

Professional Certificate of Competency in Advanced TCP/IP-Based Industrial Networking (CAV)

**Please note: These exercises are part of Assessment 1.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Module number** | Modules 1 to 6 | | |
| **Assessment type** | Exercises 15b, 16 & 38 | | |
| **Remote Lab Types:** | 1. EIT PC with hardware, 2. EIT PC with Simulation Software, 3. Cloud PC with software, 4. Student/Home PC | | |
| **Version** | 3 | | |
| **Created by** | D. Reynders | **Date** | 28 October 2013 |
| **Last Reviewed by** | D. Reynders | **Date** | 7 July 2021 |

|  |  |
| --- | --- |
| **Exercise 15b:** | **IPv6 Packet** |
| **Remote Lab PC:** | **Labs 1 & 2** |
| **Protocol Analyser:** | **Wireshark** |
| **Frame Generator:** | **Ostinato** |
| **Remote Lab Type:** | **D** |

**Objective**

To analyse the structure of an IPv6 packet (frame).

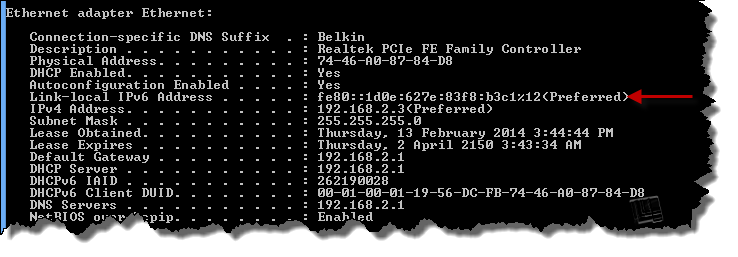
**Preparation**

It is assumed that you already have Wireshark installed or are accessing it via the remote labs. Obtain a trial version of Ostinato [here,](http://ostinato.org/pricing) scrolling down to ‘Ostinato Trlal’, end clicking on ‘Try’. You will receive download details via email. If using Gmail, check the ‘Promotions’ tab.

*Note that the following screenshots of ostinato may change. In this case, the instructions on the ostinato website will apply. The rest of this document will neverheless give you a good idea of what to do.*

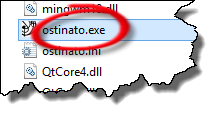
**Instructions**

First check the IPv6 address of the computer, e.g. by using the *ipconfig /all* command from the command prompt, or with IP Configuration Manager or something similar.

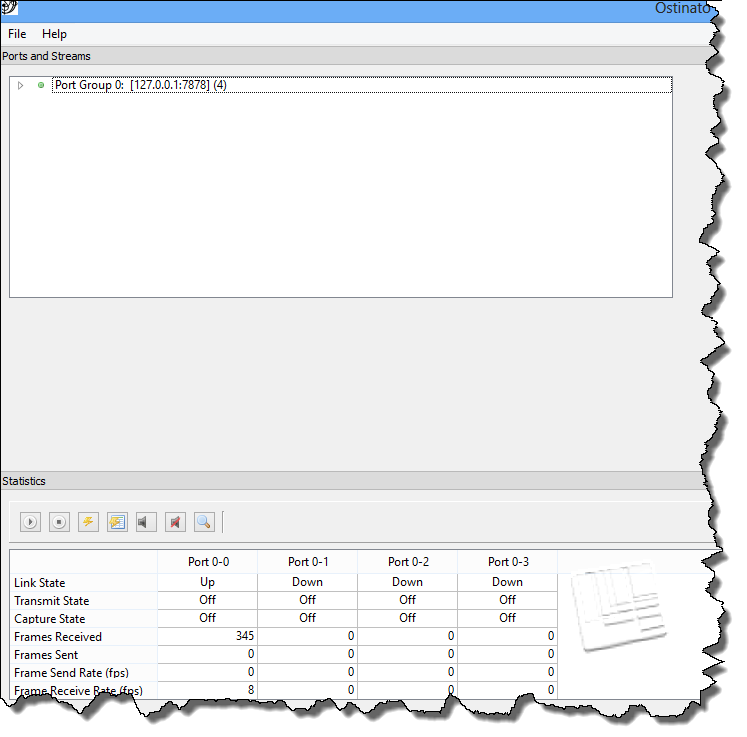


In our case the IPv6 address (on a Private network) is a Link-Local address. Apart from it being listed as such, it is also abvious because of the fe80 at the beginning of the address. Refer to the IP Reference Card (in your reading material folder on Moodle) for more detail in IPv6 classes.

