User’s Manual

Net.Shark
Net.Hunter

GbE Frame Capture and Analysis Guide

ALBEDO
Telecom

Copyright

Issue 4, 10/13

For any query or requirement regarding the *Net.Shark GbE Filtering & Aggregation Tap or the Net.Hunter Network Capture Device*, contact with ALBEDO Telecom using the following contact details:

ALBEDO Telecom S.L.
C/ Joan d’Àustria 112
08018 Barcelona - Spain

E-mail: *support.telecom@albedo.biz*
Telephone: *+34 93 221 28 73*

Table Of Contents

Chapter 1
Introduction

The ALBEDO Telecom Net.Shark is a handheld intelligent tap which is capable to filter, select and capture Ethernet traffic in optical or electrical interfaces up to 1 Gb/s. Thanks to these capabilities, Net.Shark is useful both for protocol analysis and content analysis. ALBEDO Net.Hunter offers the same than Net.Shark but it adds bidirectional wirespeed capture of network traffic that totals up to 2 Gb/s capture throughput.

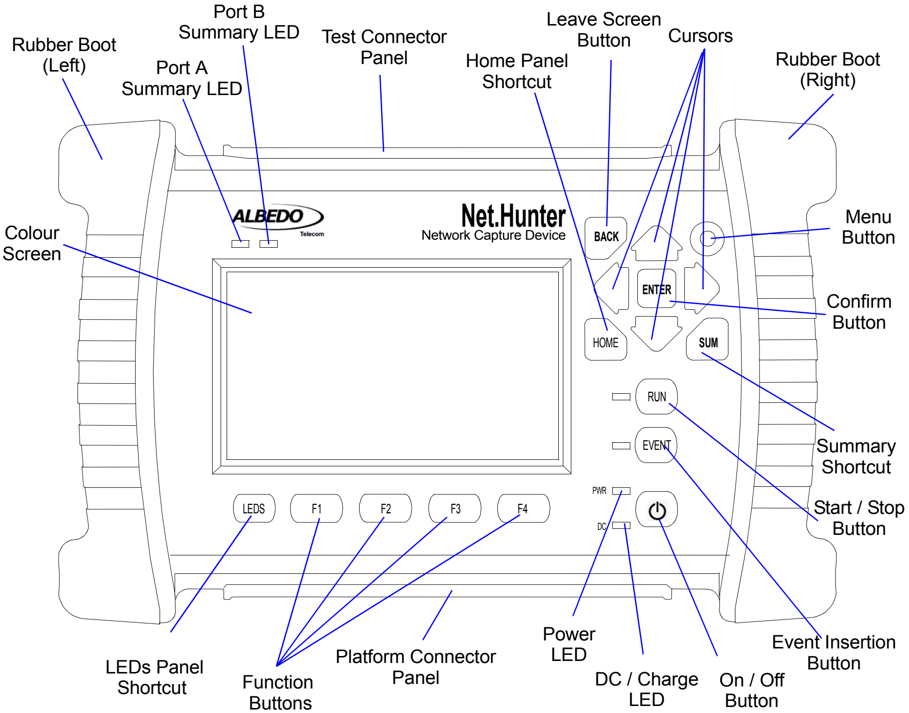


Figure 1.1: Net.Hunter front view. The equipment presents results by means a colour screen and the LEDs. The configuration is performed through the keyboard.

Net.Shark / Net.Hunter has an external DC input but it also has internal batteries. This makes this equipment suitable both for laboratory applications and field applications which require versatile and reliable operation.

Within your Net.Shark kit you will find the following items:

- One Net.Shark / Net.Hunter unit.
- Two electric SFPs with RJ-45 connector to be connected to the line ports.
- One AC/DC adapter with a power cord specific for your country.
- One Carrying bag.
- Two Cat. 5e cables with RJ-45 connectors certified for operation at 1 Gb/s rates.
- Two SFPs for connection to optical interfaces (if ordered).
- Two MMF or SMF cables to be used with the SFPs (if ordered).
- One CD-ROM with user documentation.
- One printed copy of this user manual (if ordered).

Check with your distributor the availability of other optional items for your Net.Shark unit.

1.1. Important Notice

Operation, manipulation and disposal warnings for your Net.Shark unit are listed below.

1.1.1. Warranty

The ALBEDO Telecom Net.Shark /Net.Hunter is supplied with a warranty that includes replacement of damaged or faulty components in the terms and period described in the ordering information. This Warranty does not apply to:

1. Product subjected to abnormal use or conditions, accident, mishandling, neglect, unauthorized alteration, misuse, improper installation or repair or improper storage.
2. Product whose mechanical serial number or electronic serial number has been removed, altered or defaced.
3. Damage from exposure to moisture, humidity, excessive temperatures or extreme environmental conditions.
4. Damage resulting from connection to, or use of any accessory or other product not approved or authorized by the Company;
5. Product damaged from external causes such as fire, flooding, dirt, sand, weather conditions, battery leakage, blown fuse, theft or improper usage of any electrical source.

1.1.2. Battery Safety

The ALBEDO Telecom Net.Shark / Net.Hunter contains a built-in battery, improper use of which may result in explosion. Do not heat, open, puncture, mutilate, or dispose of the product in fire. Do not leave the device in direct sunlight for an extended period of time, which could cause melting or battery damage.

1.1.3. WEEE Notice

This product must not be disposed of or dumped with other waste. You are liable to dispose of all your electronic or electrical waste equipment by relocating over to the specified collection point for recycling of such hazardous waste. For more information about electronic and electrical waste equipment disposal, recovery, and collection points, please contact your local city centre, waste disposal service, or manufacturer of the equipment.

1.2. The Equipment

Interaction with Net.Shark / Net.Hunter is based on a high resolution colour screen, different kinds of status LEDs, and a keyboard. These are the keyboard elements:

- *Cursors*: Enable navigation through the graphical user interface. Including menus, keyboards and configuration lists. To leave a menu or configuration list, the left arrow can be used. In menus, the right arrow enters in the lower level menu or list.
- *ESC*: Leaves the current panel (menus, lists, and special panels).
- *ENTER*: In menus, enters in the lower level menu or list. In a configuration list, it selects the current item and leaves. In keyboards and some special panels, it selects the current item without leaving.
- *HOME*: Shortcut to the *Home* panel. From any menu, list, or special panel, it returns directly to *Home*.
- *SUMMARY*: Displays the *Summary* panel. If the Summary panel is already shown, it returns to the previous panel.
- *LEDS*: Displays the *LEDs* panel. If the LEDs panel is already shown, it returns to the previous panel.
- *MENU* (Net.Hunter only)*:* This key is reserved for future applications.
- *EVENT:* This key doesn’t have any defined functionality for this model.
- *Start / Stop:* This button starts / stops traffic capture / drop actions. Most of the configuration is blocked during the capture / drop action.
- *Function keys (F1, F2, F3, F4)*: These keys do not have a fixed purpose. Their associated action depends on the panel being displayed.
- *On / Off key*: If the equipment is in off status, push to switch it on. If the equipment is on, use this key to switch it off (long push).

There are four LEDs (PWR, DC, line / mirror Port A summary, line / mirror Port B summary). Their description is given below:

- *PWR*: Displays the current equipment on / off status. The green colour is displayed under normal operation conditions. Orange and red are shown to indicate a low battery load.

- *DC*: This led is lit when the DC input is connected. Orange indicates a charging batteries status and green means that the internal batteries are ready.
- *Port A / Port B Summary*: These LEDs provide a permanent indication of the current status of the line ports (SFP ports) and mirror ports (RJ-45 ports). The LEDs summarize the Port A and Port B information given by the event LEDs. If any event LEDs for a network port is in ‘red’ status, the port summary led will be set to ‘red’. If any event LED is ‘orange’ but there is no ‘red’ event, the summary led will be set to ‘orange. The ‘green’ colour is used when no events are found in the input signal. Finally, the LED is switched of when the port is disabled.

1.2.1. Network Connectors

Net.Shark is connected to the DUT / SUT through the network connector panel. Ports and elements included in this panel are described in the following list:

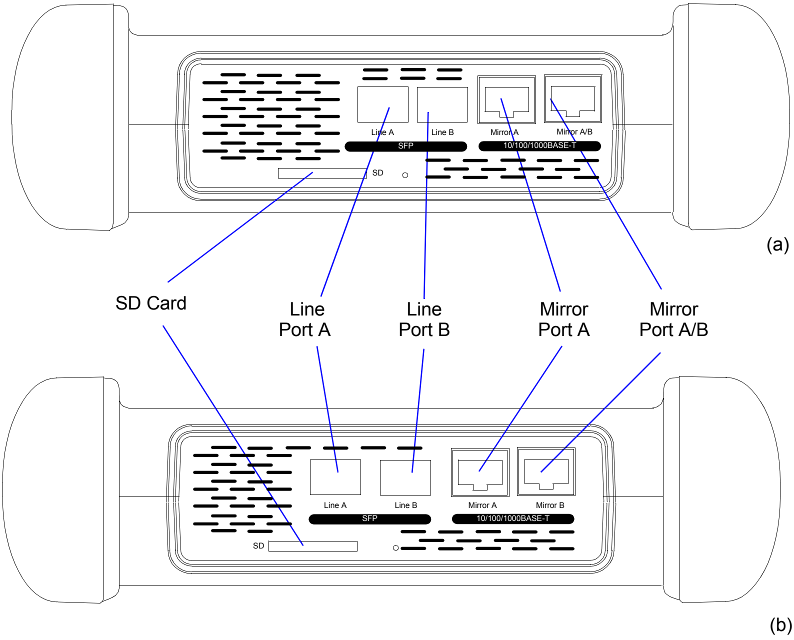


Figure 1.2: Test connector panel. Connection to the DUT / SUT is done in this panel: (a) Net.Hunter test connection panel, (b) Net.Shark test connection panel

- *Line port A*. This port is used to connect the equipment to the network with the help of an optical or electrical SFP module.

- *Line port B*. This port is used to connect the equipment to the network with the help of an optical or electrical SFP module. This port is identical to the RJ-45 Port B.
- *Mirror port A*. This is the secondary Net.Shark mirror port. Traffic from the line port A is selectively copied and dropped to this interface. This port is an Ethernet 10/100/1000BASE-T interface enabled for transmission only (it ignores all the traffic it receives).
- *Mirror port A/B*. This is the primary Net.Shark mirror port. Traffic from the line port B and, depending on the configuration, from line port B is selectively copied and dropped to this interface. This port is an Ethernet 10/100/1000BASE-T interface enabled for transmission only (it ignores all the traffic it receives).
- *SD Card*: Slot for SD Cards. These cards can be use as external storage devices for captures (Net.Shark only) or for configurations.

1.2.2. Platform Connectors

There is a connector panel specifically devoted to the platform ports. This panel includes capabilities like remote control and external device connection. A more detailed description is given below:

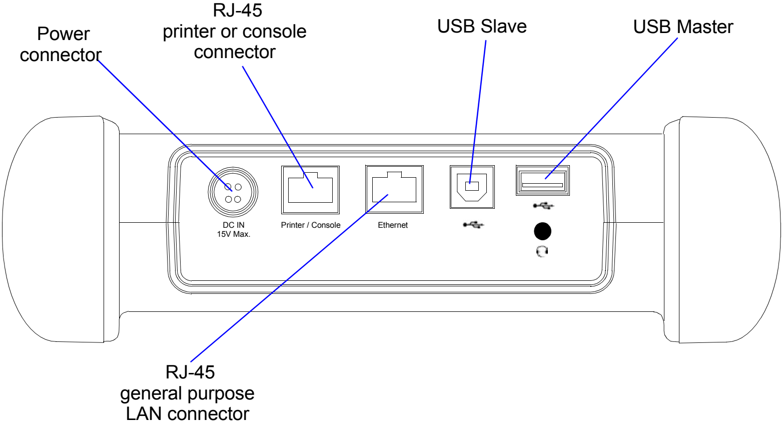


Figure 1.3: Net.Shark / Net.Hunter platform connector panel. This panel includes connectivity to USB devices, remote control and other features.

- *Power connector*: The input must be 12 V DC, 4 A. A suitable external AC/DC adapter for your country is provided with the equipment.
- *RJ-45 printer or console*. Console connector. This interface is prepared for connecting a serial printer.

- *USB Slave*. Use a USB cable with Slave type connector (Type B, *Device*) for this port. Currently this port enables connection of a PC to the equipment and access to the equipment internal file system.
- *USB Master*: Use a USB cable with a Master type connector (Type A, *Host*) for this port. Currently this port is used for software upgrades and connection of external storage devices.
- *RJ-45 general purpose LAN connector*: This is the platform Fast Ethernet connector (10/100BASE-T). It is used for remote management of the unit.

1.3. The Graphical User Interface

The Net.Shark / Net.Hunter graphical user interface is based in a 480 x 272 colour screen and a set of keys attached to the front panel. Some of these keys have a permanent purpose but the specific function for some other keys depend on the context.

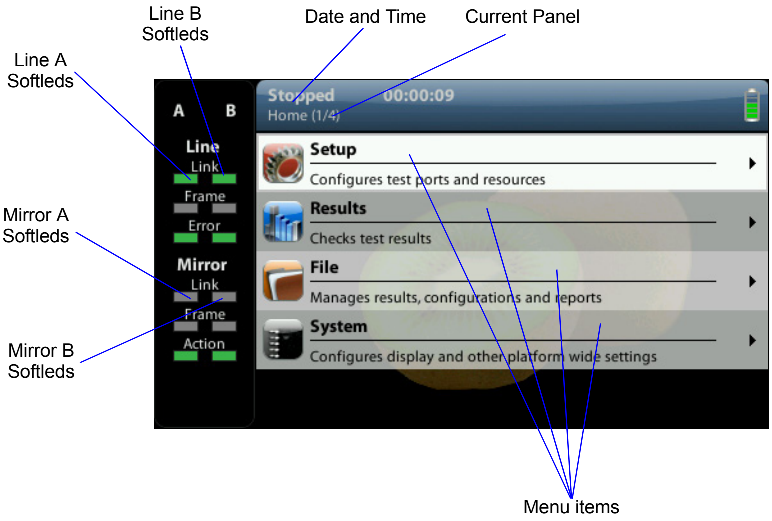


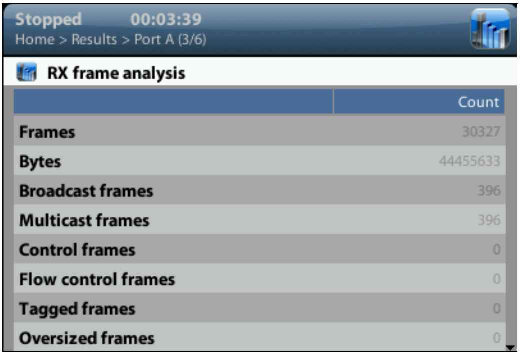
Figure 1.4: Net.Shark / Net.Hunter, the Home panel.

The keyboard and the screen allow the user setting configuration values, starting captures / drops and displaying results. The user is always aware of the current status of the received signal through the Softleds shown on the left. The Softleds are always displayed and they work even when there is no capture / drop enabled. On the top side of the screen there is a header zone which contains information about the current equipment status (date, time, capture / drop action running) and an identifier for the currently displayed panel.

Most of the graphical user interface panels are menus containing a variable number of items. All the menus are available from the *Home* panel. Users can press the HOME button at any time to go to the *Home* panel. The *Home* panel contains the following menu items:

- *Setup*: Provides access to capture / drop configuration. For the tap and capture application, the setup submenus contain configuration of the line / mirror ports and storage devices.
- *Results*: This item enables the user to browse traffic results and retrieve information about current connection status. Some statistics are not available if a capture / drop action has not been previously started.
- *File*: File management menus. Includes configuration, capture and report file management. Files can be deleted, copied, exported or imported.
- *System*: Provides platform management tools. For example language selection, screensaver configuration and others.

