Better Analysis

Your First Wideband Capture

How to Achieve the Perfect Capture

Introduction

Ellisys wideband sniffers are designed to be very easy to use. With zero configuration, captures can be initiated with a single click. The user can start a capture, connect the devices of interest, and immediately begin to understand a wide variety of performance and other behaviors, including conformance to design criteria, reliability aspects, errors, coexistence issues, etc.

With the wideband approach, all traffic will immediately be captured and displayed live, which is good, but this also presents an obvious need to understand how to drill down to isolate your devices of interest. To get the ideal capture, there are a few helpful things that you should know. This expert note will guide you through a few of the steps required to ensure that you can maximize the effectiveness of your Ellisys analyzer.

Good Things to Remember

The Ellisys wideband sniffer is designed to learn and retain important device parameters from the captured information or from information entered by the user, such as a link key. Information such as the BD_ADDR, friendly name, SDP parameters, L2CAP channels, link key, audio codecs, etc. are all necessary in order to display the information successfully.

The link key can be captured (over HCI), entered manually, or even injected programattically and will be saved for use in future connections. If your device uses (and transmits) an Identity Resolving Key (IRK), the same thing applies – the analyzer will capture it and the software will remember it.

HELPFUL HINT: In very busy environments, you may wish to deselect the “eye” icon on the toolbar of the Instant Piconet **(Show/Hide Broadcast Devices)** to hide broadcast traffic, to help isolate established (or establishing) piconets visually. Once you install your device filter, turn this back on so you can see broadcast events created by devices included in your filter.

Capture Process

1 Position

Position the analyzer and the devices in reasonable proximity. (See Expert Note, EEN_BT04 “Optimal Placement of your Analyzer” for details on optimal placement.)

2 Configure

Configure the recording settings as needed (Record | Recording Options menu). This tells the analyzer what over-the-air traffic types and/or wired traffic types you want to capture, and controls other things, like radio sensitivity and a long-term capture mode.

3 Record

Now you’re ready to start the capture by simply selecting the Record button, located on the main toolbar.

REC

4 Connect

Connect your Bluetooth devices under test.

5 Stop

Once you’ve captured enough of the device traffic, simply select the Stop button to halt the capture. You can now drill down into the data using the many Ellisys software views. You can do this as recording is ongoing as well.

STOP

6 Save

Save the trace for further analysis in the future. All devices in the area will be saved, but there is a method that allows you to save just the devices you’re interested in (discussed later).

Filtering

There are many approaches to filtering information in the analyzer software. In the case of a wideband sniffer, learning to use these various filter mechanisms is key to drilling down quickly and efficiently to isolate to devices of interest, protocols of interest, packets of interest, etc.

The "biggest" filter is the device-based filter, one that shows or hides user-specified devices.

On your initial recording, you may see dozens or even hundreds of devices present, and odds are, your interests lie with a few of these, so installing a device filter is often a first step by many users. There are quite a few approaches to installing a device-based filter, including a right-click in the Instant Piconet view on the Piconet desired, or on a communicating pair in the Communication column of an Overview, using the Device Traffic Filter dialog, and other approaches as described in more detail in the User Guide.