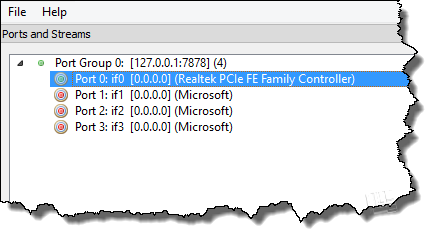
Now run Ostinato by clicking on the executable.



The program will open.



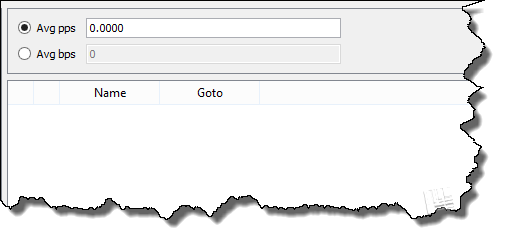
Double-click on Port Group 0 (top left). A drop down list with available interfaces will appear. We will select the Ethernet interface (see below).



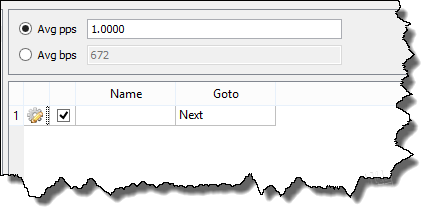
Another box will open in the top right-hand corner.

**NB Your antivirus program might object. Just tell it to allow the program to run.**

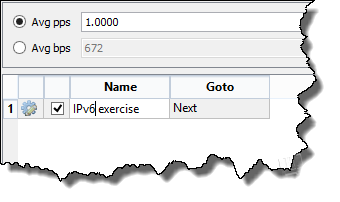




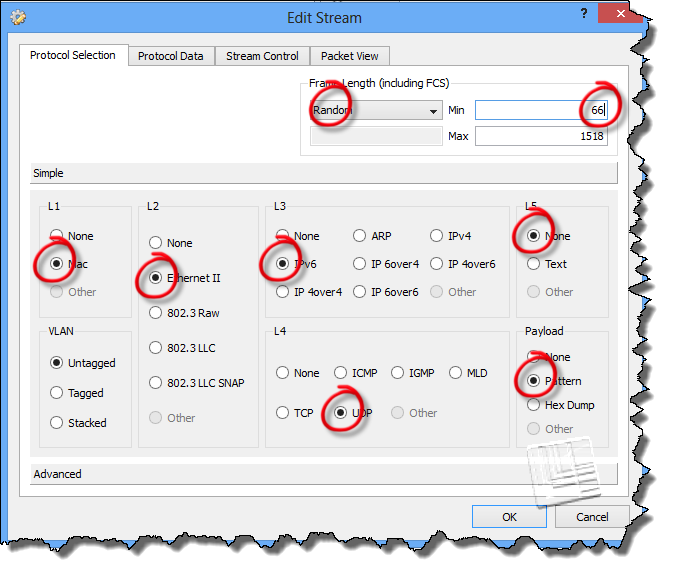
Right-click within this area. A menu will drop drown and we will select ‘New Stream’.



Double-click in the ‘Name’ field and give it any name you like.



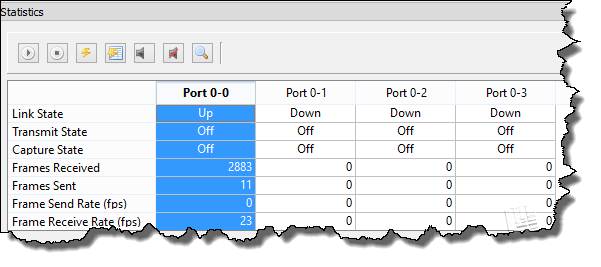
Now, once again, right-click in this area, and select ‘Edit Stream’. Use the radio buttons to configure a packet with a hex payload, using UDP over IPv6 over Ethernet.



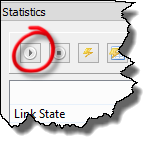
Click on the other tabs to check the packet composition and default parameters. Note that the default rate is 1 packet per second.

**NB DO NOT FORGET TO CLICK ‘APPLY’**

In the statstics window, select the same port (select the whole column by clicking on the column heading) for which you configured the stream (IMPORTANT).