There are two special panels as well. These are the *Summary* panel and the *LEDs* panel. The *Summary* provides some details about the current configuration and results. The *LEDs* panel gives extended the information about the received signal status already given by the Softleds. Both the *Summary* panel and the *LEDs* panel can be displayed at any moment by pressing the *SUM* and *LEDS* buttons*.*



Stopped 00:03:39

Home > Results > Port A (3/6)

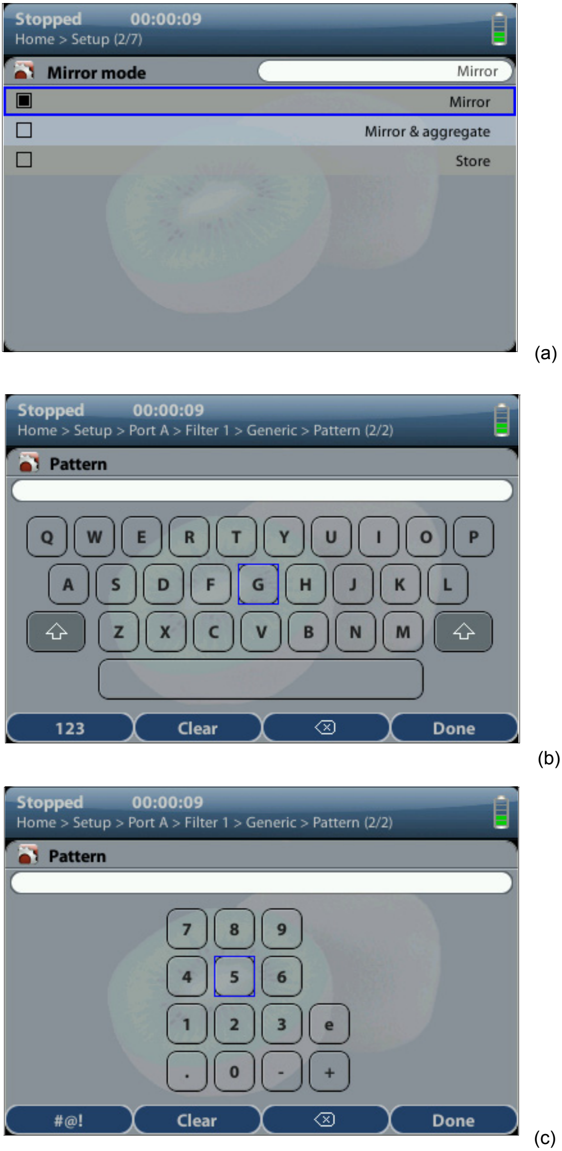
RX frame analysis

	Count
Frames	30327
Bytes	44455633
Broadcast frames	396
Multicast frames	396
Control frames	0
Flow control frames	0
Tagged frames	0
Oversized frames	0

Figure 1.5: Net.Shark /Net.Hunter frame statistics represented as a counter list

Menus and submenus are organized in a tree. The root of the tree is the *Home* panel and the leaves are configuration or result panels. Results are usually presented in a list or a table. If all results cannot be simultaneously displayed, then the user is allowed to use the cursors up and down to browse the list.

Configuration panels are usually selection lists. Sometimes you can select only one simultaneous item in the list and sometimes selection of several items at the same time is possible. Keyboards are available if selection through lists is not possible. There is one keyboard for numeric settings and one for alphanumeric settings.



Stopped 00:00:09
Home > Setup (2/7)

Mirror mode: Mirror

- ■ Mirror
- □ Mirror & aggregate
- □ Store

(a)

Stopped 00:00:09
Home > Setup > Port A > Filter 1 > Generic > Pattern (2/2)

Pattern

(b)

Stopped 00:00:09
Home > Setup > Port A > Filter 1 > Generic > Pattern (2/2)

Pattern

(c)

Figure 1.6: Different kinds of configuration panels: (a) Selection list, (b) Alphanumeric keyboard, (c) Numeric keyboard.

1.4. Running Drops / Captures

In Net.Shark / Net.Hunter, captures and drop actions are controlled by the RUN button. Also, most of the statistics are not available until you start a capture / drop action. This section provides a high level description of the procedure to follow to configure your unit, start a capture / drop action and review the results.

1. Configure the equipment to send / receive signals in the right operation mode and through the right ports (See section 2.1, See section 2.2). Connect it to the network.
2. Configure at least one filtering block in port Line A or Line B (or both) to start matching traffic from the network (See section 4.2).
   Matching traffic is copied to the corresponding mirror port (port Mirror A or Mirror B) as soon as the the corresponding filtering block is enabled. Computing statistics or capturing traffic requires additional steps.
3. Program the capture start time and duration with the help of the *Program* menu (within *Setup*) or start immediately by pressing RUN.
   Note: Most of the configuration is blocked when there is an ongoing action.
4. Wait for the capture to finish or press RUN to finish immediately.
5. Check the results in the *Results* menu (See section 3.1, See section 3.2, See section 3.3, See section 4.3).
   Note: All statistics are upgraded in real time as the capture / drop progresses. That means that is not really necessary to wait for the current capture / drop to finish to check the results.
6. Download the capture by removing the SD card from the unit (*Net.Shark* only) or using the web interface (See section 5.3, See section 6.1.6).

1.5. Upgrading the Unit

The unit software can be upgraded with the help of a USB memory stick. Before proceeding with the upgrade copy the ALBEDO software to the root directory in the memory stick. The file name of the upgrade package (*albedo-tap.pki* or *albedo-hunter.pki* depending on the model) must not be modified. The USB must have a FAT32 file system.

Once the USB memory stick is ready. Follow this procedure to install the new software:

1. Switch the unit off
2. Press HOME and ENTER simultaneously and, without releasing the keys, press the On / Off button.
3. Now, keeping all three the keys pressed, wait until you hear a beep. Then release the keys.
   The ALBEDO Software Installer is loaded and executed. An informative panel displays the Net.Shark / Net.Hunter software version number found in the storage device.

4. Press ENTER to continue with the installation process.
5. Select *Install* or *Upgrade. Install* regenerates all the software in the unit even if it is up to date. Upgrade regenerates only the software that has changed since the last upgrade. Use *Install* (F2 key) if you need to recover the unit after operation failure due to corrupted software. Use *Upgrade* (F1 key) otherwise.
6. Confirm your previous selection by pressing ENTER or cancel with ESC.
7. Wait for the installation process to finish.
   Note: The full process may take a few minutes.
   Note: Do no disconnect the unit or remove the USB memory stick during installation.
8. Press ENTER to close the Software Installer and finish the installation process. The unit will be automatically restarted. The new software will be loaded.

Chapter 2
Connection to the Network

The Net.Shark /Net.Hunter is equipped with two 1 Gb/s SFP ports and two 1 Gb/s RJ-45 ports. The SFP ports, or line interfaces A and B, are generally connected in through mode to the link or network to be analysed. If used, the RJ-45 ports, or mirror interfaces A and A / B, connect Net.Shark / Net.Hunter with an external storage device or a protocol analyser.

It is important to notice that although line ports and mirror ports are both labelled with the letters A and B, they are not interchangeable. Line and mirror interfaces have a fixed role and their purpose cannot be configured by users.

This chapter describes how to connect the equipment to the network and how to configure it to receive and send signals. The general procedure to do that is:

1. Connect the line ports to the network under test. Use the electrical or optical SFP modules depending on the particular transmission medium.
2. If necessary, connect the mirror interface A and mirror interface A / B in order to forward the matching traffic to an external device (protocol analyser or external storage).

2.1.Operation Modes

Net.Shark / Net.Hunter works as an Ethernet tap / capture device in pass through connection mode or it simply selects (filters) some traffic with specific properties in endpoint connection mode. The tap and the filter modes constitute two separate configuration modes in the equipment.

The procedure to configure Net.Shark / Net.Hunter as a tap or as a filter is as follows:

1. From the *Home* panel, go to *Setup*,
   The test port settings panel is displayed.

2. Go to *Tap* and select *On* (operation as a tap connected in through mode) or *Off* (operation as a filtering device connected in endpoint mode).

Table 2.1: Net.Shark / Net.HunterTap Configuration

Setting	Description
On	Traffic is forwarded between the line ports A and B without any modification or delay. Filtered traffic is forwarded towards the mirror ports or an storage device. It is required at least one common supported bit rate for line port A and line port B when this mode is configured
Off	Traffic from the line ports is filtered and forwarded towards the mirror ports or an storage device. No traffic is forwarded between line port A and line port B. No common supported bit rate in port A and port B is required when this port is configured. It is even possible to connect only one line port to the network.

2.2. Traffic Aggregation and Storage

Net.Shark / Net.Hunter can be configured to aggregate traffic from the forward and backward transmission directions and present them as a single stream. This kind of stream aggregation is useful to check interactions between the communication ends like for example requests and replies in a web application. However, if the aggregated bandwidth is higher than the mirror channel capacity, some frames will be lost (*OverL* event). To avoid this problem you can disable the traffic aggregation.

Table 2.2: Net.Shark / Net.Hunter Operation Modes

Setting	Description
Mirror	Matching traffic from line port A is forwarded to mirror port A and matching traffic from line port B is forwarded to mirror port B.
Mirror & aggregate	Matching traffic from line ports A and B is forwarded to mirror port B.
Store	Matching traffic from line ports A and B is forwarded to the SD card (Net.Shark) or the high speed internal storage (Net.Hunter). Note: Before installing an SD card in the tester it is recommended to switch the equipment off and connect it again once the card is correctly installed in its slot.

Also, matching traffic does not need to be forwarded to a mirror port. As an alternative, traffic is captured and stored in an SD card (Net.Shark) or high speed internal storage (Net.Hunter). If the SD card is used for storage, matching traffic throughput is limited to 3.2 Mb/s. Maximum throughput for the high speed storage (Net.Hunter only) is 1 Gb/s + 1 Gb/s, equivalent to the maximum line interface nominal speed.

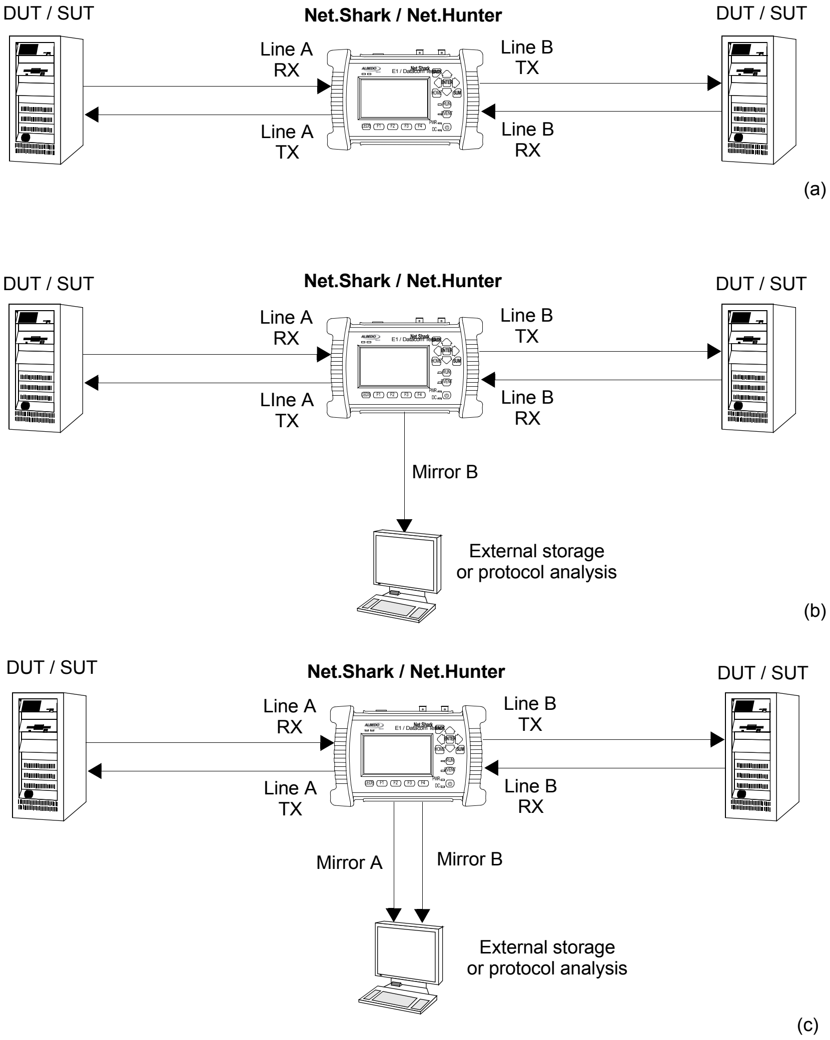


Figure 2.1: ALBEDO Net.Shark connection when operating in *Tap & Filter* mode: (a) Capture with storage in internal memory, (b) Capture and aggregation to mirror port A, (c) Capture without aggregation.

This is the procedure to follow to configure traffic aggregation or traffic storage to the SD card (*Net.Shark*) or the internal drive (*Net.Hunter*):

1. From the *Home* panel, go to *Setup*,
   The test port settings panel is displayed.
2. In *Mode*, select either *Mirror* (drop traffic without aggregating transmission directions), *Mirror & aggregate* (drop and aggregate matching traffic) or *Store* (save matching traffic to the SD card or the high speed internal storage).

2.3. Operation in Tap & Filter Mode

The right way to connect the equipment for capturing or mirroring Ethernet traffic is generally in pass through mode, allowing the traffic to go through the tester. If the equipment is to be connected to the network through optical interfaces, use optical SFPs. For an electrical capture / drop action, use electrical SFPs. Use always the line ports (SFP ports) for connection to the DUT / SUT.

Net.Shark / Net.Hunter operation is bi-directional. That means that both transmission directions are simultaneously processed by the equipment.

2.3.1. Checking LIne Port A and B Auto-negotiation

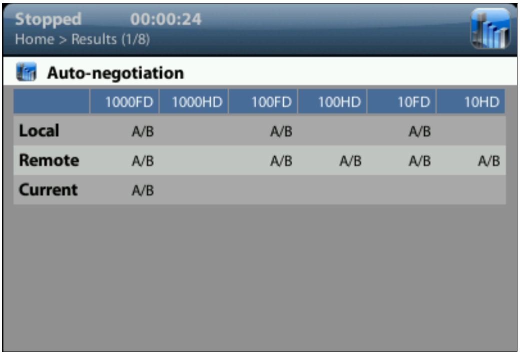
Interfaces connected to Line A and Line B (SFP ports) are required to have at least one common operation bit rate. Also, interfaces to be connected to ports Line A and Line B must support full duplex operation.

If these two conditions are met, Net.Shark / Net.Hunter will be able to forward traffic between ports A and B. Bit rate configuration can be automatic using Ethernet auto-negotiation but users are allowed to force a fixed bit rate as well. If auto-negotiation is enabled in Net.Shark / Net.Hunter, the equipment will configure its ports to the common maximum rate for port A and port B. If auto-negotiation is not enabled, The user will configure the desired bit rate, but Port A and Port B configuration are always coupled so that the bit rate in both ports always remains the same.

To configure the auto-negotiation settings in your Net.Shark / Net.Hunter unit follow these steps:

1. From the *Home* panel, go to *Setup*,
   The port setup panel is displayed.
2. Go to *Port A* or *Port B*.
   The port specific configuration menu is displayed.
3. Go to *Line port auto-negotiation*.
4. Set *Enable auto-negotiation* to *Yes* to configure the link speed using auto-negotiation or to *No* to disable auto-negotiation and force the link speed to a fixed value
5. If you have enabled auto-negotiation in the previous step configure the allowed bit rates through the *1000*, *100* and *10* menus. If auto-negotiation is disabled, set the *Forced bit-rate* to *10 ,100, 1000*.

Note: Forced 1000 Mb/s bit rate is available only through suitable optical SFP transceivers. This mode can be used for monitoring in already established optical link through a passive splitter.. In all other test setups, the 1000 Mb/s bit tare cannot be forced and it is only available through auto-negotiation.
Note: If you are using 100 Mb/s optical interfaces (100BASE-FX), the operation is always forced and the bit rate is fixed to 100 Mb/s. You are not allowed to enable auto-negotiation in this interface.



Auto-negotiation

	1000FD	1000HD	100FD	100HD	10FD	10HD
Local	A/B		A/B		A/B	
Remote	A/B		A/B	A/B	A/B	A/B
Current	A/B					

Figure 2.2: ALBEDO Net.Shark / Net.Hunter auto-negotiation results panel.

Once the tester has been connected to the network and the right connector type as been configured, follow these steps to check auto-negotiation:

1. From the *Home* panel, go to *Results*,
   The test port results panel is displayed.
2. Press *Line port auto-negotiation* to display the auto-negotiation for line ports A and B.

Table 2.3: Auto-negotiation results

Result	Description
Local	Displays the bit rate and duplex mode supported by the Net.Shark Port A or Port B. If you are using electrical SFPs, the supported bit rates are 10 Mb/s, 100 Mb/s and 1000 Mb/s (1000FD, 100FD and 10FD). If the SFP connector is optical, the supported bit rate is 1000 Mb/s (1000FD) or 100 Mb/s (100FD), depending on the particular SFP you are using. Net.Shark / Net.Hunter requires full-duplex operation. For this reason, the equipment always forces full-duplex mode.

Table 2.3: Auto-negotiation results

Result	Description
Remote	Displays the bit rate and duplex mode supported by the remote device connected to Port A or Port B. It is one or several of the 1000FD, 1000HD, 100FD, 100HD, 10FD, 10HD set. If the remote device does not support auto-negotiation or auto-negotiation is disabled in this device, it will behave as if no interfaces were supported and the equipment will be unable to work.
Current	Bit rate and duplex operation agreed during the auto-negotiation process. It is one (and only one) of the 1000FD, 100FD and 10FD set. If there is more than one compatible interfaces, the one with higher bit rate is preferred.