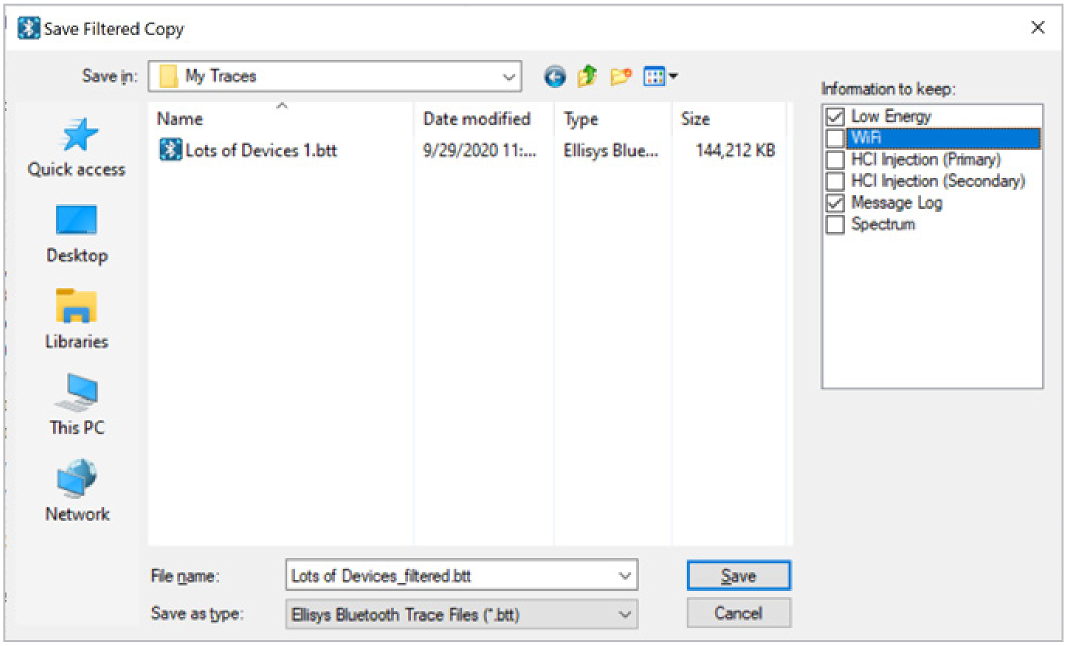

Figure 1 Saving a Filtered Copy.

HELPFUL HINT: Once a device filter is installed, the user can elect to save the existing trace into a new trace that includes only those devices included by the device filter. This feature is executed using **Save Filtered Copy**, located in the **File** menu. **See Figure 1**. In addition, the user can opt to remove certain capture components, such as raw spectrum information, Wi-Fi captures, HCI captures, etc. This can significantly reduce file size and makes sharing of files much easier (see the built-in cloud-based trace sharing, located in the **File** menu). The original trace is NOT replaced – a new trace is created from the original.

Populating the Devices Database Automatically

As mentioned above, the Ellisys wideband 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 (Bluetooth device address). The BD_ADDR of one of two communicating devices is determined when a connection is captured (either paging or advertising, depending upon the use of classic BR/EDR or Low Energy, respectively), but the BD_ADDR of a connecting device cannot be known from the connection. An easy way to have all devices send out their BD_ADDR is by doing a discovery from a Bluetooth device. When a Bluetooth inquiry (BR/EDR) is sent, for example, all nearby devices will send FHS packets containing their BD_ADDR and other useful information. See **Figure 2** for a typical FHS packet's contents.

HELPFUL HINT: The user can change the name of a device using the **Edit** button in the Device Traffic Filters dialog, accessible from the **Filter**: drop-down > **Configure**, located atop the GUI.

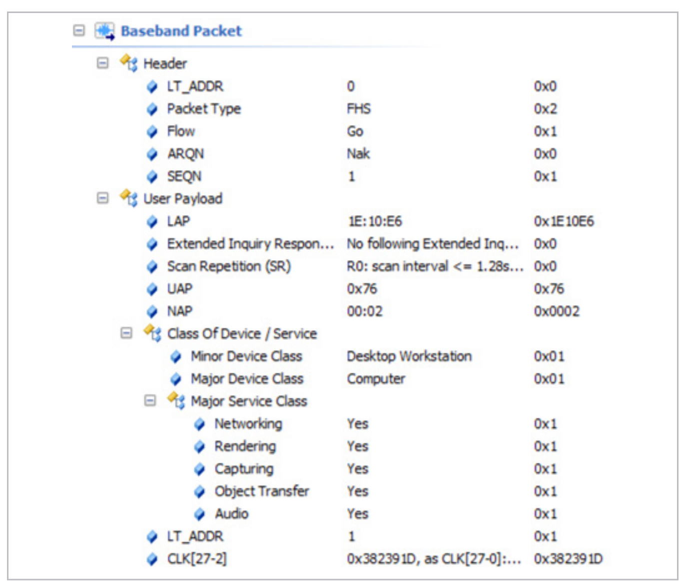

Figure 2 Typical FHS Packet Content.

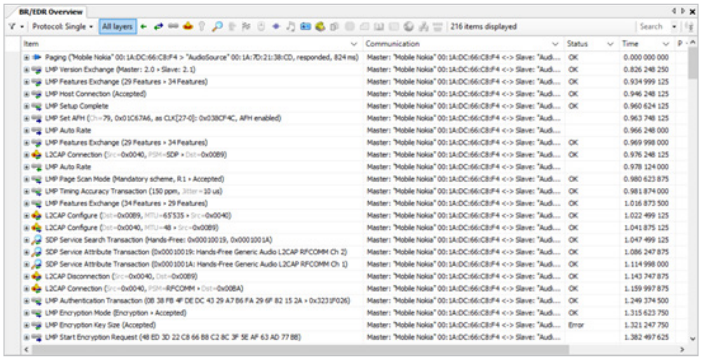

Figure 3 Capture and Display of LMP Name.