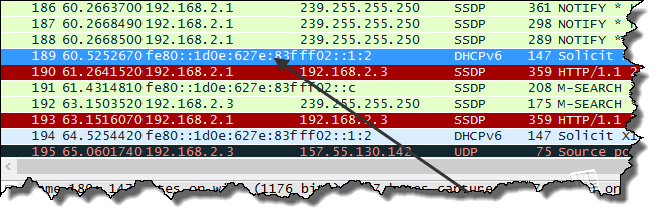
Now click on ‘Start Tx’ in the bottom left-hand corner.



It might be a good idea to close your browser in order to reduce the amount of captured traffic.

Run Wireshark and check the bottom of the list until you see the IPv6 packets appearing. Then stop.

Select any of the IPv6 frames. Note your own IPv6 address.



Note the following:

1. Version (6)
2. Traffic class
3. Flow label
4. Payload length
5. Next header (UDP)
6. Hop limit
7. Source IP address
8. Destination IP address

Compare this with the details of the IPv4 header as captured earlier in Exercise #15a.

|  |  |
| --- | --- |
| **Exercise 16a:** | **TCP/HTTP** |
| **Remote Lab PC:** | **N/A** |
| **Protocol Analyse:** | **Wireshark** |
| **Remote Lab Type:** | **D** |

**Objective**

To analyse the TCP connection (3-way handshake) between a browser and a webserver, and to observe the HTTP/TCP/IP/Ethernet protocol stack in operation.

**Preparation**

Download and install the latest version of Wireshark [here.](http://www.wireshark.org/download.html) Go for the default installation, and be sure to install WinPcap as well (win10pcap if you are running Windows10).

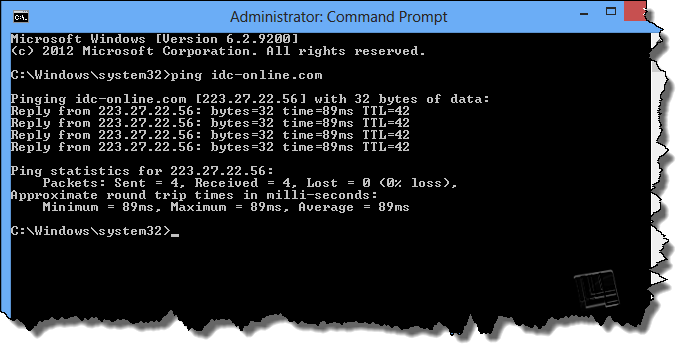
Also have some screen capture ready. SnagIt is a wel-known product, but Jing and ScreenHunter Free will also work. Be careful with free software, and do not allow the installation of toolbars etc.

**Instructions**

For this demonstration we will connect [here](http://www.idc-online.com). You may also be requested to connect to a specific (different) webserver, so read the instructions in your assignment carefully.

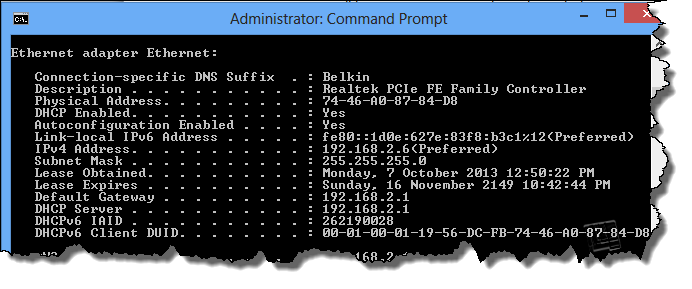
Ping the webserver you are going to use (e.g. *idc-online.com*) from the DOS (‘command’) prompt and note the IP address. In this case it is 223.27.22.56. You will have to snag your result, as shown below, as evidence of your destination IP address.

**Snag Your Result**

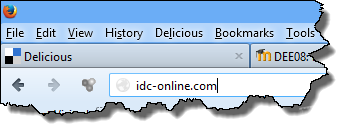


From the same DOS prompt, type ipconfig /all and note your own IP address. In our case this is 192.168.2.6 (Default Gateway = ADSL router = 192.168.2.1). You need this as evidence of your source IP address.