3. If you are using line port A, check that there is an “A” shown in the *Current* row in the auto-negotiation table. If you are using line port B, check that there is a “B” shown in the *Current* row.
   Note: If the equipment is configured in *Tap & Filter* mode, both line ports are required to operate at the same time with the same bit rate. In this case, “A/B” should be displayed in the *Current* row for some interface.

2.3.2. Checking Mirror Port A and B Auto-negotiation

If used, mirror ports also auto-negotiate the optimum bit rate and duplex mode. Operation for mirror ports is not as strict as for the line ports. Specifically, they don’t need to negotiate the same operation speed and they not even need to be used at the same time. For example, if traffic aggregation is configured, then mirror port A is not used and there is no need to connect it.

The mirror port auto-negotiation configuration panels are similar to the equivalent line port auto-negotiation panel. Follow these steps to configure auto-negotiation in the mirror ports:

1. From the *Home* panel, go to *Setup*,
   The port setup panel is displayed.
2. Go to *Port A* or *Port B*.
   The port specific configuration menu is displayed.
3. Go to *Mirror port auto-negotiation*.
4. Set *Enable auto-negotiation* to *Yes* to configure the link speed using auto-negotiation or to *No* to disable auto-negotiation and force the link speed to a fixed value
5. If you have enabled auto-negotiation in the previous step configure the allowed bit rates through the *1000*, *100* and *10* menus. If auto-negotiation is disabled, set the *Forced bit-rate* to *10* or *100*.

Note: The 1000 Mb/s rate cannot be forced and it is only available through auto-negotiation due to IEEE 802.3 restrictions.

There is a auto-negotiation result table for mirror ports which contains similar information that the one for line ports. Follow these steps to display this result table:

1. From the *Home* panel, go to *Results*,
   The test port results panel is displayed.
2. Press *Mirror port auto-negotiation* to display the auto-negotiation for mirror ports A and B.
3. If you are using mirror port A, check that there is an “A” shown in the *Current* row in the auto-negotiation table. If you are using mirror port B, check that there is a “B” shown in the *Current* row.

2.3.3. Using the SFPs for the Line Interfaces

Net.Shark / Net.Hunter has two SFP ports to connect the equipment to the link to be analysed. They can be used for both for electrical and optical operation if compatible SFPs are connected.

Table 2.4: Ethernet SFP Results

Result	Description
SFP present	Shows information about presence of an SFP in the current port. The information is displayed even if the port is configured to operate over the attached RJ-45 interface
Interface	• 10BASE-T: Used for transmission at 10 Mb/s over two pairs of Cat. 3 UTP cable with range of 100 m. • 100BASE-TX: Used for transmission at 100 Mb/s over two pairs of Cat. 5 UTP cable with range of 100 m. • 1000BASE-T: Used for transmission at 1000 Mb/s over four pairs of Cat. 5e UTP cable with range of 100 m. • 100BASE-FX: Optical interface for transmission of 100 Mb/s over MMF operating in the 1310 nm optical window. This interface is available by an external compatible SFP supplied by ALBEDO Telecom. • 1000BASE-SX: Used for transmission at 1000 Mb/s over two MMF operating in the 850 nm optical window. Ranges are usually a few hundred metres. • 1000BASE-LX: Used for transmission at 1000 Mb/s over two MMF or SMF in the 1310 nm optical window. Ranges use to be a few kilometres.

Table 2.4: Ethernet SFP Results

Result	Description
SFP vendor	If there is an SFP connected to the port, this field shows information about the vendor. This information is recorded within a memory in the SFP when it is manufactured.
SFP part number	If there is an SFP connected to the port, this field shows information about the vendor. This information is recorded within a memory in the SFP when it is manufactured.

To display the SFP interface information follow these step sequence:

1. From the *Home* panel, go to *Results*,
   The test port results panel is displayed.
2. Enter in *SFP*.
3. Check the *Interface*, *SFP present*, *SFP vendor* and *SFP part number*.

2.4. Operation in Filter-only Mode

Use the *Filter* operation mode when you don't need to monitor both transmission directions in through mode or when you have to perform bidirectional monitoring but joint line port A and B auto-negotiation cannot be achieved.

The *Filter* mode does not require that the line port A and line port B operate at the same bit rate. It is even allowed operation from a single line port. However, if this mode is selected, traffic will not be forwarded between line ports A and B. In fact, transmission is disabled for these ports in *Filter* mode.

Configuration and operation of the equipment in *Filter* mode is not very different than in *Tap & Filter* mode but the equipment is connected to the network in a different way. In filter mode the traffic does not pass through the Net.Shark / Net.Hunter line ports, for this reason it has to be connected so that it does not interrupt the network traffic. Usually this is accomplished by mirroring the traffic to an spare port in a switch or router.

The port mirroring capabilities of routers and switches may or may not have aggregation of transmission directions. If port mirroring with aggregation is not available in the switch or router, aggregation can still be achieved with Net.Shark /

Net.Hunter. To do that, connect the traffic to be aggregated to line ports A and B and configure *Mirror & aggregate* (or *Store*) in *Mirror mode*.

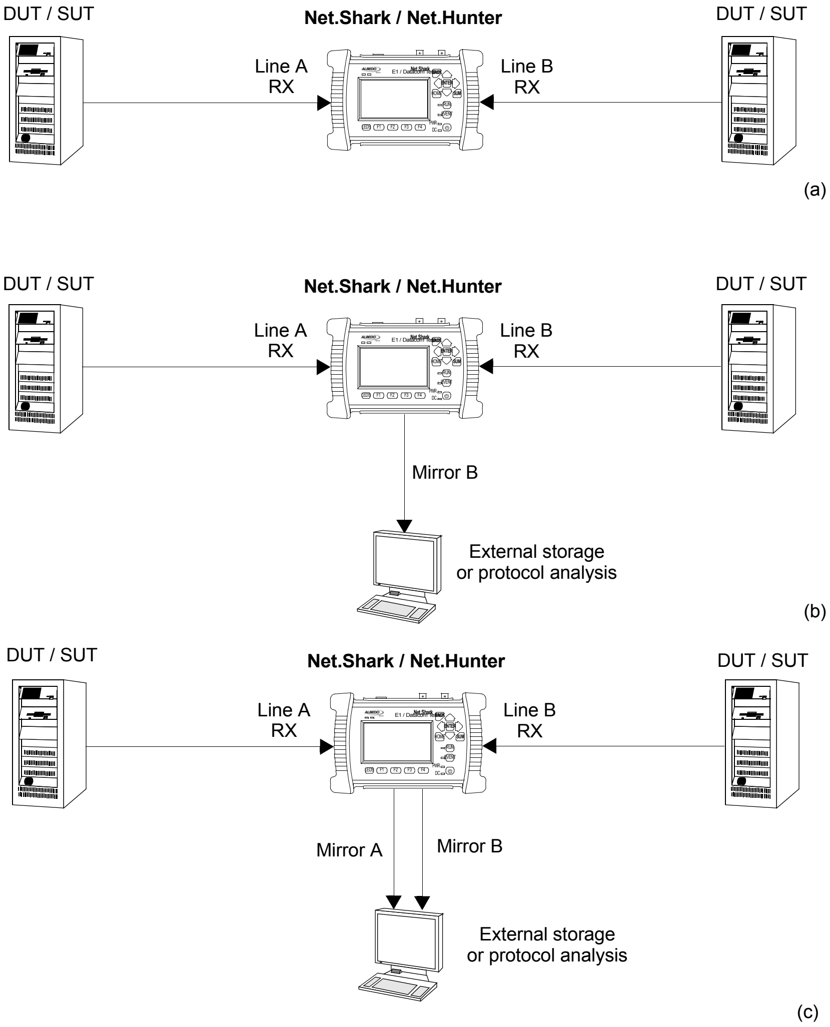


Figure 2.3: ALBEDO Net.Shark connection when operating in *Filter* mode: (a) Capture with storage in internal memory, (b) Capture and aggregation to mirror port A, (c) Capture without aggregation.

2.5. USing the PoE / PoE+ Bridge

It could be desirable to install Net.Shark / Net.Hunter between a PoE or PoE+ *Power Sourcing Equipment* (PSE) and a PoE / PoE+ *Powered Device* (PD) without interrupting the DC current and voltage required to keep the PD operation. To achieve this objective, Net.Shark / Net.Hunter can be configured to allow a DC power signal to be transmitted between the line ports.

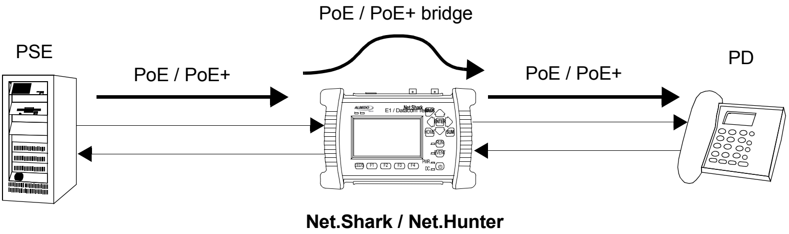


Figure 2.4: PoE / PoE+ bridge operation in Net.Shark / Net.Hunter.

When the PoE / PoE+ bridge is enabled in the unit, the functionality of *Line A* port is swapped with *Mirror A* and *Line B* is swapped with *Mirror A/B*. That means, that the network is connected to Net.Shark / Net.Hunter through the *Mirror A* and *Mirror B* ports rather than through the line ports as usual. In this operation mode any filtered signal is dropped to Line A / Line B ports or copied to an storage device.

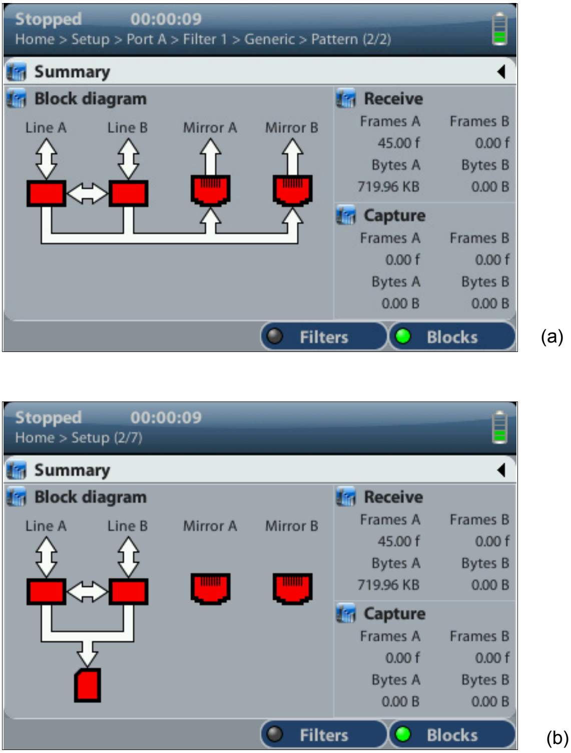
In order to configure the PoE / PoE+ bridge in your unit follow these steps:

1. From the *Home* panel, go to *Setup*,
   The test port settings panel is displayed.
2. Go to *POE Bridge* and select *On* to enable DC power signal pass-through and line / mirror port swapping or *Off* to return to the default operation mode.

The PoE / PoE+ bridge functionality is an optional feature for Net.Shark and Net.Hunter that is implemented as a hardware option. The *POE Bridge* menu is not available in units without the hardware supporting this feature.

2.6. Using the Block Diagram Panel

The Net.Shark / Net.Hunter block diagram is available through the summary (SUM) button. The block diagram summarizes the incoming and outgoing traffic flows associated to Net.Shark / Net.Hunter. The block diagram is different for different operation modes because the traffic flows are modified when the mode varies. For example, if aggregation is configured (*Mirror & aggregate* mode) the traffic flow to mirror port A is suppressed.



Stopped 00:00:09
Home > Setup > Port A > Filter 1 > Generic > Pattern (2/2)

Summary

Block diagram

Receive

Frames A	Frames B
45.00 f	0.00 f
Bytes A	Bytes B
719.96 KB	0.00 B

Capture

Frames A	Frames B
0.00 f	0.00 f
Bytes A	Bytes B
0.00 B	0.00 B

(a)

Stopped 00:00:09
Home > Setup (2/7)

Summary

Block diagram

Receive

Frames A	Frames B
45.00 f	0.00 f
Bytes A	Bytes B
719.96 KB	0.00 B

Capture

Frames A	Frames B
0.00 f	0.00 f
Bytes A	Bytes B
0.00 B	0.00 B

(b)

Figure 2.5: Two examples of Net.Shark / Net.Hunter block diagram panel. (a) Connectors and traffic flows when the equipment is configured in *Drop* mode. (b) Connector usage and traffic flows when the equipment is configured in *Capture* mode.

In order to display the block diagram follow these steps:

1. Press the summary (SUM) key.
   The *Summary* panel is displayed.
2. Press the *Blocks* contextual button (F4).
   The block diagram is displayed.

Chapter 3
Traffic Statistics

The ALBEDO Telecom Net.Shark / Net.Hunter can be used to get basic traffic statistics about Ethernet networks operating at rates up to 1 Gb/s. These statistics include frame and error counts.

Statistics for line Port A and Port B are identical. Some statistics are referred to the traffic detected in the line ports, some others correspond to the monitored frames, including frames dropped to the primary or secondary mirror interfaces or stored in storage media. The next sections provide details about how to use the equipment to get statistics for the line and mirror interfaces.

3.1. Mirrored Frame Analysis

Drop / Capture statistics account for the frames forwarded towards a mirror port or storage device after being filtered by one or more filtering blocks (See chapter 4).

Drop / Capture frame statistics are controlled by the RUN button. That means that results are not collected if a capture / drop action is not started before. Once the capture / drop action is running results are upgraded in real time.

To display the statistics referred to the traffic forwarded to specific mirror interface or storage device, follow this procedure:

1. From the *Home* panel, go to *Results*,
   The general results and statistics menu is displayed.
2. Enter in *Mirror frame analysis*.
3. Check the *Frames*, *Bytes*, *Broadcast frames*, *Multicast frames*, *Control frames, Tagged frames* and *FCS errored frames.*
   Note: Each field contains one counter for frames coming from line port A and one accounting for frames received from line port B.

4.

Table 3.1: Mirror Frame Analysis

Metric	Description
Frames	Total number of frames stored / transmitted by one mirror port since the capture / drop action started.
Bytes	Total byte count stored / transmitted by the mirror port from the beginning of the capture / drop. One byte is defined as an 8-bit word.
Broadcast frames	Total number of Ethernet broadcast frames transmitted / stored from the beginning of the capture / drop. Broadcast frames carry the broadcast Ethernet address (*FF:FF:FF:FF:FF:FF*) in the destination field.
Multicast frames	Transmitted / stored Ethernet multicast frames from the beginning of the capture / drop. Ethernet multicast frames have their multicast bit in their destination MAC address set to ‘1’. The multicast bit of a MAC address is the least significant bit of the more significant address byte.
Control frames	Total number of Ethernet MAC control and supervision frames transmitted to a mirror port or stored in an internal / external storage device from the beginning of the capture / drop. Ethernet control frames are recognised due to an special Ethertype (Type / Length field) value (*0x8808*).
Tagged frames	Total number of Ethernet VLAN frames transmitted to a mirror port or stored in an internal / external storage device from the beginning of the capture / drop. IEEE 802.1Q VLAN frames contain an special Ethertype (Type / Length field) value (*0x8100*).
FCS errored frames	Count of all the FCS errors transmitted to a mirror port or stored in an internal / external storage device from the beginning of the capture / drop. A frame with a FCS error is a frame with a legal size which contains an invalid FCS field. FCS errors are caused by transmission errors. An optical Ethernet link with a poor power budget may experience FCS errors.

3.2. Line Port Frame Statistics

The procedure to get the statistics corresponding to the line interface is similar to the mechanism already described for mirror ports. These are the steps to follow:

1. From the *Home* panel, go to *Results*,
   The general results and statistics menu is displayed.
1. Enter in *RX frame analysis*.
2. Check the *Frames*, *Bytes*, *Broadcast frames*, *Multicast frames*, *Control frames*, *Flow control frames*, *Tagged frames*, *Oversized frames*, *Undersized frames*, *Jabbers* and *FCS errors* counters.
   Note: Each field contains one counter for line port A frames and the second one for line port B.

Net.Shark does not automatically discards frames with errors. If configured in pass-through mode, errored frames are forwarded if possible. Errored frames are also dropped or stored whenever possible.