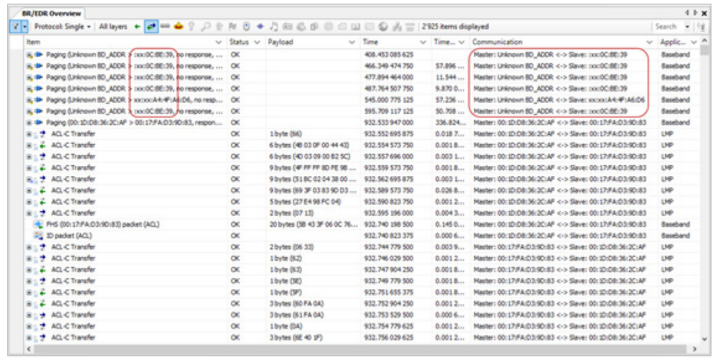

Figure 4 When the BD_ADDR is Not Fully Known.

Even better, most Bluetooth stacks determine the LMP name as well, so this will also be "learned" by the sniffer and used in several places throughout the software. See the Communication column in **Figure 3**.

When the full BD_ADDR of a device is not known by the analyzer prior to 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 **Figure 4**. 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 to the security algorithms.

Populating the Device Database Manually

An alternative method to informing the analyzer software of a device address is to populate the Device Database manually. This can be done in the **Device Traffic Filters** dialog. To get there, select **View** from the main menu, then **Device Traffic Filters** (or select **Configure** from the drop-down menu on the main toolbar labeled **Device Filter**).

The New Device button (**Figure 5**) enables creation of a new device from scratch, including its BD_ADDR, its friendly name and associated color. Before creating a new device, it is useful to check if the device is not already in the database. The **Search** field is quite useful for this purpose. This dialog also enables the user to update the information of an existing device, which is useful especially to update a partial BD_ADDR (as discussed in the section above). It is also possible to delete an existing device if no longer needed.

HELPFUL HINT: Capturing the pairing process is **key**.

Learning the Next Pieces of Information

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 and link establishment exchanges, the devices discover each other's capabilities and exchange the information that is useful in order for the sniffer to correctly decode the protocols, profiles, and services.

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 deduce the Link Key. In other cases, the Link Key needs to be entered into the Security pane (or if HCI is being captured, the software will automatically extract a link key should it be exchanged over that interface).

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.

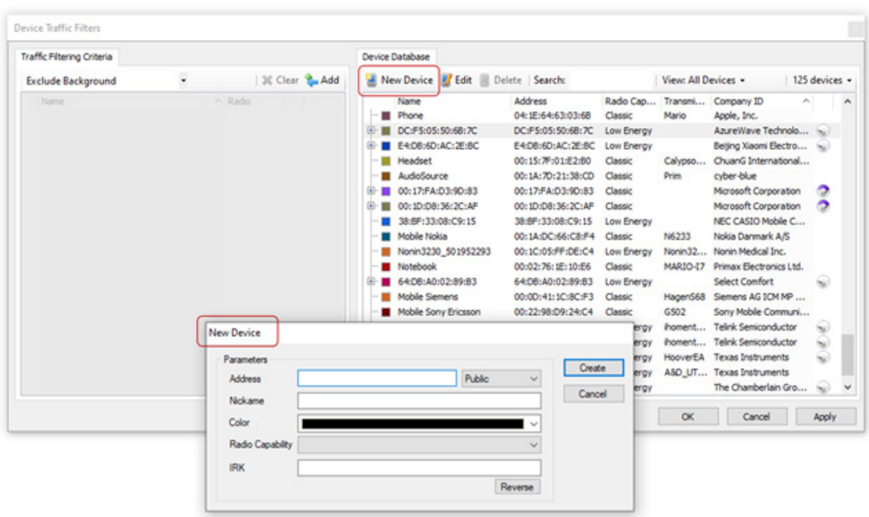

Figure 5 Populating the Device Database Manually.

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 wideband sniffer is capable of capturing any Bluetooth packet without this information, even encrypted traffic.

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 usable right away. However, the missing information may be updated at a later point, and older traces can 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.

> HELPFUL HINT: Once you have the BD_ADDR of one device, by performing a second capture, the device database will learn the BD_ADDR of the other device, adding this to your device database.

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 the 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 that now contains all of what is needed. It can thus be exchanged with a remote colleague who never had access to the actual devices.

Conclusion

In this Ellisys Expert Note we learned that a wideband sniffer captures all traffic sniffed as part of a typical capture. And, to achieve a more perfect capture, we explored and learned new methods to populate the devices database both automatically and manually, to save files more efficiently by using device-based filters, and how the analyzer captures and stores critical elements linke link keys, IRKs, codecs, etc.

Visit ellisys.com or email support@ellisys.com for more information.

Other Interesting Reading

- EEN_BT06 - Bluetooth Security - Truths and Fictions
- EEN_BT07 - Secure Simple Pairing Explained

More Ellisys Expert Notes available at:
www.ellisys.com/technology/expert_notes.php

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

Sales Contact:

USA: +1.866.724.9185
Asia: +852 2272 2626
Europe: +41 22 777 77 89

sales@ellisys.com

www.ellisys.com

Connect with us.