**SNAG YOUR RESULT**



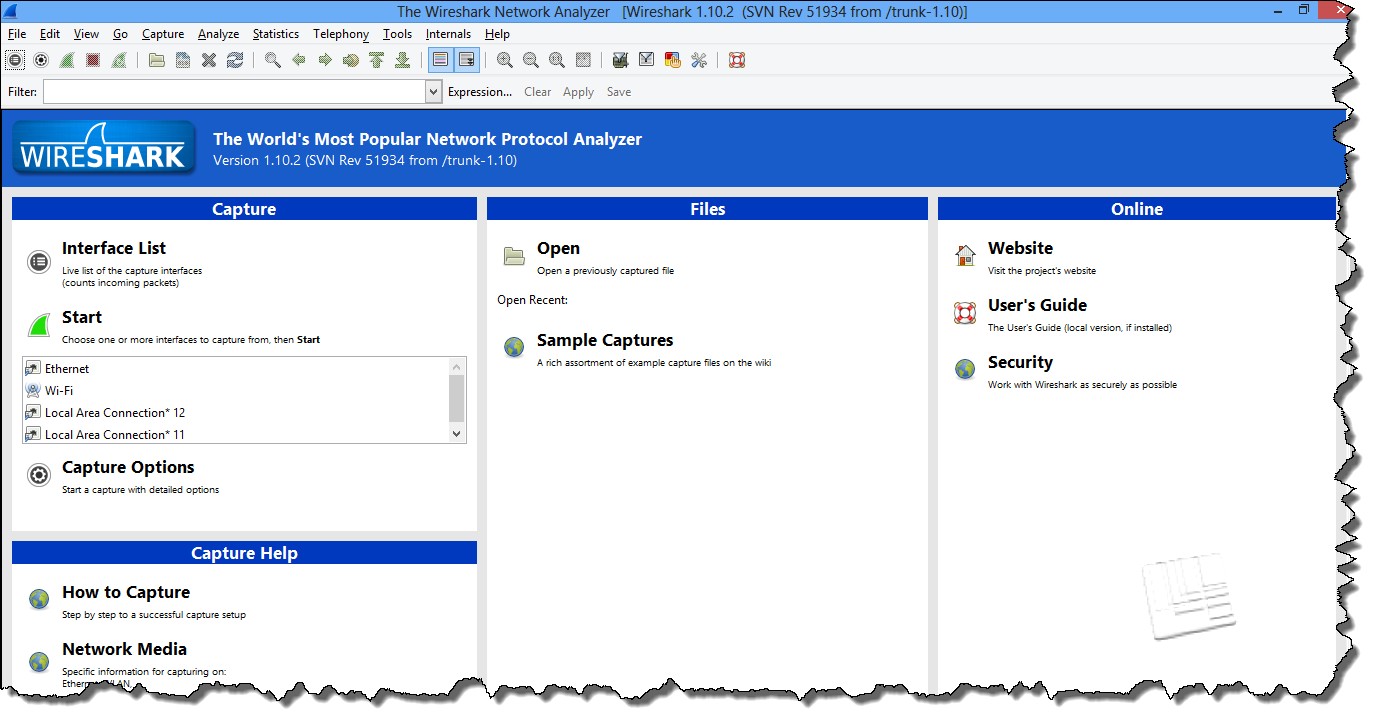
Now open your browser and type *idc-online.com* in the address bar. However, DO NOT HIT ‘RETURN’ YET, as you are not ready to capture the traffic yet.



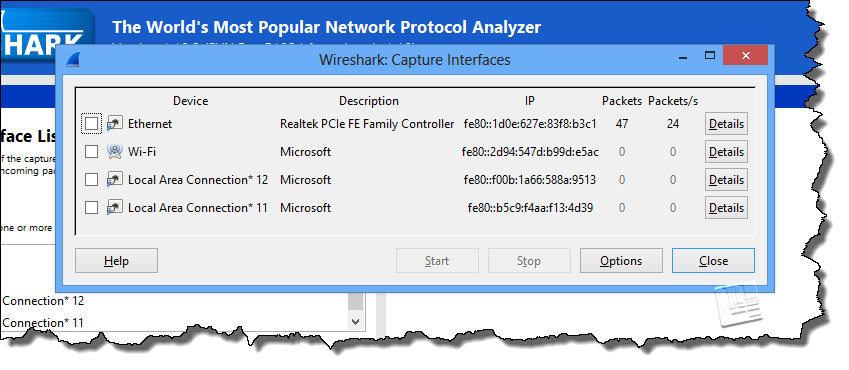
Get Wireshark going by clicking on the shark fin.



Wireshark will open up and look like this:



Click *Capture->Interfaces* and tick the box against the row that is showing activity.