Table 3.2: RX Frame Analysis

Metric	Description
Frames	Total number of frames received by one equipment line port since the capture / drop action started.
Bytes	Total byte count received by the current line port from the beginning of the capture / drop action. One byte is defined as an 8-bit word.
Broadcast frames	Total number of Ethernet broadcast frames received from the beginning of the capture / drop. Broadcast frames carry the broadcast Ethernet address (*FF:FF:FF:FF:FF:FF*) in the destination field.
Multicast frames	Received Ethernet multicast frames from the beginning of the capture / drop. Ethernet multicast frames have their multicast bit in their destination MAC address set to ‘1’. The multicast bit of a MAC address is the least significant bit of the more significant address byte.
Control frames	Total number of Ethernet MAC control and supervision frames received from the beginning of the capture / drop. Ethernet control frames are recognised due to an special Ethertype (Type / Length field) value (*0x8808*).

Table 3.2: RX Frame Analysis

Metric	Description
Flow control frames	Total number of Ethernet *Pause* frames received from the beginning of the capture / drop. Pause frames are an special type of control frames and therefore their Ethertype is 0x8808. The specific features of *Pause* frames is that their *Opcode* field is 0x0001 and their destination MAC address is *01:80:C2:00:00:01* (a multicast MAC address).
Tagged frames	Total number of Ethernet VLAN frames received from the beginning of the capture / drop. IEEE 802.1Q VLAN frames contain an special Ethertype (Type / Length field) value (*0x8100*).
FCS errored frames	Count of all the FCS errors detected from the beginning of the capture / drop. A frame with a FCS error is a frame with a legal size which contains an invalid FCS field. FCS errors are caused by transmission errors. An optical Ethernet link with a poor power budget may experience FCS errors
Oversized frames	Total number of received frames which are larger than the configured MTU.
Undersized frames	Total number of received frames which are smaller than 64 bytes.
Jabbers	Jabber count from the beginning of the capture / drop. A Jabber is defined as a frame greater than 1518 bytes with a bad CRC.

3.3.Size Histogram

Frame size is important because it tells the way the network is used. Some applications, like VoIP use short frames while most data applications based on a client / server use short frames length for the client requests and long frames for the server replies. The ALBEDO Telecom Net.Shark / Net.Hunter provides frame size results as described in standard RFC 2819. The procedure for displaying the received frame size histogram is as follows:

1. From the *Home* panel, go to *Results*,
   The general results and statistics menu is displayed.
1. Enter in *Frame size histogram*.

2. Check the frame size intervals: *size or less*, 65 - 127, *128-255*, *256 - 511*, *512-1023*, *1024 - 1518*.
   Note: Each field contains one counter for frames coming from line port A and one accounting for frames received from line port B.

3.4. The LEDs Panel

The LEDs panel offers a quick view of the current Net.Shark / Net.Hunter connection and operation status. They are permanent indicators. That means that no action has to be started to get the information from the LEDs.

There are two hardware global summary LEDs in the equipment (one for Port A and one for Port B), three summary LEDs for each line port (*Link*, *Frame*, and *Error*), and three summary LEDs for each mirror port (*Link*, *Frame* and *Error*). These summary LEDs summarize the information of the events shown in the LEDs panel. To display the LEDs panel use the LEDS key. If the LEDs panel is already visible press LEDS again to return to the previous screen.

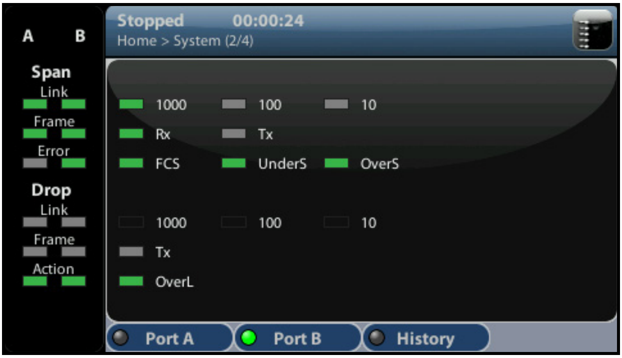


Figure 3.1: ALBEDO Net.Shark / Net.Hunter LEDs panel.

The LEDs have two operation modes:

- *Live*: Events are shown in real time. If something happens the corresponding LEDs change their colour to signal the event. LEDs return to their original status once the event disappears.
- *History*: The LEDs keep their original Anomaly / Defect status when the event disappears. This is useful when the equipment is left a long time under operation and the user wants to receive quick feedback of past events.

The live or history modes can be configured from the LEDs panel by means the contextual keyboard. The *History* (F3) contextual button sets or unsets the history mode. If the history mode is enabled, then the Reset (F4) button resets the LEDs history.

It follows a description with the possible LED status:

- 🟩: OK, the event or events that correspond with the LED are not found in the incoming signal.
- 🟧 and 🟥: This is the colour displayed if faulty conditions are found in the signal. Conditions marked with 🟥 tend to be more important than the ones marked with 🟧.
- ▨: Shows that no operation condition can be established due to the lack of matching traffic for the corresponding event. For example, the FCS LED is ▨ if no traffic is received because there are no frames where to check the FCS. It can also indicate that the LED has been disabled due to the presence of a more important event.

Table 3.3: Line Port LED Indications

Metric	Description
1000	The line port is operating at 1000 Mb/s. Port speed is decided immediately after connecting the port to the line interface using Ethernet auto-negotiation.
100	The line port is operating at 100 Mb/s. Port speed is decided immediately after connecting the port to the line interface using Ethernet auto-negotiation (if supported by the SFP) or it is forced if the user decides to do so..
10	The line port is operating at 10 Mb/s. Port speed is decided immediately after connecting the port to the line interface using Ethernet auto-negotiation or it is forced if the user decides to do so.
Rx	At least one frame was received during the current second in the current interface.
Tx	At least one frame was transmitted during the current second in the current interface. Transmitted frames always come from the second line port receiver. No frames are internally generated by Net.Shark

Table 3.3: Line Port LED Indications

Metric	Description
FCS	At least one frame with FCS errors have been found during the current second. A frame with a FCS error is a frame with a legal size which contains an invalid FCS field. FCS errors are caused by transmission errors. An optical Ethernet link with a poor power budget may experience FCS errors
Jabber	At lease one jabber has been received during de current second. A Jabber is defined as a frame greater than 1518 bytes with a bad CRC.
UnderS	At least one undersized frame was received during the current second. An undersized frame is a frame which has a size smaller than 64 bytes.
OverS	At least one oversized frame was received during the current second. An oversized frame is a frame which has a size larger than the configured MTU.

Some Net.Shark / Net.Hunter LEDs are referred to the line ports and some others to mirror ports / captures. There are currently nine LED for line ports (*1000*, *100*, *10*, *Rx*, *Tx*, *FCS*, *Jabber, UnderS* and *OverS*) and six for mirror ports (*1000*, *100*, *10*, *Tx*, *OverL*)

Table 3.4: Mirror Port LED Indications

Metric	Description
1000	The mirror port is operating at 1000 Mb/s. Port speed is decided immediately after connecting the mirror port to the Ethernet line using Ethernet auto-negotiation.
100	The port is operating at 100 Mb/s from the electrical port. Port speed is decided immediately after connecting the port to the line interface using Ethernet auto-negotiation.
10	The port is operating at 10 Mb/s from the electrical port. Port speed is decided immediately after connecting the port to the line interface using Ethernet auto-negotiation.

Table 3.4: Mirror Port LED Indications

Metric	Description
Tx	At least one frame was transmitted during the current second in the current mirror interface. Note: This LED applies to captures in the same way that it does for drop actions. In a capture, this LED indicates that at least one frame has been sent to the capture device during the current second.
OverL	Overload event. The equipment is unable to send towards mirror port A or mirror port B all the traffic received in the line ports because there is not enough capacity. This event may happen when the equipment forwards a high amount of traffic to a low capacity drop channel. The OverL can be avoided if the equipment is configured to operate without aggregation and if the mirror interfaces are configured to operate at the maximum possible speed. This event applies to captures to SD channels. An SD capture is equivalent to a 3.2 Mb/s drop channel. The 3.2 Mb/s limit is used to avoid overloading the capture software.
Dmp	The capture device is unable to store all the traffic it receives. Due to insufficient processing capacity, not enough space left on device or any other reason.

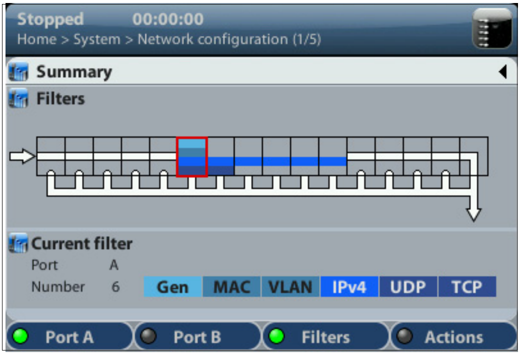
Chapter 4
Filtering

Net.Shark / Net.Hunter is capable of processing and computing statistics over fractions of the Ethernet traffic meeting specific conditions. The process of selecting a fraction of traffic is called filtering. The result of the filtering process is one or several traffic streams called flows.

This chapter describes how to configure Net.Shark / Net.Hunter for packet filtering and how to get basic flow statistics.

4.1.Enabling and Disabling Filters

Traffic selection or filtering is configured by first enabling one or several filtering blocks and after that setting the filtering criteria. Net.Shark supports Ethernet, VLAN, IPv4, IPv6, TCP / UDP, fixed offset, fixed pattern and fixed length filters. Net.Hunter add to the previous ones specific filters for C-VLAN and S-VLAN tags in Q-in-Q frames (IEEE 802.1ad).



Stopped 00:00:00
Home > System > Network configuration (1/5)

Summary

Filters

Current filter

Port A

Number 6

Figure 4.1: Filter setup panels. Filter status can be checked from this panel. It is also possible

The ALBEDO Telecom Net.Shark / Net.Hunter is equipped with 16 independent filters. Each filter has a priority number. If one frame is selected by an specific filter it will not be processed by any lower priority filters.

Current status of all filters can be checked at any moment with the help of the filter summary panel by pressing the SUM key.

4.2. Configuring Filters

If a filter is left with its default configuration, it will not accept any frame. To allow the filter to accept and process frames, a correct filtering criteria must be configured before.

To configure the correct filtering criteria two decisions must be taken. First, it is necessary to know which frame fields are going to be matched and after that, which are the value or values to be matched. The first decision involves choosing whether the filtering is going to be done at MAC, IP or transport layer and which specific frame field or fields are going to be used for filtering (MAC addresses, IP addresses, Ethertype field, protocol or any other). The second is carried out by configuring the field value and sometimes a mask. The mask selects which field bits are taken into account when a frame is matched. Matching masks are not related to IP subnet masks even if they can be applied to IP addresses. Specifically, the binary representation of a matching mask does not need to be a sequence of ‘1’ followed by a sequence of ‘0’ like IP network masks are.

The generic procedure to configure one or several matching rule with Net.Shark / Net.Hunter is the following:

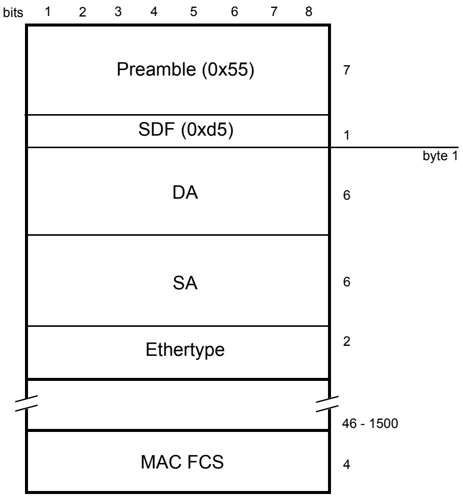
1. From the *Home* panel, go to *Setup*,
   The test port settings panel is displayed.
2. Select either *Port A or Port B* to enter in the port specific configuration.
3. Select one of the filtering menus labelled as *Filter 1*, *Filter 2*, etc.
4. Enable the filter by setting the *Block* field to the value *No*.
   Different types of filtering criteria families are enabled for that filter: *Generic*, *MAC*, *C-VLAN, S-VLAN, IPv4*, *IPv6*, *TCP*, *UDP*.
5. Select the matching rule.
6. Choose a matching field and configure de matching mode for this field. Most of the matching fields have at least two matching modes. For example, the *Equal* mode selects frames matching the configured value or values for the field.
7. Configure the field value to be matched by the filter.
8. If the matching field has this capability, enter the mask value. To select a single value, set of the mask bit values to all ones.
9. Optionally, configure more matching rules for the current filter by repeating steps 3, 4, 5, 6, 7 and 8 as many times as necessary.

Quick information about currently matching rules for each filter is available in the *Summary* panel.

4.2.1. MAC Selection

MAC frames are envelopes in which the Ethernet frames are sent and received. MAC frame format is currently specified by the standard IEEE 802.3. This format is shared by all existing Ethernet interfaces thus making Ethernet the most scalable transmission technology currently available.

Preamble: Synchronization pattern
SDF: Start Frame Delimiter
DA: Destination MAC Address
SA: Source MAC Address
Ethertype: Length / Type Field
MAC FCS: Frame Check Sequence



IEEE 802.3 Ethernet MAC Frame

Figure 4.2: IEEE 802.3 frame structure

The ALBEDO Telecom Net.Shark / Net.Hunter provides frame selection based on the MAC address source and destination and Ethertype value. It is possible to configure a matching mask for all three fields to select a value set rather than a single value.

Table 4.1: MAC Selection

Parameter	Description
Source address match	Enables selection by source MAC address in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are:

Table 4.1: MAC Selection

Parameter	Description
	• *Ignore*: The source MAC address is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal with mask*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Source address* and *Source address mask* fields.
Source address	This is a 48-bit MAC address in the standard hexadecimal-digit format *XX:XX:XX:XX:XX:XX*. Source addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching frames to this filter are configured in the *Source address mask*.
Source address mask	This is the mask for the source MAC address filter selection rule. Before comparing the *Source address* field with the frame addresses, bit wise AND operations are carried out between the value configured here and the *Source address* field so that only the values surviving the AND are taken into account for matching the filter.
Destination address match	Enables selection by destination MAC address in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source address selection* setting.
Destination address	This is a 48-bit MAC address in the standard hexadecimal-digit format *XX:XX:XX:XX:XX:XX*. Destination addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching frames to this filter are configured in the *Destination address mask*.
Destination address mask	This is the mask for the destination MAC address filter selection rule. Before comparing the *Destination address* field with the frame addresses, bit wise AND operations are carried out between the value configured here and the *Destination address* field so that only the values surviving the AND are taken into account for matching the filter.

Table 4.1: MAC Selection

Parameter	Description
Ethertype match	Enables selection by Ethertype value in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source address selection* and *Destination address selection* settings.
Ethertype	This setting contains a 2-byte field that constitutes the Ethertype value to be matched in the incoming traffic. Ethertypes matching some or all bytes of the value configured here will be allowed to pass through the filter. Bits used for matching frames to this filter are configured by means the *Ethertype mask*.
Ethertype mask	This is the mask for the Ethertype filter selection rule. Before comparing the *Ethertype* field with the frame Ethertype, bit wise AND operations are carried out between the value configured here and the *Ethertype* field so that only the values surviving the AND are taken into account for matching the filter.

4.2.2. VLAN Selection

Within enterprise networks, VLANs are important because they enable network segmentation on an organisational basis, by functions, project teams or applications, rather than on a physical or a geographical basis. The network can be reconfigured through software, instead of physically unplugging and moving devices or wires.

VLANs are an important contribution to scalable Ethernet networks, because they limit broadcast traffic inherent to the bridging mechanism. Large amounts of broadcast traffic may damage performance and even collapse network equipment, which is why it must be controlled.

