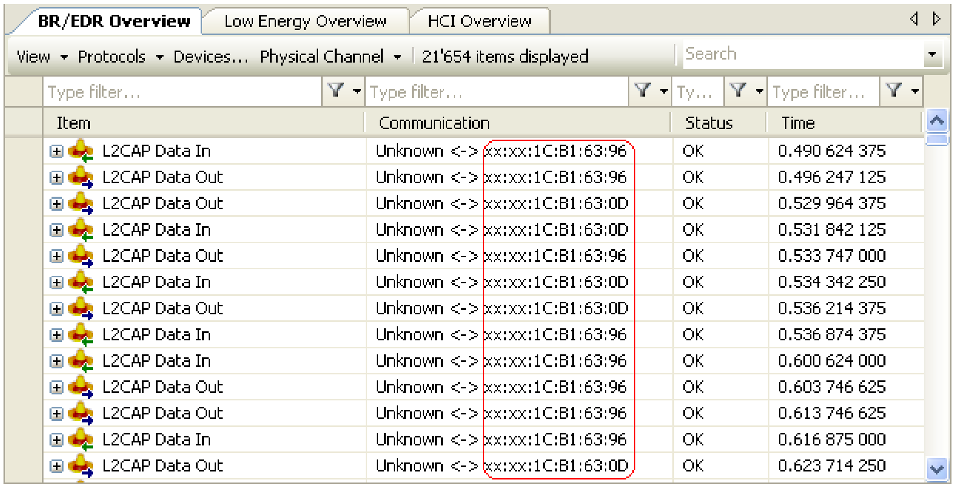
Populating the Ellisys Database

As mentioned above, the Ellisys software will unobtrusively learn various details about devices from the captured traffic. The first piece of information needed by the Ellisys software is the BD_ADDR of the devices. The BD_ADDR of one of two communicating devices is determined when a connection (paging) is captured, but the BD_ADDR of the other device cannot be known from the connection. The easiest way to have all devices send out their BD_ADDR is by doing a discovery (inquiry). When doing a *Bluetooth* inquiry, all nearby devices will send FHSs containing their BD_ADDR. Even better, most *Bluetooth* stacks determine the LMP name as well, so this will also be "learned" by the sniffer.



Item	Status	Time
Inquiry 1 (3 responders)	OK	0.000 000 000
Paging 1 (Siemens Phone > Dongle)	OK	0.728 750 125
LMP Name Transaction ("C2153WN4D")	OK	2.459 997 625
LMP Detach (Remote user terminated connection)	OK	2.503 747 625
Paging 2 (Siemens Phone > Nokia Phone)	OK	5.518 744 875
LMP Name Transaction ("N6233")	OK	8.191 242 250
LMP Detach (Remote user terminated connection)	OK	8.208 743 250
Paging 3 (Siemens Phone > AudioSource)	OK	11.023 739 625
LMP Name Transaction ("Prim")	OK	11.053 739 750
LMP Detach (Remote user terminated connection)	OK	11.171 239 625
Paging 4 (Siemens Phone > AudioSource)	OK	20.984 902 500
LMP Features Transaction	OK	21.166 154 000
LMP Host Connection (Accepted)	OK	21.194 903 125
LMP Version Transaction (Master: 1.2, Slave: 2.1 + EDR)	OK	21.295 530 250
LMP Extended Features Transaction	OK	21.323 030 000
LMP Setup Complete	OK	21.353 028 625
LMP Set AFH	OK	21.356 152 875

When the full BD_ADDR of a device is not known by the analyzer prior capturing the device's traffic, the sniffer can still partially determine the BD_ADDR, most of the time. In this case, the upper bytes will be indicated as missing with "xx" in the BD_ADDR as shown in the analyzer's capture below. The traffic can still be captured successfully, but it will not be possible to decrypt the traffic on-the-fly if the BD_ADDR is not fully known, since this is one of the inputs of the security algorithms.



Item	Communication	Status	Time
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:96	OK	0.490 624 375
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:96	OK	0.496 247 125
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:0D	OK	0.529 964 375
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:0D	OK	0.531 842 125
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:96	OK	0.533 747 000
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:0D	OK	0.534 342 250
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:0D	OK	0.536 214 375
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:96	OK	0.536 874 375
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:96	OK	0.600 624 000
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:96	OK	0.603 746 625
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:96	OK	0.613 746 625
L2CAP Data In	Unknown <-> xx:xx:1C:B1:63:96	OK	0.616 875 000
L2CAP Data Out	Unknown <-> xx:xx:1C:B1:63:0D	OK	0.623 714 250

Once we have full BD_ADDRs of the devices, capturing the pairing procedure will then enable the Ellisys software to learn the missing pieces. During pairing, the devices discover each other's capabilities and exchange information that is useful in order for the sniffer to correctly decode the protocols and profiles.

The pairing is also useful for determining the Link Key. In the case of a PIN-code based pairing, or with an SSP pairing in Debug Mode, the sniffer will automatically deduct the Link Key. In other cases, the Link Key needs to be entered into the Security pane.

After these steps, all further connections involving these two devices will be decoded perfectly by the sniffer. The sniffer will remember all data necessary to display useful information, including the Link Key for decrypting the data.

Different Approaches

The steps above are obviously just suggestions and various other approaches can be used. The most important thing to understand is that the device information mentioned above is required only in order to decrypt the data and decode it into various protocols. It is not required however for the capture itself, since a whole-band sniffer is capable of capturing any *Bluetooth* packet without this information.

Another important concept is that the Ellisys software learns information and then stores it in its local database, as well as in the capture file itself. If some information is missing at capture time, the trace might not be unusable right away. However, the missing information may be captured at a later point, and older traces could be reopened successfully as soon as this information is learned by the software.

Let's take a simple example. We are capturing two completely new devices with the analyzer. These two devices are already paired and we don't want to re-pair. We also don't want to do an inquiry, so we start capturing the connection right away. In this case, the Ellisys sniffer will just know the BD_ADDR of the master device and nothing else, so it is not possible to decrypt the data. We save this capture. We then do a second capture where the device that was slave is now the master. At this point we know the BD_ADDRs of both devices and we can decrypt data when the link key is provided. Now that all information is known, we can reopen the first capture, which will be successfully decrypted and decoded as the required information has been learned by the software. The new information will be saved in this trace so now contains all what is needed. It can thus be exchanged with a remote colleague who never had access to the actual devices.

Feedback

Feedback on our Expert Notes is always appreciated. To provide comments or critiques of any kind on this paper, please feel free to contact us at expert@ellisys.com.

Other interesting readings

- EEN_BT01 - Capturing Bluetooth Traffic, the Right Way
- EEN_BT02 - Bluetooth Analysis Tutorial
- EEN_BT04 - Optimal Placement of Your Analyzer
- EEN_BT05 - Understanding Antenna's Radiation Pattern
- More Ellisys Expert Notes available at: http://www.ellisys.com/technology/expert_notes.php

Rev. A. Updated 2011-05-16