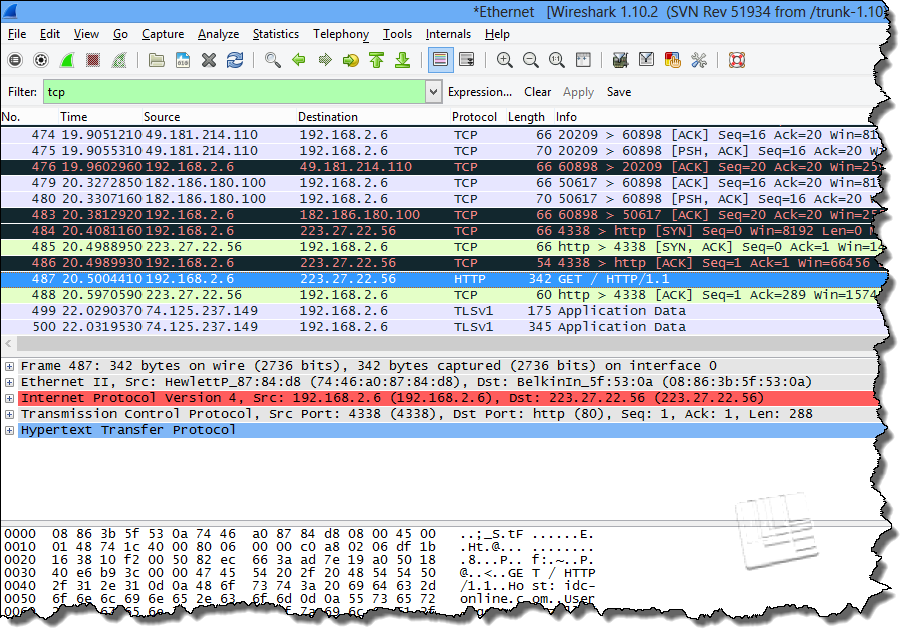


Now you have to perform the following steps in quick succession:

1. Click ‘start’ to commence the capturing
2. Hit ‘enter’ on the browser so that the *idc-online* website will open up
3. Hit the rectangular red ‘stop’ button (top, left) on Wireshark after a few seconds.

You do not want to end up with a proliferation of packets, but you want to start capturing *before* the connection is made.

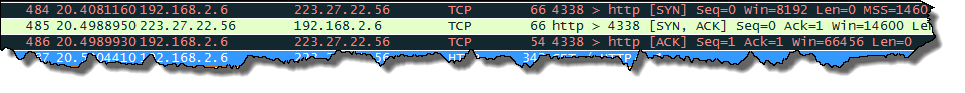
Type *tcp* in the Wireshark filter box (the background will turn green) and hit ‘Apply’. This will temporarily hide all non-TCP packets, which will make life a little easier.



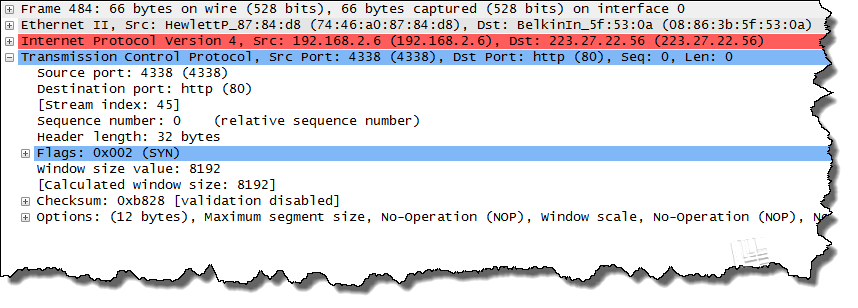
The generic format for a TCP connection is SYN X, ACK X+1,SYN Y, ACK Y+1. With that in mind, Go to the uppermost frame on Wireshark, and scroll down until you see 223.27.22.56 for the first time. At that point you will find the triple handshake between your machine and the webserver. Make sure that the IP addresses match. You will find:

1. A SYN packet from your browser (client) to the server
2. A SYN/ACK packet from the server back to your browser
3. An ACK packet from your browser to the server

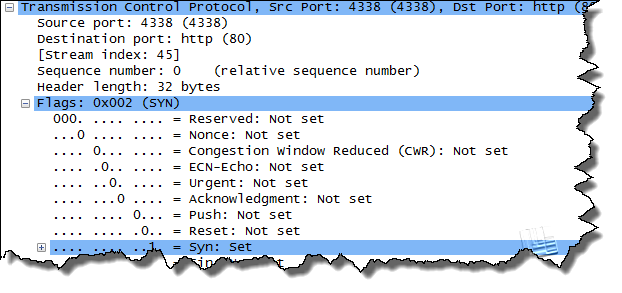
**SNAG YOUR RESULT**



Select the first packet in the handshake, and look at the centre window. Expand (only) the TCP header by clicking on the [+].