Standard IEEE 802.1Q specifies the most popular VLAN frame format. VLAN frames carry a 16-bit header which specifies the VLAN Identifier (VID) and the frame priority within the VLAN. Many carrier Ethernet networks use the VID for segmentation just like enterprises. The VID in carrier Ethernet networks is used by service providers as general purpose identifiers. They can be associated to an specific service, customer, node or several of them at the same time. Sometimes, service providers use a two-level VLAN structure. Levels are designated as customer VLAN (C-VLAN) and service

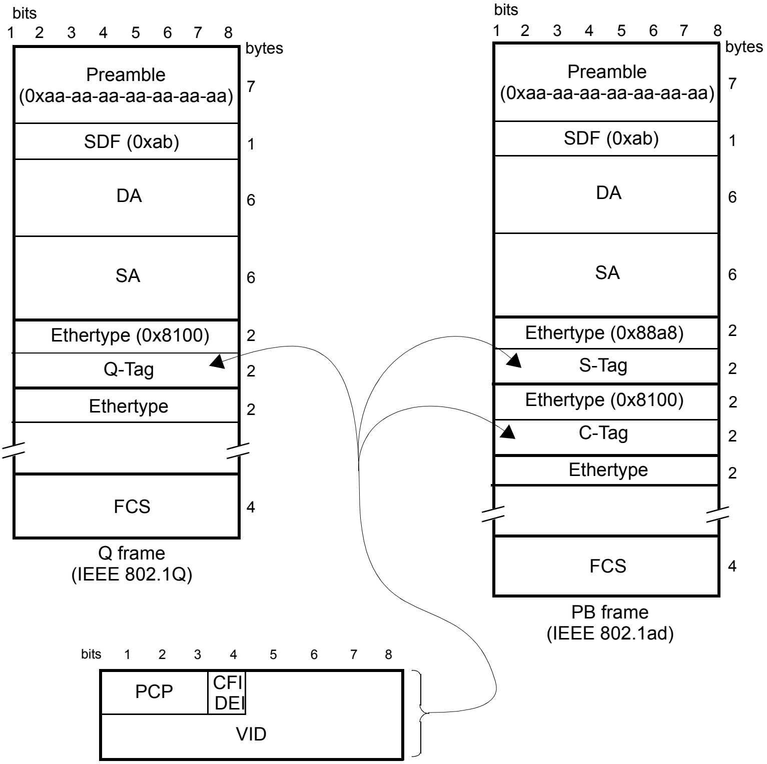


Figure 4.3: IEEE 802.1Q y IEEE 820.1ad frame structures.

VLAN (S-VLAN). This double VLAN structure is know as Q-in-Q. The standardised version of the Q-in-Q frame is defined in IEEE 802.1ad.

Table 4.2: C-VLAN and S-VLAN Selection

Metric	Description
VID match	Enables selection by VID in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: The VID is ignored and not taken into account for the purpose of the filter. No frame is selected by the filter when *Ignore* has been configured. • *Equal to*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *VID* field.

Table 4.2: C-VLAN and S-VLAN Selection

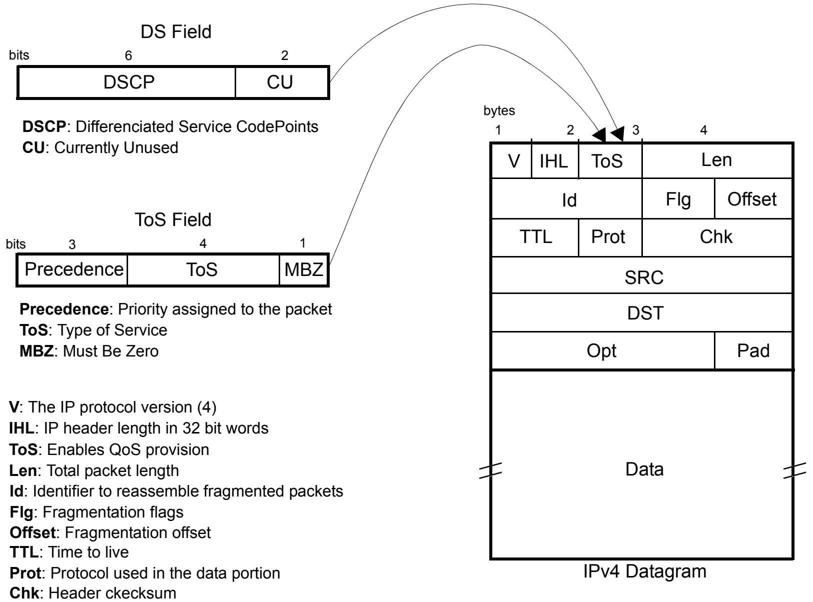
Metric	Description
VID	This setting contains a 10-bit identifier that constitutes the VID value to be matched in the incoming traffic. For Net.Hunter, it is possible to match the S-VID or the C-VID through separated entries in the filtering menu. For single-tagged frames it is assumed that the frame does not contain S-VID field and therefore configuration is done through the C-VLAN menu.
Priority codepoint match	Enables selection by IEEE 802.1Q/p priority bits in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *VID match* field.
Priority codepoint	This setting contains a 3-bit identifier that constitutes the priority value to be matched in the incoming traffic. For Net.Hunter, it is possible to match the S-VLAN or the C-VLAN priority codepoints through separated entries in the filtering menu. For single-tagged frames it is assumed that the frame does not contain S-VLAN priority codepoint field and therefore configuration is done through the C-VLAN menu.
Drop-eligible indicator match	Enables selection by the S-VLAN drop-eligible indicator in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and with ones are blocked. The available configuration values for this field are the same that for the *VID match* field.
Drop-eligible indicator	This is a single bit field that is used to signal which frames have precedence when the node has to drop some information due to congestion or other causes. The Drop-eligible indicator is defined only for the S-VLAN tag of double-tagged frames. For this reason is only available for S-VLAN matching rules.

Net.Shark provides frame selection based on the VLAN tag based either in the VID or the priority bits. Net.Hunter support both the single-level Q-frame and the two-level Q-in-Q-frame.

4.2.3. IPv4 Selection

The Internet Protocol version 4 (IPv4) was conceived by the U.S. *Department of Defence* (DoD) to facilitate communication between dissimilar computer systems and is a reliable technology. The IPv4 constitutes the most important protocol of the Internet and it is today widely used to connect heterogeneous packet networks everywhere.

IP is based on variable length data packets called datagrams. The datagram header includes two 32-bit addresses that identify the datagram source and destination and other fields with miscellaneous purposes.



DSCP: Differenciated Service CodePoints
CU: Currently Unused

Precedence: Priority assigned to the packet
ToS: Type of Service
MBZ: Must Be Zero

V: The IP protocol version (4)
IHL: IP header length in 32 bit words
ToS: Enables QoS provision
Len: Total packet length
Id: Identifier to reassemble fragmented packets
Flg: Fragmentation flags
Offset: Fragmentation offset
TTL: Time to live
Prot: Protocol used in the data portion
Chk: Header ckecksum
SRC: Source IPv4 address
DST: Destination IPv4 address
Opt: Options, variable length
Pad: Padding, fills out the 32 bit words
Data: Data, variable length

Figure 4.4: IPv4 Datagram structure

Net.Shark / Net.Hunter filtering capabilities can be programmed to match fields within the IPv4 datagram. It is currently supported IP datagram matching based on source IP address, destination IP address, protocol and DSCP. Source and destination IP addresses can be matched by means selection masks.

Table 4.3: IPv4 Selection

Parameter	Description
Source address match	Enables selection by source IPv4 address in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: The source IP address is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal with mask*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Source address* and *Source address mask* fields.
Source address	This is a 32-bit IPv4 address in the standard four-dotted decimal format *A.B.C.D*. Source addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching frames to this filter are configured in the *Source address mask*.
Source address mask	This is the mask for the source IPv4 address filter selection rule. Before comparing the *Source address* field with the frame addresses, bit wise AND operations are carried out between the value configured here and the *Source address* field so that only the values surviving the AND are taken into account for matching the filter.
Destination address match	Enables selection by destination IPv4 address in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source address selection* setting.
Destination address	This is a 32-bit IPv4 address in the standard four-dotted decimal format *A.B.C.D*. Destination addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching frames to this filter are configured in the *Destination address mask*.

Table 4.3: IPv4 Selection

Parameter	Description
Destination address mask	This is the mask for the source IPv4 address filter selection rule. Before comparing the *Source address* field with the frame addresses, bit wise AND operations are carried out between the value configured here and the *Source address* field so that only the values surviving the AND are taken into account for matching the filter.
Protocol match	Enables selection by the 1-byte IPv4 protocol field in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: The *Protocol* field is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Standard protocol selection* field and *Protocol* fields.
Standard protocol match	Configures the protocol to be filtered when protocol selection is enabled. The available configuration values are the following: • *UDP*: Matches traffic with a *User Datagram Protocol (UDP)* envelope. Traffic commonly transported over UDP includes IP voice, IP video and DNS. • *TCP*: Matches traffic carried over the *Transfer Control Protocol* (TCP). Most data applications (web, file transfer, e-mail...) ere normally based on TCP transport. • *ICMP*: *Matches Internet Control Message Protocol* (ICMP) packets. IP operation and maintenance traffic like ping use ICMP. • *Numeric*: Use this control if the traffic to be matched is different of UDP, TCP and ICMP and it has an specific protocol identifier assigned by the IANA.

Table 4.3: IPv4 Selection

Parameter	Description
Protocol	This setting contains an 8-bit word that constitutes the protocol identifier to be matched in the incoming traffic. Configuring this field to 17 is equivalent of setting the *Standard protocol selection* to UDP. TCP uses 6 as the protocol number and ICMP uses number 1. This control is enabled only if *Standard protocol selection* has been previously set to *Numeric*.
DSCP selection	Enables selection by the 6-bit DSCP field in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source address selection* field.
DSCP	This setting contains a 6-bit word in decimal format that constitutes the DSCP to be matched in the incoming traffic.

4.2.4. IPv6 Selection

Version 6 of the IP protocol is increasingly important in current network deployments. IPv6 is based on a packet structure different to IPv4. The main difference being the replacement of 32-bit IPv4 addresses by a larger set of 128-bit IPv6 addresses. The new address space is designed to solve the IPv4 address scarcity problem.

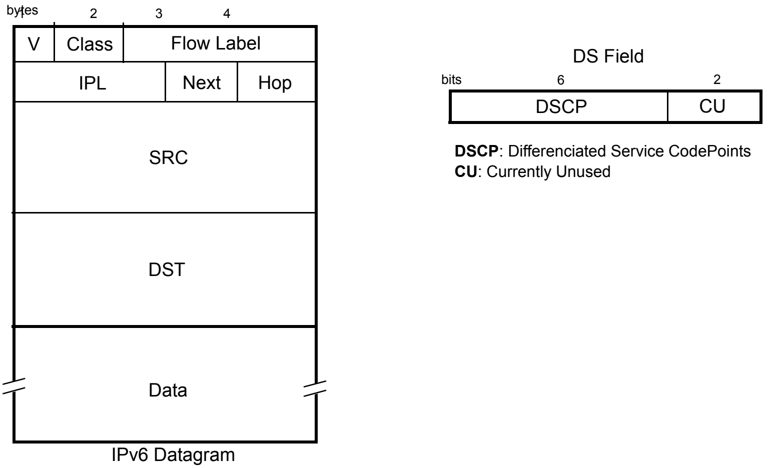


Figura 5: IPv6 datagram structure.

Net.Shark / Net.Hunter supports filtering based on various IPv6 packet fields including addresses, CoS marks, flow labels and higher layer protocol identifiers.

Table 4.1: IPv6 Selection

Metric	Description
Source IPv6 address match	Enables selection by source IPv6 address in the current filter. The value configured here is used to determine which packets are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: The source IP address is ignored and not taken into account for the purpose of the filter. All packets are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal to*: All packets matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Source IPv6 address* and *Source IPv6 address mask* objects.
Source IPv6 address	This is a 128-bit IPv6 address in the A:B:C:D:E:F:G:H format, where A, B, C, D, E, F, G and H are hexadecimal numbers between *0000* and *ffff*. Source addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching packets to this filter are configured by means the *Source IPv6 address mask* field.
Source IPv6 address mask	This is the mask for the source IPv6 address filter selection rule. Before comparing the *Source IPv6 Address Match* field with the actual IPv6 addresses, bit wise *AND* operations are carried out between the value configured here and the source address so that only the values surviving the *AND* are taken into account for matching addresses.
Destination IPv6 address match	Enables selection by destination IPv6 address in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source IPv6 address match* setting.
Destination IPv6 address	This is a 128-bit IPv6 address in the A:B:C:D:E:F:G:H format, where A, B, C, D, E, F, G and H are hexadecimal numbers between *0000* and *ffff*. Destination addresses matching some or all bits of the address configured here will be allowed to pass through the filter. Bits used for matching packets to this filter are configured by means the *Destination IPv6 address mask* field.

Table 4.1: IPv6 Selection

Metric	Description
Destination IPv6 address mask	This is the mask for the destination IPv6 address filter selection rule. Before comparing the *Destination IPv6 Address Match* field with the actual IPv6 addresses, bit wise *AND* operations are carried out between the value configured here and the source address so that only the values surviving the *AND* are taken into account for matching addresses.
Next Header match	Enables selection by the 8-bit IPv6 next header field in the current filter. The value configured here is used to determine which packets are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source IPv6 address match* setting.
Next Header	Configures the protocol identifier to be filtered when *Next Header matching* is enabled. The available configuration values are the following: • *Numeric*: Use this control if the traffic to be matched is different of UDP, TCP and ICMP and it has an specific protocol identifier assigned by the IANA. • *UDP*: Matches traffic with a *User Datagram Protocol (UDP)* envelope. Traffic commonly transported over UDP includes IP voice, IP video and DNS. • *TCP*: Matches traffic carried over the Transfer Control Protocol (TCP). Most data applications (web, file transfer, e-mail...) ere normally based on TCP transport. • *ICMP*: Matches *Internet Control Message Protocol* packets. IP operation and maintenance traffic like ping use ICMP.
Next Header number	This object contains an 8-bit word that constitutes the next header identifier to be matched in the incoming traffic. For example, configuring this object to 17 matches UDP traffic, TCP uses 6 as the protocol number and ICMPv6 uses number 58. The *Next Header number* configuration field is enabled only if *Next Header* is configured to *Numeric*.

Table 4.1: IPv6 Selection

Metric	Description
Flow Label match	Enables selection by the 20-bit IPv6 flow label field in the current filter. The value configured here is used to determine which packets are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source IPv6 address match* setting.
Flow label	The *Flow Label* IPv6 field contains a 20-bit word that identifies an unidirectional data flow. These labels remain at disposal of intermediate routers for stateful and stateless processing at flow level. For example, the flow label could be used to prevent load balancing on a particular traffic flow.
DSCP match	Enables selection by the 6-bit differentiated services code point (DSCP) field in the current filter. The value configured here is used to determine which packets are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source IPv6 address match* field.
DSCP	This object contains a 6-bit word in decimal format that constitutes the DSCP to be matched in the incoming traffic.

4.2.5. TCP Selection

The IP protocol does not offer reliable end-to-end communications. This capability is one of the attributions of the transport layer that operates above the IP layer. The *Transfer Control Protocol* (TCP) offers error recovery mechanisms that enable reliable data transmission over IP.

TCP also offers a communication channel for IP applications. Applications speak through special 16-bit identifiers called ports in the same way that hosts use IP addresses. Net.Shark / Net.Hunter filtering capabilities include matching of TCP source and destination ports in each of the filtering blocks.

Table 4.2: TCP Selection

Parameter	Description
Source port match	Enables selection by source TCP port in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: The source port is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *In range*: All frames with a destination port in an specified range are allowed to pass through the filter. The port range is specified with the help of the *Minimum source port* and *Maximum source port* fields.
Minimum source port	This is the minimum 16-bit TCP source port allowed to pass through the Source port selection filter. The port is configured and displayed in decimal format.
Maximum source port	This is the maximum 16-bit TCP source port allowed to pass through the Source port selection filter. The port is configured and displayed in decimal format.
Destination port match	Enables selection by the 2-byte TCP protocol field in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are the same that for the *Source address selection* field.
Minimum destination port	This is the minimum 16-bit TCP destination port allowed to pass through the Destination port selection filter. The port is configured and displayed in decimal format.
Maximum destination port	This is the maximum 16-bit TCP destination port allowed to pass through the Destination port selection filter. The port is configured and displayed in decimal format.

4.2.6. UDP Selection

Some applications do not require reliable transmission at the transport layer either because they implement their own error control mechanisms or because the mechanisms used by TCP are too slow for them. These applications can use the light weight User Datagram Protocol (UDP). Like TCP, UDP provides communications through ports to applications but it doesn't have any error recovery capability.

Net.Shark / Net.Hunter features related with UDP are equivalent to the TCP capabilities, including matching of the source and destination UDP ports.

4.2.7. Fixed Offset Selection

Fixed offset selection is part of a family of protocol-independent filtering criteria classified under the designation of Generic Selection. Fixed offset selection is the matching mode to be used when the Ethernet frames carry uncommon protocols or if inspection beyond the UDP and TCP transport protocols is required. This selection mode defines an offset and a mask. Frames matching the specified mask in the configured offset are selected.

Table 4.3: Fixed offset Selection

Parameter	Description
Payload selection	Defines the payload type and the reference point within the frame to start counting the filter offset. The reference can be the beginning of the MAC, IPv4, UDP or TCP payload depending on the chosen value. It is also possible to set the reference to the beginning of the Ethernet frame (first byte immediately after the SDF) by configuring *Whole frame* in this field. The frame start field is shared by all the port A or port B filters. If the *Frame start* field is modified for one specific filter, the remaining filters will be set to the same value.
Filter match	Enables fixed offset selection in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: Fixed offset matching is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal with mask*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Frame start*, *Offset (bytes)* and *Match code* fields.