Now we are going to look at the flags. Expand the by clicking on the [+] next to them.

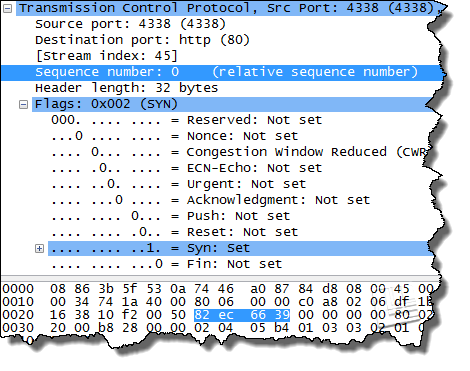


Note the following:

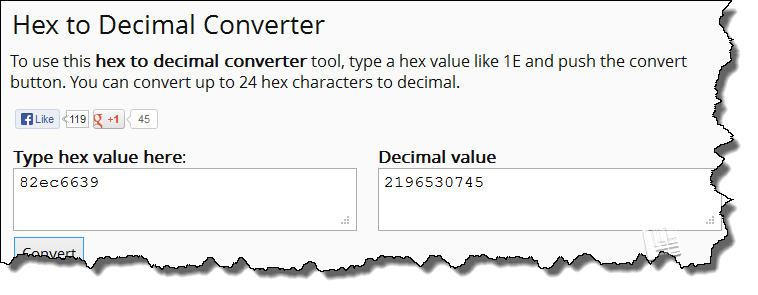
1. Source port (on the browser side) = 4338, which is a registered port >1023 and <49151.
2. Destination port (on the server side) = 80, which is a well-known port.
3. S flag is set.

The question is now: what is the Initial Sequence Number (ISN) proposed by the client?

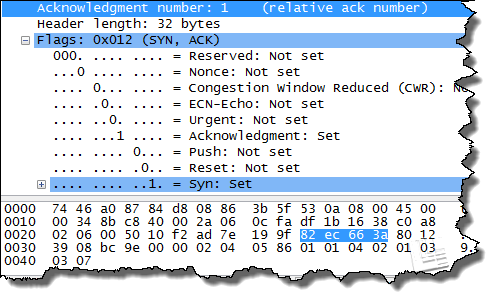
Highlight the Sequence Number (two lines above the flags). Wireshark shows it as 0, but this is not the real value…it is only a relative value to assist you. The actual value (see hex in the bottom frame) is 82ec6639 hex (0x82ec6639).



The decimal equivalent is 2196530745. The first leg of the 3-way handshake is therefore:  
SYN 2196530745.



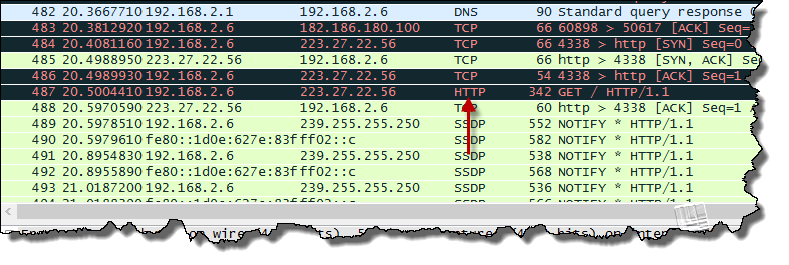
Now highlight the SECOND leg of the handshake on the top frame, and look at the Acknowledgement Number. Its relative value is 1 (so it has, quite correctly, incremented) and its actual value is 0x83ec663a, which is 1 more than its original value.



In similar fashion you can check the server’s proposed ISN in the second packet of the handshake with the client’s corresponding Acknowledgement Number in the third packet.

This concludes the handshake.

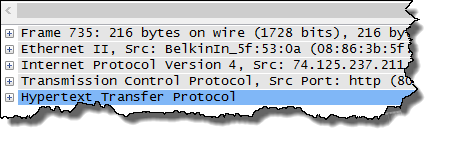
Now clear your filter, and put in a new one viz. http, then ‘apply’. Just below the handshake you will see a GET request from your browser. This is the browser requesting the page.



If you scroll further down you will now see HTTP coming into play: the packets comprising the web page are transmitted using HTTP over TCP, over IP.

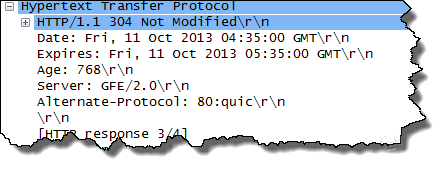


Select one of the HTTP messages, resulting in the 4-layer sequence (HTTP/TCP/IP/Ethernet as shown in the following image.



Open up the HTTP header (only) and snag it. Note that the blue inderlined text below it is for analysis only, and does not represent specific header fields.

**SNAG YOUR RESULT**



|  |  |
| --- | --- |
| **Exercise 16b:** | **UDP** |
| **Remote Lab PC:** | **Labs 1 & 2** |
| **Protocol Analyser:** | **Wireshark** |
| **Frame Generator:** | **Ostinato** |
| **Remote Lab Type:** | **D** |

**Objective**

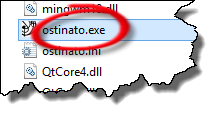
To analyse the structure of a UDP packet (frame).

**Preparation**

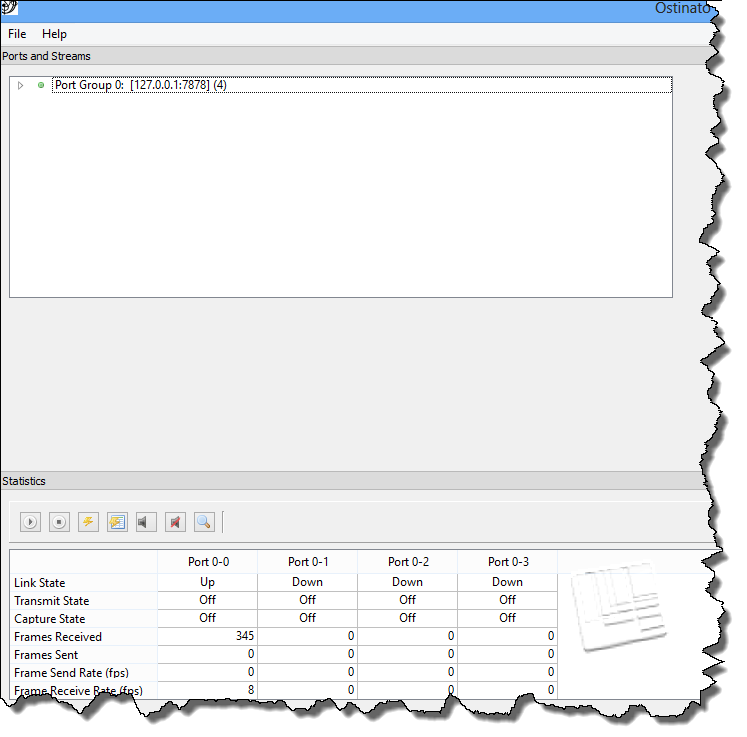
Run Wireshark and Ostinato on Lab 1 or 2 in the Remote Labs (or search on Electromeet). If you are interested in the Ostinato dicumentation, you will find it on the Ostinato site. Download Ostinato [here](http://ostinato.org/pricing) -> Ostinato Trial -> Try. You will receive an email with the download instructions. Newer versions of Ostinato may differ slightly from the following screenshots.

**Instructions**

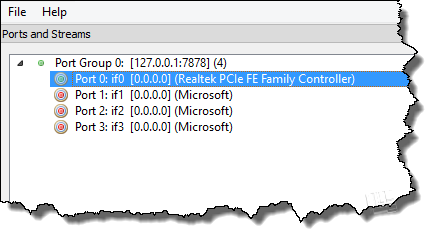
Run Ostinato by clicking on the executable.



The program will open.



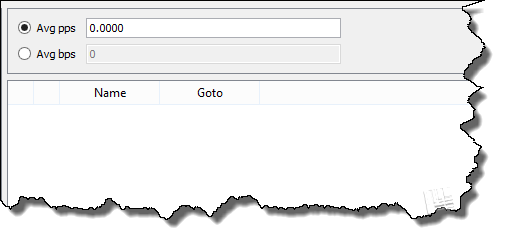
Double-click on Port Group 0 (top left). A drop down list with available interfaces will appear. We will select the Ethernet interface (see below).