Table 4.3: Fixed offset Selection

Parameter	Description
Offset (bytes)	This field defines the offset expressed in bytes from the reference point defined with the *Frame start* control. The value 0 corresponds with the first byte of the MAC, IPv4, UDP or TCP payload (or the first frame byte, if *Frame start* is set to *Start*). If due to the limited frame size, some or all the byte positions defined by the *Offset (bytes)* field, do not exist in the corresponding payload, the equipment will consider that the frame does not match the filtering criteria. The offset field is shared by all the port A or port B filters. If the *Offset (bytes)* field is modified for one specific filter, the remaining filters will be set to the same value.
Match code	16-bit code expressed with four hexadecimal digits used to match frames.in the current filter.
Mask	This is a mask for the generic filter match code. Before comparing the *Match code* with the selected bytes in the Ethernet frame, bit wise AND operations are carried out between the value configured here and the *Match code* field so that only the values surviving the AND are taken into account for matching the filter.

4.2.8. Length Selection

The length selection is a protocol independent filtering criterium that matches frames by their length. It is useful to drop frames known to have an specific length or an specific length range.

Table 4.4: Length Selection

Parameter	Description
Payload selection	Defines the protocol payload used to measure frame lengths. The payload could be the Ethernet MAC, IPv4, UDP or TCP payload depending on the chosen value. It is also possible to use the whole MAC frame for measure lengths by configuring *Whole frame* in this field. The frame start field is shared by all the port A or port B filters. If the *Frame start* field is modified for one specific filter, the remaining filters will be set to the same value.

Table 4.4: Length Selection

Parameter	Description
Filter mode	Enables length selection in the current filtering block: The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: Length matching is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection criteria. • *In Range*: All frames with a size between a minimum and a maximum length are allowed to pass through the filter. The length range is specified with the help of the *Minimum length* and *Maximum length* fields. • *Out of range*: All frames with a size not between a configurable length range are allowed to pass through the filter. The length range is specified with the help of the *Minimum length* and *Maximum length* fields.
Minimum length	Minimum frame size in the length range configured for the *In Range* and *Out of range* filter modes.
Maximum length	Maximum frame size in the length range configured for the *In Range* and *Out of range* filter modes.

4.2.9. Fixed Pattern selection

This generic filtering mode is appropriate when it is wanted to search for an specific text pattern in the traffic stream but it is not known the exact location of the string within the frame.

Table 4.5: Pattern Selection

Parameter	Description
Payload select	Defines the payload type and the point within the frame to start looking for matches with the search string. The allowed points are the beginning of the MAC, IPv4, UDP or TCP payload depending on the chosen value. It is also possible to start searching from the beginning of the Ethernet frame (first byte immediately after the SDF) by configuring *Start* in this field. The frame start field is shared by all the port A or port B filters. If the *Frame start* field is modified for one specific filter, the remaining filters will be set to the same value.

Table 4.5: Pattern Selection

Parameter	Description
Filter mode	Enables fixed pattern selection in the current filter. The value configured here is used to determine which frames are allowed to pass through the filter and which ones are blocked. The available configuration values for this field are: • *Ignore*: Fixed pattern matching is ignored and not taken into account for the purpose of the filter. All frames are allowed to pass through the filter when *Ignore* is configured if they are not blocked by other selection rule. • *Equal with mask*: All frames matching a user configurable pattern are allowed to pass through the filter. The matching pattern is configured with the help of the *Pattern* field.
Pattern	This configuration field contains the text pattern to be matched in the *Fixed pattern* search. The pattern is specified as an ASCII encoded text string. In this string, some characters and character combinations have special meanings: • “?”: This code matches any ASCII character, for example, the “AB?D” matches “ABCD”, “ABzD” or “AB%D”. To match the “?”, the “?” symbol is escaped with “\”. For example, “AB\?D” matches “AB?D” but not “ABzD” and not “AB\?D” and not “AB\zD”. • “\”. This is the escape character. The combinations “\?”, and “\\” restore the “?” and “\” their original meanings. Any alphabetic character can be escaped if it is wanted to ignore its case. For example, “A\bCD” matches both “ABCD” and “AbCD”. “\A\B\c\D” matches “ABCD”, “ABCd”, “ABcD”, “abcd” and any possible combination of the upper case and lower case “A”, “B”, “C” and “D” letters.

Net.Shark has 16 fixed pattern filters per port, one for each filtering block. Net.Hunter has only one fixed pattern filter per port (*Filter 1* block of each port). However, the matching pattern is allowed to be longer in Net.Hunter (128 bytes) than in Net.Shark (32 bytes).

4.3. Getting Statistics about Filters

Once the filter has been configured, it is usually desirable to know how many matching frames have been found for each filter. To get statistics about filtered frames follow this procedure:

1. From the *Home* panel, go to *Results*,
   The test port results panel is displayed.
2. Select *Filter statistics*.
   Counts of all frames matching the selection rules for all filters and line port A and B are displayed.

Chapter 5
Capturing Traffic

Capturing network traffic is the main Net.Hunter application and there is also a simplified traffic capture for Net.Shark. Traffic may potentially overload Net.Shark under certain circumstances. Some captures may contain many millions of frames an several tens of Gigabytes of disk space. Controlling the capture and data management are critical topics for anybody working with Net.Hunter and Net.Shark.

Due to the increased complexity of traffic captures generated by Net.Hunter, this chapter is mostly dedicated to this equipment. Differences with Net.Shark are highlighted when necessary.

5.1. Configuring the Capture

Capturing with Net.Shark / Net.Hunter is pretty much the same than mirroring traffic to a mirror port. The equipment is connected in the same way (line ports) and filter configuration is the same. It follows a high level description of capture configuration in Net.Shark / Net.Hunter:

1. From the *Home* panel, go to *Setup*,
   The test port settings panel is displayed.
2. Configure *Mode* either to *Tap & Filter* or *Filter* depending on how the tester is connected to the network.
3. Configure *Mirror mode* to *Store*.
4. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
5. Go to *Capture files*.
6. Configure *Action when disk* full to Stop if you want the capture to stop in case the storage space is finished or Wrap around if you want the capture to cycle when no space is left in the internal storage disk.
   Note: The *Action when disk full* control is not available in Net.Shark.
   Note: A *wrap around* capture mode never deletes any data stored in captures previously existing in the disk but older frames within the current capture may be

deleted if there is no other space left on the storage device.
Note: Avoid using *Wrap around* on small regions of the internal storage disk or device performance and life may be affected.

7. Configure at least one filter in line port A or line port B (or both) (See section 4.2).
8. Start the capture by pressing the RUN button.
9. Use the *Filter statistics* and *Capture status* panel to display information about the ongoing capture (See section 5.2)
10. Stop the capture by pressing the RUN button a second time.
11. Download the capture by removing the SD card from the unit (*Net.Shark* only) or using the web interface (See section 5.3).

Despite the operational similarities between the Net.Shark software-based captures and the hardware-assisted Net.Hunter captures, they are both very different from the point of view of performance in terms of available storage space and throughput.

Net.Hunter provides 60 GB or 120 GB of internal storage space. On the other hand, the Net.Shark storage capacity depends on the particular SD card used to save capture files. Capture throughput is limited to a few Mb/s in the case of Net.Shark but it is large enough to enable wirespeed data captures (1 Gb/s + 1 Gb/s) in most situations.

5.2. Controlling the Capture

You can always get real-time information about how many frames have been captured from the Filter statistics panel. The Filter statistics is useful to know which port and which filtering blocks are matching the traffic you receive from the network. Usage of the Filter statistics panel is explained in the chapter specifically devoted to filtering data (See section 4.3)

The second tool to be used to get information about captures is the Capture status panel. This panel displays miscellaneous details about the ongoing capture. Some results form this panel are relevant for drop actions as well. To display the Capture status panel follow this procedure:

1. From the *Home* panel, go to *Results*,
   The general results and statistics menu is displayed.
2. Enter in *Capture status*.

3. Check the *Dumping*, *Remaining capacity, Suppressed frames form port A* (drop actions only), *Suppressed frames from port B* (drop actions only) and *Suppressed frames* results.

Table 5.1: Capture Status

Metric	Description
Capturing	This field shows whether the equipment is copying frames to the internal high speed storage device or attached SD card. This is usually the case when the equipment is configured for storing and there is an ongoing capture (RUN button). However, the *Capturing* indicator may be set to “No” in front of several capture error conditions. For example, if there is no space left in the Internal storage / SD card for newly captured frames.
Remaining capacity	Displays the capacity available in the attached SD card for new captures in Gigabyte (GB), Megabyte (MB), Kilobyte (kB) or byte (B).
Current capture size	Shows the current capture size expressed in Gigabyte (GB), Megabyte (MB), Kilobyte (kB) or byte (B). For Net.Hunter, the file size is approximate. The file size is increased in steps of 16 MB and the minimum size is 32 MB. If the actual file size is smaller that 32 MB this field will display 32 MB anyway.
Suppressed frames from port A	Accounts for the number of filtered line port A frames lost due to insufficient mirror / capture channel capacity. Frames may be lost if the mirror port bit rate is smaller than the line port capacity. Frames are more likely to be lost when aggregation is used to add the traffic from both transmission directions. No frame loss events should happen when no aggregation is configured and the mirror ports are operating at their maximum speed. Note: For Net.Hunter, this result applies both to captures and drop actions. In Net.Shark, this result applies to copies to mirror ports. The relevant Net.Shark suppressed frame counter for captures is the *Suppressed frames* result.
Suppressed frames from port B	This field is equivalent to the *Suppressed frames from port A* counter but it accounts for frames received from line port B.

Table 5.1: Capture Status

Metric	Description
Suppressed frames	This field displays the number of frames which cannot be processed by the capture software because the Net.Shark CPU is busy in other tasks. Depending on the operating conditions, Net.Shark may lose frames even if the capture rate is smaller than the maximum capture throughput for this equipment (3.2 Mb/s). Suppressed frames are lost and they will be missing in the capture file. Note: The amount of suppressed frames can be estimated only when the capture has finished. Note: This result is always disabled in Net.Hunter. The *Suppressed frames from port A* and *Suppressed frames from port B* are used for the same purpose in this equipment

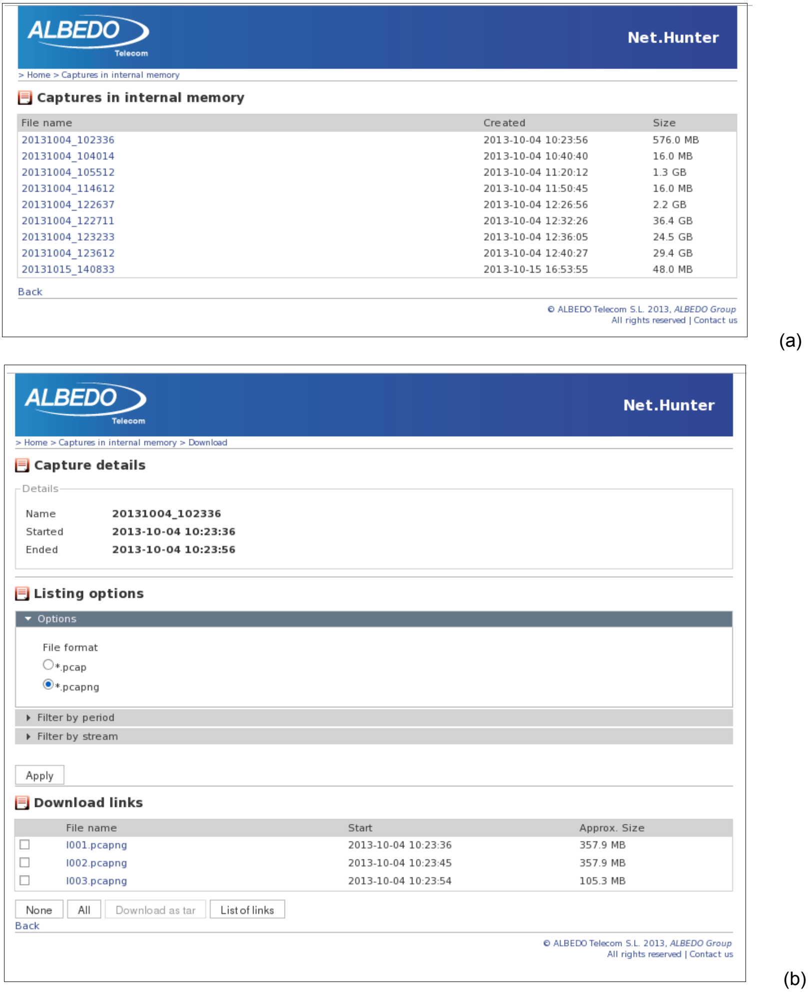
5.3. Capture Management

Capture exporting with Net.Hunter is slightly different to capture downloading from Net.Shark. This section is devoted to Net.Hunter capture file exporting. Details about file downloading with Net.Shark can be found on the sections devoted to generic file management (See section 6.1.6).

To export captures from net Hunter you need to use the web interface. To use the web interface you need to connect the platform network connector to the management network and configure the management Ethernet interface (See section 6.3.1). Once you have done this, follow this procedure:

1. Open a browser in a computer with network connection.
2. Type the IP address you have assigned to the tester in the browser destination URL.
   The web interface home panel is displayed in the Internet browser.
3. Select the *Internal memory* link in the *Capture files* menu.
   A list with all the captures currently available for download is displayed.
   Note: The list is not available if there is an ongoing capture. Stop the current capture or wait for the capture to finish before displaying previous captures.
   Note: The size assigned to the captures represents the raw size of the capture before being exported to any external media. The raw size is always a multiple of 16 MB. The raw size matches the storage space required to keep the capture data in the SSD but, once it has been exported, the real size of the corresponding PCAP / PCAPNG files could be smaller or larger than that size.
4. Choose the capture that contains the traffic you want to export.
   A list with all PCAPNG files available for download is displayed. Each file is identi-

fied with a unique file name. The exact time and date the file was generated and the approximative file size is shown for each PCAPNG file.
Note: The exported file size is always smaller than the value indicated in the screen. The size indication is only a worst case estimation of the PCAPNG size after being exported.



> Home > Captures in internal memory

Captures in internal memory

File name	Created	Size
20131004_102336	2013-10-04 10:23:56	576.0 MB
20131004_104014	2013-10-04 10:40:40	16.0 MB
20131004_105512	2013-10-04 11:20:12	1.3 GB
20131004_114612	2013-10-04 11:50:45	16.0 MB
20131004_122637	2013-10-04 12:26:56	2.2 GB
20131004_122711	2013-10-04 12:32:26	36.4 GB
20131004_123233	2013-10-04 12:36:05	24.5 GB
20131004_123612	2013-10-04 12:40:27	29.4 GB
20131015_140833	2013-10-15 16:53:55	48.0 MB

Back

© ALBEDO Telecom S.L. 2013, *ALBEDO Group*
All rights reserved | Contact us

(a)

> Home > Captures in internal memory > Download

Capture details

Details

Name	**20131004_102336**
Started	**2013-10-04 10:23:36**
Ended	**2013-10-04 10:23:56**

Listing options

▾ Options

File format
○ *.pcap
● *.pcapng

▸ Filter by period
▸ Filter by stream

Apply

Download links

	File name	Start	Approx. Size
□	l001.pcapng	2013-10-04 10:23:36	357.9 MB
□	l002.pcapng	2013-10-04 10:23:45	357.9 MB
□	l003.pcapng	2013-10-04 10:23:54	105.3 MB

None | All | Download as tar | List of links

Back

© ALBEDO Telecom S.L. 2013, *ALBEDO Group*
All rights reserved | Contact us

(b)

Figure 5.1: Net.Hunter web interface: (a) Capture list (b) Capture download panel with export filter selection.

5. Select the capture file format (**.pcap* or **.pcapng*) with the help of the *Options* menu and, if necessary, press *Apply* to confirm your selection.
6. Optionally, configure one or several export filters (See section 5.3.1).
7. Download some or all the PCAP / PCAPNG files by clicking at the corresponding hyper links or select the files with the help of the check boxes and download all them at the same time as a TAR package file.
   Note: Simultaneous downloading of several capture files is not supported. Please, wait for a file to be downloaded before starting with a new one.
   Note: It is possible to generate a download list for a download manager by clicking at the files to be downloaded and pressing list of links.
8. Open the PCAP / PCAPNG files using WireShark or any other protocol analysis software compatible with the file format.

Table 5.2: Capture file download option

Function	Description
None	Deselects all currently downloaded PCAP / PCAPNG files.
All	Selects all PCAP / PCAPNG files for downloading through the *Download as tar* or *List of links* functions.
Download as tar	Generates a TAR file with all the currently selected PCAP / PCAPNG files that is available for downloading as a single archive.
List of links	Generates a list of HTML links you can use as the input for a download manager. The download manager can be programmed to download all selected PCAP / PCAPNG files sequentially without user intervention.

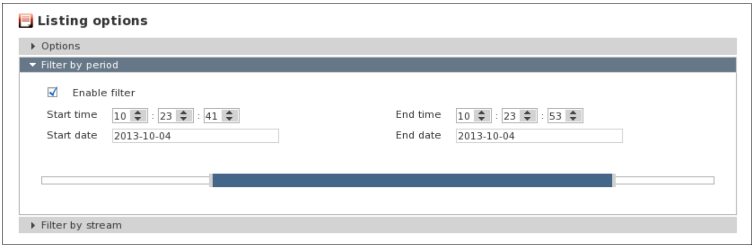
5.3.1. Using Export Filters

Net.Hunter can potentially include many millions of packets. Downloading the entire capture could be difficult and time consuming. For this reason, the equipment let's you select the data to be downloaded.

To use export filters, proceed as explained in previous sections but enable and configure at least one export filter just before exporting to PCAP or PCAPNG formats. The detailed procedure is as follows

1. Optionally, from the PCAP / PCAPNG, download panel in the web interface, configure the range of data to be exported by enabling (*Enable filter* check box) *Filter by period* and selecting the start and end date / time of the data to be exported.
2. Optionally, configure the origin of data to be exported by enabling (*Enable filter* check box) *Filter by stream* and selecting the traffic flows to be exported.
   Note: When the frames are captured they are marked with flow identifier that

depends on the filter matching each frame. These marks can be used to choose which frames to export once the capture has finished.



Listing options

Options

Filter by period

☑ Enable filter

Start time 10 : 23 : 41

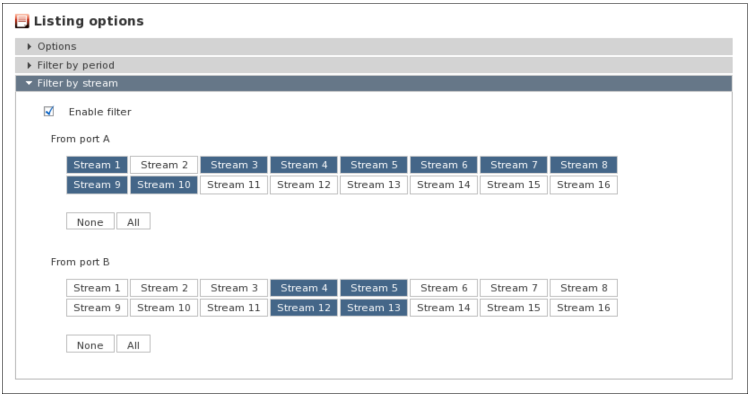
Start date 2013-10-04

End time 10 : 23 : 53

End date 2013-10-04

Filter by stream

(a)



Listing options

Options

Filter by period

Filter by stream

☑ Enable filter

From port A

[✓ Stream 1]	Stream 2	[✓ Stream 3]	[✓ Stream 4]	[✓ Stream 5]	[✓ Stream 6]	[✓ Stream 7]	[✓ Stream 8]
[✓ Stream 9]	[✓ Stream 10]	Stream 11	Stream 12	Stream 13	Stream 14	Stream 15	Stream 16

None All

From port B

Stream 1	Stream 2	Stream 3	[✓ Stream 4]	[✓ Stream 5]	Stream 6	Stream 7	Stream 8
Stream 9	Stream 10	Stream 11	[✓ Stream 12]	[✓ Stream 13]	Stream 14	Stream 15	Stream 16

None All

(b)

Figure 5.2: Net.Shark / Net.Hunter web interface: (a) *Filter by period* panel (b) *Filter by stream* panel.

3. Press Apply and wait until the PCAP / PCAPNG files are regenerated.
   Note: Depending on the capture size, the PCAP / PCAPNG file regeneration may take several minutes.
   Note: The unit is not available so start a new capture when the PCAP / PCAPNG files are being generated. Please, wait to the entire process to finish before starting a new capture.
   Note: When you configure and apply one of the *Filter by period* or *Filter by stream* filter, the *Net.Hunter* hardware computes the PCAP / PCAPNG file size exactly and not only approximately.

Export filters are available only for Net.Hunter. This section does not apply to Net.Shark.

5.4. SSD Aging Indication

Net.Hunter storage is based on Solid-State Drive (SSD) technology which require special care to guarantee a good performance during all the equipment life. Specifically, the flash memory used by SSDs supports a limited number of write cycles which is usually about a few thousands. Very long or very intensive usage of the Net.Hunter SSDs would cause the equipment to lose part of the initial capture speed. Please, contact with the Albedo Telecom staff or with your Albedo Telecom sales representative to get more information about potential limitations of the Net.Hunter SSD based captures.

Table 5.3: SSD Status

Metric	Description
SSD firmware version	Firmware version that corresponds with the Solid State Drive currently installed in your unit. This firmware version is different to the Net.Hunter firmware version available from the *System* menu.
Power on hours	Accounts for the time the Net.Hunter internal drive has been on. The drive is on if the Net.Hunter unit is on and configured in *Store* mode (*Mode* setting from the *Setup* menu).
Wearout indicator	This number is an indication of the drive age. The more used is the SSD, the larger is the value of the wearout indicator.
Total writes on disk	Contains information about how much information has been written in the SSD since it was installed in the Net.Hunter test unit.

To check the aging information about the SSD currently installed in your unit follow these steps:

1. From the *Home* panel, go to *Results*,
   The general results and statistics menu is displayed.
2. Enter in *SSD Status*.
3. Check the *SSD firmware version*, *Power on hours*, *Wearout indicator* and *Total writes on disk* fields.

Chapter 6
Test Management

This chapter describes all those features available in your test unit that are not directly related with configuring your tester or reading measurement results but they are important for proper test management. Specifically, configuration management, capture management and test platform settings are covered in the following sections.

6.1. File Management

Net.Shark / Net.Hunter stores configurations and captures in files. These files can be deleted, renamed and, depending on the file type, exported to an external USB memory or SD card. Configurations can be shared between different Net.Shark / Net.Hunter units by means compatible storage devices. Captures are stored in a suitable format (PCAP or PCAPNG) to be loaded in a protocol analyser like WireShark.

6.1.1. Saving Configurations

To store the current configuration follow these steps:

1. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
2. Select *Configuration files* to go to the configuration file settings.
3. Select the location to save the configuration: *Internal memory*, or *External devices*.
   Note: If you select E*xternal devices*, you will be asked to choose the specific storage device (USB device or SD card).
   Note: If there is no external device connected to the Net.Shark / Net.Hunter unit, a *No devices present* popup panel is displayed.
4. Press the *Save* (F2) contextual button.
5. Enter a file name for the configuration file that is going to be saved and confirm with the *Done* (F4) contextual button.

6.1.2. Renaming Files

Both configuration and capture files can be renamed after they are created. To rename files follow these sequence:

1. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
2. Select *Configuration files* or *Capture files*.
3. For configurations, select the location of the file you want to rename: *Internal memory*, or *External devices*.
   Note: If you select E*xternal devices*, you will be asked to choose the specific storage device (USB device or SD card).
   Note: If there is no external device connected to the Net.Shark / Net.Hunter, a *No devices present* popup panel is displayed.
4. Select the file you want to rename with the help of the cursors and the ENTER button.
   Note: You can select several files in the list, but renaming of many files at the same time is not allowed.
5. Press the *Rename* contextual button.
6. Enter the new file name for the selected configuration or report file with the alphanumeric keyboard. Confirm with the *Done* (F4) contextual button.

6.1.3. Deleting Files

With the file manager you can delete files that are not needed anymore. To do that follow these steps:

1. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
2. Select *Configuration files* or *Capture files*.
3. For configuration, select the location of the file you want to delete: *Internal memory*, or *External devices*.
   Note: If you select E*xternal devices*, you will be asked to choose the specific storage device (USB device or SD card).
   Note: If there is no external device connected to the Net.Shark / Net.Hunter unit, a *No devices present* popup panel is displayed.
4. Select the file you want to delete with the help of the cursors and the ENTER button.
   Note: You can select several files in the list at the same time.
5. Press the *Delete* contextual button.
6. Enter the new file name for the selected configuration or report file with the alphanumeric keyboard. Confirm with the *Done* (F4) contextual button.

6.1.4. Exporting Configurations to External Devices

Configuration files can be exported to external devices like USB memories or SD cards. The procedure is as follows:

1. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
2. Select *Configuration files* or *Report files*.
3. Select *Internal memory*, to list the files currently stored in the Net.Shark / Net.Hunter unit.
4. Select the files you want to export with the help of the cursors and the ENTER button.
5. Press the *Export* contextual button.
   A popup menu to select the external device where the files will be exported is opened.
   Note: If there is no external device connected to Net.Shark / Net.Hunter, a *No devices present* popup panel is displayed.
6. Select an external device, confirm, and wait for the files to be copied.
7. Remove the USB storage device or SD card from the unit.

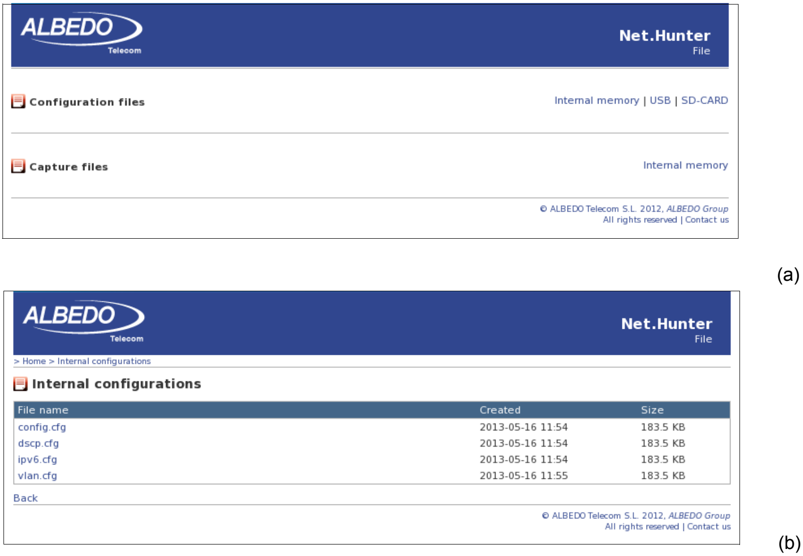
6.1.5. Importing Configurations

If you have a configuration file from a compatible tester you can import and load this file in your unit to reproduce similar measurements. This is the procedure you have to follow:

1. From the *Home* panel, go to *File*,
   The tester file manager base menu is displayed.
2. Select *Configuration files* to go to the configuration file settings.
3. Select *External devices* to list the files currently stored in the external device.
   A popup menu to select the source external device is opened.
   Note: If there is no external device connected to the Net.Shark / Net.Hunter unit, a *No devices present* popup panel is displayed.
4. Select the configuration files you want to import with the help of the cursors and the ENTER button.
5. Press the *Import* contextual button, confirm, and wait for the files to be copied from the internal memory.
6. Remove the USB storage device or SD card from the unit.

6.1.6. Using the Embedded Web Server

As an alternative of using a USB external storage device or an SD card for file management, Net.Shark / Net.Hunter has a web interface that can be used for the same purpose.



ALBEDO Telecom

Net.Hunter
File

Configuration files — Internal memory | USB | SD-CARD

Capture files — Internal memory

© ALBEDO Telecom S.L. 2012, *ALBEDO Group*
All rights reserved | Contact us

(a)

ALBEDO Telecom

Net.Hunter
File

> Home > Internal configurations

Internal configurations

File name	Created	Size
config.cfg	2013-05-16 11:54	183.5 KB
dscp.cfg	2013-05-16 11:54	183.5 KB
ipv6.cfg	2013-05-16 11:54	183.5 KB
vlan.cfg	2013-05-16 11:55	183.5 KB

Back

© ALBEDO Telecom S.L. 2012, *ALBEDO Group*
All rights reserved | Contact us

(b)

Figure 6.1: Net.Shark / Net.Hunter web interface: (a) Home panel (b) Configuration management panel.

The web interface can be used for downloading configurations and captures from a remote computer without using any accessory other than an standard network connection. Currently, the web interface does not support file uploading but for this purpose, the USB and SD interfaces are still available. Note than if you are using Net.Shark, you can export the PCAP capture files with the help of the SD card you use for storage but if you are capturing data with Net.Hunter, the only way to copy captures to an external location is through the web interface.

To use the web interface you need to connect the platform network connector to the management network and configure the management Ethernet interface (See section 6.3.1). Once you have done this, follow this procedure:

1. Open a browser in a computer with network connection.
2. Type the IP address you have assigned to the tester in the browser destination URL.
   The web interface home panel is displayed in the Internet browser.

3. Choose the files you want to display (*Configuration flies*, *Capture files* or any other if available) and the location of these files (*Internal memory*, *USB*, *SD-CARD*) and press to the correct hyper link.
   A list with the available files for the selected category is displayed in the web browser.
4. Select the file you want to download it to the local computer.
   The web browser displays a dialogue that requests your configuration to download the selected file. If you accept, the file will be downloaded.

This procedure has to be modified if it has to be applied to capture management in Net.Hunter. To know the details about how to download captures from the internal Net.Hunter storage device read the corresponding section in the chapter devoted to capture management (See section 5.3).

6.2. Programming Tests

Net.Shark / Net.Hunter is able to start and finish tests without direct user intervention. All automatic testing features are included within the *Autostart/stop* menu

Follow these steps to program an automatic measurement in the test unit.

1. From the *Home* panel, go to *Test*,
   The test configuration panel is displayed.
2. Select *Autostart/stop* to enter in the automatic test programming menu.
3. If you want the automatic test to start at a specific date and time set *Start mode* to *Auto* and enter the start date and time in *Start time.*
   Note: Manual start has precedence over autostart. That means that if a tester is started by pressing RUN but there is an automatic test programmed the manual test will start anyway.
4. If you want the automatic test to stop at a specific time after it has started set *Stop mode* to *Auto* and enter test duration with the help of the *Duration* and *User duration* controls*.*
   Note: Manual stop has precedence over autostop. That means that if a tester is stopped by pressing RUN but there is an automatic test programmed the manual test will stop anyway.

Table 6.1: System Settings Panel

Setting	Description
Start mode	Configures the start test mode. There are two different choices here: • *Manual*: The test starts when there is not an ongoing test and the RUN key is pressed. • *Auto*: The test starts at a configured date and time without the need of pressing any key.

Table 6.1: System Settings Panel

Setting	Description
Start time	Enter the start date and time for the next automatic measurement with the following format: *dd/MM/yyyy hh:mm:ss*. To configure Start time, you have to set *Start* mode to *Auto* before.
Stop mode	Configures the stop test mode. There are two possibilities: • *Manual*: The test finishes when there is an ongoing test and the RUN key is pressed. • *Auto*: The test finishes when a configurable test duration is reached. This mode does not require user intervention once the duration has been set and the measurement has started.
Duration	Sets the duration of the next measurement. The available test durations are: 15 minutes, 1 hour, 1 day, 7 days, 30 days or user configurable duration. Setting up *Duration* requires previous configuration of *Stop Mode* to *Auto*.
User duration	Sets the duration of the next measurement when *Stop mode* has been configured to *Auto* and *Duration* to *User*. The duration has to be entered in a *hh:mm:ss* format.
Last started on	Displays the date and time when the last measurement was started.
Last stopped on	Displays the date and time when the last measurement was stopped. If there is an ongoing test, the value of this field is empty.
Last power down on	Displays the date and time when the tester was powered down for last time.

6.3. Using the System Menu

The System menu includes platform wide settings organized in four different submenus:

- *General settings*: This menu includes controls to manage the way the user interface behaves and how the information is presented.
- *Network configuration*: Includes the IP configuration corresponding with the platform NIC.
- *System information*: This menu has the test unit model name and serial number and software, firmware and hardware versions.

- *Licensed options*: This is a menu that displays the software versions installed in the tester and enables their management.

Table 6.2: System Settings Panel

Setting	Description
Brightness (%)	Sets the screen brightness from 10% to 100%. Within the *Brightness* panel, the left and right cursors are used to set the correct value and a contextual key (*Done*) is used to confirm selection.
Keyclick	Enables or disables the keyclick. The keyclick is a sound that is played each time a key is pressed.
Language	Selects the user interface language. Menus, selection lists and results are presented in the language selected here. The languages currently available are English and Spanish.
Clock setup	Configures the system time and date. You can either type the correct date and time manually or let the equipment to retrieve the correct values from a *Network Time Protocol* (NTP) server.
Time display	Select the way the time is displayed in the graphical user interface. One of the following has to be selected: • *Elapsed*: Time from the beginning of the test is displayed with the following format *hh:mm:ss*. If there is not an ongoing test, then the duration of the last test is shown • *Absolute*: The current date and time is displayed with the following format: *dd/MM/yyyy hh:mm:ss*.
Screensaver	Sets or unsets the screensaver. The screensaver reduces power consumption and increases operation time under battery operation.
Screensaver delay	Configures the delay to switch the screensaver on. The backlight brightness is set to a low value once the time configured here has finished. The display backlight is switched off after twice the screensaver delay. The available configuration values for this item are: 10s, 30s, 1min, 2min, 5min, 10min, 20min.
Remote control	Enables or disables the Ethernet / IP remote control. The remote control is an optional feature that enables remote users to use the tester from a computer running VNC.
Remote control password	Configures a password for the remote control. Any alphanumeric string should be accepted. Use the same password in the remote VNC client to access to the tester user interface.

This section supplies a description of the *General settings* menu and *System information* menu. To learn how to configure and use the network interface or how to install licenses for new software options go to the sections specifically dedicated to these topics.

Table 6.3: System information panel

Setting	Description
Model Name	Shows the test unit model name: *Net.Shark* or *Net.Hunter*.
Serial number	Displays the test unit serial number. It is a 8 character alphanumeric string
Software release	Displays the current software release.
Hardware release	Displays the current hardware release.
Firmware release	Displays the current firmware release.
PM release	Displays the current power management release.

6.3.1. Using the Network

The platform network interface is currently user for three different purposes:

- The *Ethernet / IP remote control*. This feature enables any user to access to the equipment form a remote location, configure a test, run it and display the results.
- The *Web interface*: This is used to retrieve captures, configurations or any other file available in the tester internal memory or attached storage device.
- *Maintenance and factory configuration*: The ALBEDO Telecom staff use the Ethernet interface to configure or verify the equipment in the factory. This feature is not available to ordinary users.

Table 6.4: Network Configuration Panel

Setting	Description
Ethernet interface	Configuration menu for the platform network interface. This menu can be used to configure the interface IP address and mask either automatically (DHCP) or statically.
Wireless interface	Configuration menu for the platform wireless network interface. This menu is used to set the radio parameters for the interface such as the SSID and the network parameters like the IP address and mask. The wireless interface requires a compatible WiFi adapter for the USB port. This adapter is supplied by ALBEDO as an optional accessory.

Table 6.4: Network Configuration Panel

Setting	Description
Gateway address	IP address corresponding to the IP default gateway in four dotted format. There is only one default gateway for all the network interfaces (wired and wireless). By setting up this field, the user decides which management port is used by the system to reach remote networks. It is possible to configure the gateway address automatically if either the wired or the wireless interfaces are configured to get an IP profile through the DHCP protocol.
DNS address	DNS server address used by the platform management ports to resolve domain names. It is possible to configure the DNS address automatically if either the wired or the wireless interfaces are configured to get an IP profile through the DHCP protocol.

To configure and use the Ethernet platform interface follow these steps:

1. From the *Home* panel, go to *System*,
   The general system menu is displayed in the screen.
2. Select *Network configuration* to display the network configuration and management menu.
3. Go to the *Ethernet interface*.
4. Enable the platform network interface with the *Enable interface* control.
5. Enable DHCP with the *Use DHCP* control if you want to let DHCP to configure your IP settings automatically or disable it to configure an static IP profile.
6. If you are not using DHCP, enter correct values for the *Static IP address* and *Static network mask*.
7. Leave the *Ethernet interface* panel with the ESC key.
8. If you are not using DHCP, configure the *Gateway address* and *DNS address*.
9. Connect the platform Ethernet connector (platform panel, RJ-45 connector with the *Ethernet* label) to the management network.

10. Optionally, check from a remote computer that the equipment is responding to ping requests.

Table 6.5: Ethernet Interface Configuration

Setting	Description
Enable interface	Enables or disables the network interface. Note that the link led placed in the Ethernet platform connector is lit even if the interface is not enabled.
Use DHCP	Configures the mechanism used to set the interface IP address and mask (and also other system-wide settings like the gateway address and the DNS server). If *Use DHCP* is enabled, the IP profile is configured automatically using a DHCP server installed in the network. Otherwise, the user has to enter the IP address, mask, default gateway and DNS address by hand.
Static IP address	Static IP address assigned to the interface in a decimal four dotted format. This setting makes sense only if *Use DHCP* is not enabled.
Static network mask	Static network mask in a decimal four dotted format. This setting makes sense only if *Use DHCP* is not enabled.
Fixed gateway address	IP address corresponding to the IP default gateway in four dotted format. This setting makes sense only if *Use DHCP* is not enabled.
Fixed DNS address	DNS server address used to resolve domain names. This setting makes sense only if *Use DHCP* is not enabled
Leased IP address	Current DHCP-assigned IP address in a decimal four dotted format. This is a read-only field that cannot be directly configured by users This setting makes sense only if *Use DHCP* is enabled.
Leased network mask	Current DHCP-assigned network mask in a decimal four dotted format. This is a read-only field that cannot be directly configured by users This setting makes sense only if *Use DHCP* is enabled.
Leased gateway address	IP address corresponding to the IP default gateway in four dotted format assigned by a local DHCP server. This is a read-only field that cannot be edited by the user. This setting makes sense only if *Use DHCP* is enabled.

Table 6.5: Ethernet Interface Configuration

Setting	Description
Leased DNS address	DNS server address used to resolve domain names assigned by a local DHCP server. This is a read-only field that cannot be edited by the user.
Ethernet address	48-bit physical address of the NIC attached to the test unit. This address is assigned to the NIC when it is manufactured and it cannot be changed later.

6.3.2. Installing Software Options

New software for Net.Shark / Net.Hunter can be licensed after the unit as been purchased when new testing needs arise. To install new software options for your unit follow this procedure.

Table 6.6: Licensing

Setting	Description
Licensed options	Shows a list with all the software options currently available in your test unit.
License number	8-digit hexadecimal number provided by ALBEDO Telecom that identifies the software options to be added to your unit. Enter your license number in this field before adding the new software options to your test unit.
License key	8-digit hexadecimal number provided by ALBEDO Telecom that enables secure management of the software options installed in your test unit. Enter the license key in this field before adding the new software options to your test unit.
Activate	Set this field to Yes to add new software options to your tester. You have to enter the *License number* and the *License key* before adding new options.
Status	Displays the result of the software option activation operation performed by enabling the *Activate* field.

1. Contact with your local sales representative to purchase software options for your test units.
   You will receive one license number and one license key for each tester you want to upgrade.
2. From the *Home* panel, go to *System*,
   The system configuration panel is displayed.

3. Select *Licensing* to enter in the software upgrade menu.
4. Enter the number and key supplied by your ALBEDO Telecom representative in *License number* and *License key*.
5. Enable the new software options with the *Activate* control.
6. Check that the upgrade has been successful with the help of the *Status* control.

6.3.3. Using NTP for System Clock Synchronization

The system clock controls the date and file assigned to configuration and capture files and controls the autostart / stop function. When a new capture starts, the capture clock takes the time from the system clock and therefore the system clock also has influence in the time stamps assigned to captured frames.

Net.Shark / Net.Hunter users can manually set the system time and date by entering the correct values but they can also synchronize the clock with an external NTP server. The NTP server must be available through the platform Ethernet port. Using NTP in Net.Shark / Net.Hunter has the extra benefit that the time stamps assigned to captured frames tend to be of better quality than if the clock is manually configured.

Table 6.7: Time Source Configuration Options

Setting	Description
Date	This is used to configure the current date. The date is used for the *Autostart/stop* features and other purposes. The date has to be entered with the following format: *dd/MM/yyyy*. You can only modify the date if the current time source has been set to *Manual*.
Time	This is used to configure the current time. The time is used for the *Autostart/stop* features and other purposes. The time has to be entered with the following format: *hh:mm:ss*. You can only modify the time if the current time source has been set to *Manual*.
Time source	It is either *Manua*l or *NTP.* The meaning of each configuration option is as follows: • *Manual*: The user configures the *Date* and *Time* fields manually. System date and time is controlled by the internal clock. • *NTP*: An external *Network Time Protocol* (NTP) server controls the value of the *Date* and *Time* fields. The system is synchronized with the server each time the equipment is restarted or when a new capture is run.

Table 6.7: Time Source Configuration Options

Setting	Description
UTC offset (hours)	This is the time difference in hours between your local time zone and the *Universal Time Coordinated* (UTC) time zone This setting makes sense only if *Time Source* has ben configured to *NTP*.
UTC offset (min)	This is the value to be configured when the time offset between your local time zone and the UTC zone is not an integer value of hours. This setting makes sense only if *Time Source* has ben configured to *NTP*.
NTP server	IP address or domain name corresponding to the NTP server you want to use to synchronize your unit. If you want to use a domain name for the server you need to make sure you have configured a DNS server in your network settings (See section 6.3.1). This setting makes sense only if *Time Source* has ben configured to *NTP*.
NTP status	Displays the current status of the NTP server configured in the *NTP server* field. It is one of the following: • Server not available: The server configured in NTP server is not available. Make sure that your network interface is properly configured and that the server is accessible. • *Synchronized*: The equipment is correctly synchronized with the external server configured in *NTP server*.. • *Waiting*: The test unit is still waiting for a reply from the remote server. This is a not editable field. It is only active if *Time Source* has been configured to *NTP*.

To configure the system time and date in your Net.Shark / Net.Hunter unit follow these steps:

1. From the *Home* panel, go to *System*,
   The general system menu is displayed in the screen.
2. Select *General settings* to display the platform-wide configuration.
3. Go to *Clock setup*
4. Configure *Manual* or *NTP* time and date with the help of the *Time source* field.

5. If you have configured *Time source* to *Manual* in the previous step, configure the *Date* and *Time* field to your local time and date. If *Time source* has been configured to *NTP*, enter the *UTC offset (hours)*, *UTC offset (min), NTP server* and wait for the equipment to establish synchronization with the server.

6.4. Using the Remote Control

The remote control application constitutes a remote graphical user interface that reproduces pixel by pixel the tester screen in virtually any remote device supporting the VNC protocol. This includes not only computers but also smartphones or tablets. The only requirements for the controlling devices are:

- IP connectivity with the tester. Any IP connection including Ethernet, WiFi and 3G should work.
- They must have a VNC client installed. Currently, there are VNC clients for most OS in the market. Some of them are free.

The remote control is an optional feature for Net.Shark / Net.Hunter that is supplied by ALBEDO Telecom with an special license.

Before using the remote control you need to configure the platform Ethernet interface and connect the equipment to the management network (See section 6.3.1). Once this is done, follow this procedure to use the remote control:

1. From the *Home* panel, go to *System*,
   The system configuration panel is displayed.
2. Select *General settings* to display miscellaneous system-wide settings, including the ones referred to the remote control.
3. Enable the remote control with the help of *Remote control*.
4. Optionally, supply a password with *Remote control password*. The password you configure here will be requested in all incoming VNC connections.
5. In the controlling device, run the VNC client and enter the password you have configured in *Remote control password* if you are requested to do so.
6. Use the keyboard (navigation through the mouse is not available in the remote control) to browse the instrument panels, start measurements, insert events or any other action.

Table 6.8: Remote Control Keys

Key	Description
Up, Down, Left, Right	These keys are equivalent to the cursor keys in the tester local interface. They move the focus through the different fields available in the current panel and they also help with the navigation through different panels.
Home	It is equivalent to the HOME key. It displays the *Home* panel.

Table 6.8: Remote Control Keys

Key	Description
Esc	It is equivalent to the Esc key. It leaves the current panel and displays the previous one in the panel hierarchy.
Enter	It is equivalent to the ENTER key. It confirms settings.
Ctrl+L	It is equivalent to LEDS. It displays the *Leds* panel.
Ctrl+S	It is equivalent to SUM. it displays the *Summary* screen
Ctrl+R	It is equivalent to RUN. It starts / stops a measurement
Ctrl+E	It is equivalent to EVENT. It starts / stops event insertion
F1, F2, F3, F4	They are equivalent to the F1, F2, F3, F4 contextual keys. The purpose of these keys depend on the current screen.

Appendix A
Technical Specification

A.1. Ports and Interfaces

- RJ-45 port for electrical connection 10/100/1000BASE-T for mirror ports.
- Optical and electrical SFPs ports operating at up to 1 Gb/s for line ports.
- SFP interfaces support: 10BASE-T, 100BASE-TX, 1000BASE-T, 100BASE-FX, 1000BASE-SX, 1000BASE-LX, 1000BASE-ZX.

A.2. Operation Modes

- Tap & filter: Traffic is forwarded between line ports, traffic is selectively copied to the mirror ports, stored in an SD card (Net.Shark) or stored the internal high speed storage device (Net.Hunter).
- Filter: Traffic is filtered and forwarded to the corresponding mirror port, stored in an SD card (Net.Shark) or stored in high speed internal storage device (Net.Hunter).

A.3. Formats and Protocols

- Ethernet frame: IEEE 802.3, IEEE 802.1Q, IEEE 802.1ad (Net.Hunter only).
- IP packet: IPv4 (IETF RFC 791), IPv6 (IETF RFC 2460).
- Jumbo frames: up to 10 kB MTU (Maximum Transmission Unit).
- Throughput between measurement ports: 1 Gb/s or 1,500,000 frames/s in each direction.
- PoE (IEEE 802.3af) and PoE+ (IEEE 802.3at) pass-through

A.4. Auto-negotiation

- Auto-negotiation and forced bit rate modes supported by mirror and line ports.
- Negotiation of bit rate. Allow 10 Mb/s, allow 100 Mb/s, allow 1000 Mb/s.

A.5. Configuration

- Configurable MTU size from 1518 bytes to 1000 bytes.
- Enable / disable traffic aggregation of both transmission directions to a single mirror port.

A.6. Results

- Auto-negotiation results including current bit rate, duplex mode, Ethernet interface.
- SFP presence, interface, vendor, and part number.
- Separate traffic statistics for each port.
- Separate statistics for transmit and receive directions.
- Frame counts: Ethernet, and IEEE 802.1Q (VLAN), control frames.
- Frame counts: unicast, multicast and broadcast.
- Error analysis: FCS errors, undersized frames, oversized frames, fragments, jabbers.
- Frame size counts: 64, 65-127, 128-255, 256-511, 512-1023, and 1024-1518 bytes.
- Byte counts: Port A (Tx / Rx) and Port B (Tx / Rx).
- Traffic counters follow RFC 2819.

A.7. Filters

- Up to 16 fully configurable and independent filters for each test port.
- User-configurable filters defined by field contents on Ethernet, IP, UDP and TCP headers.

A.7.1. Generic Filters

- Agnostic filters defined by 16-bit masks and user defined offset.
- Pattern filter (one per port) to match alphanumeric words or expressions
- Length filters to match frames by their length

A.7.2. Ethernet Filters

- MAC address: source, destination.
- MAC address group: subset of addresses filtered by a mask.
- Ethertype field with selection mask.
- VID (Net.Shark) or C-VID and S-VID (Net.Hunter)
- VLAN priority or C-VLAN priority and S-VLAN priority.
- S-VLAN DEI.

A.7.3. IPv4 Filters

- Selection by IPv4 source or destination address (or both at the same time). It is possible to select address sets by using masks.
- Selection by protocol encapsulated in the IP packet (TCP, UDP, Telnet, FTP, etc.).
- Selection by DSCP value.

A.7.4. IPv6 Filters

- Selection by IPv6 source or destination address (or both at the same time). It is possible to select address sets by using masks.
- Selection by IPv6 flow label.
- Selection based on the next header field value.
- Selection by DSCP value.

A.7.5. TCP / UDP Filters

- Selection by *TCP / UDP port*. Either as a single value or a ranges

A.7.6. Statistics

- Frame counters for each configured filter.

A.8. Capture

- Traffic capture to SD card (Net.Shark).
- Wirespeed traffic capture to internal SSD with capacity of 60 or 120 GB (Net.Hunter).
- Capture format is PCAP or PCAP Next Generation (Net.Hunter only).
- Hardware time stamping of captured data. Timestamp error smaller than ±20 ns.
- Export filters: Based on date / time or previous capture filter settings (Net.Hunter only).
- Phase synchronization of capture timestamps through NTP.

A.9. User Interface

- Direct configuration and management in graphical mode using the keyboard and display of the instrument.
- Remote access for configuration and management in graphical mode from remote IP site thought the Ethernet interface of the control panel.

A.10.General

- Operation time with batteries: 3.5 hours (minimum, two battery packs).

- Configuration and report storage and export through attached USB port.
- 4.3’’ TFT colour screen (480 x 272 pixels).
- Dimensions: 223 mm x 144 mm x 65 mm.
- Weight: 1.0 kg (with rubber boot, one battery pack).