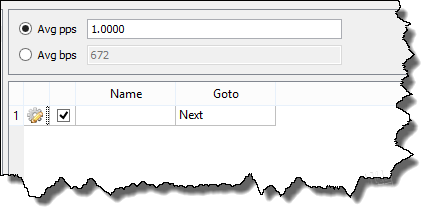
Another box will open in the top right-hand corner.

**The antivirus program might object. Just tell it to allow the program to run.**

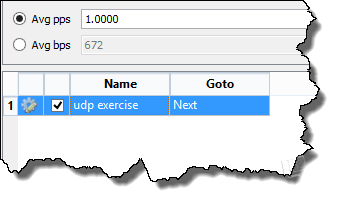




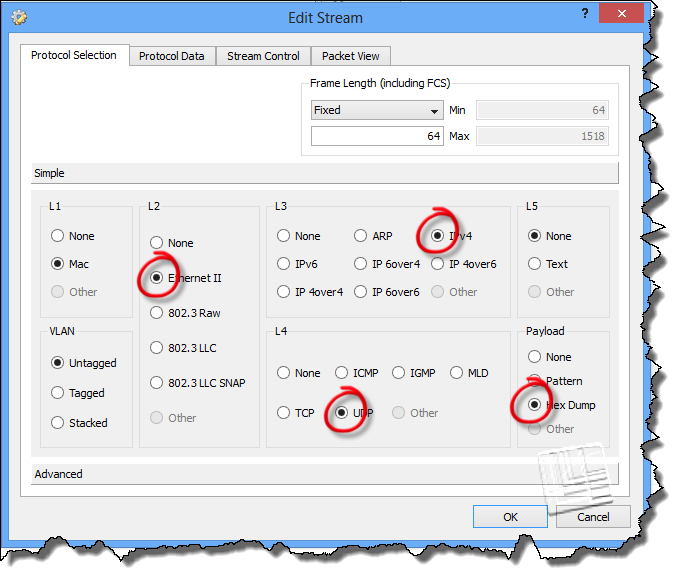
Right-click within this area. A menu will drop drown and we will select ‘New Stream’.



Double-click in the ‘Name’ field and give it any name you like.



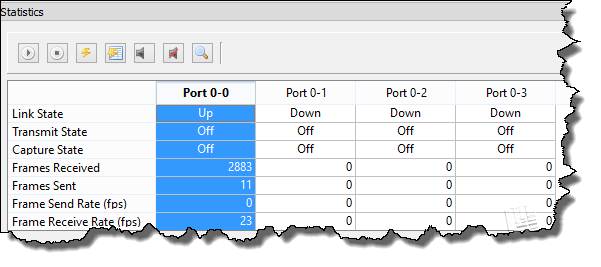
Now, once again, right-click in this area, and select ‘Edit Stream’. Use the radio buttons to configure a packet with a hex payload, using UDP over IPv4 over Ethernet.



Click on the other tabs to check the packet composition and default parameters. Note that the default rate is 1 packet per second.

**NB DO NOT FORGET TO CLICK ‘APPLY’**

In the statstics window, select the same port (select the whole column by clicking on the column heading) for which you configured the stream (IMPORTANT).

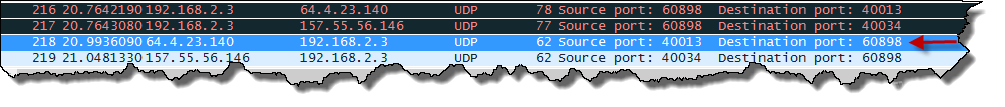


Now click on ‘Start Tx’ in the bottom left-hand corner.

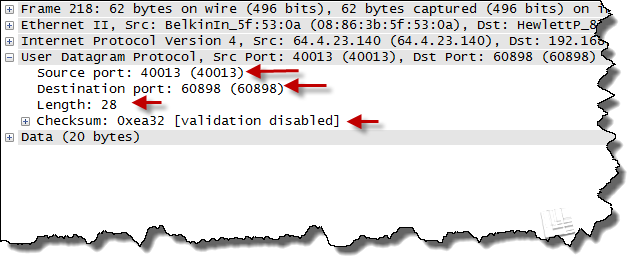
It might be a good idea to close your browser in order to reduce the amount of captured traffic.

Run Wireshark and check the bottom of the list until you see the UDP packets appearing. Then stop.

Select any of the UDP frames.



Now open up the UDP header.



Note the following:

1. The source and destination ports are used for identification only (no connection)
2. No handshaking
3. Header length only 8 bytes (the 28 includes the 20 bytes of application data).

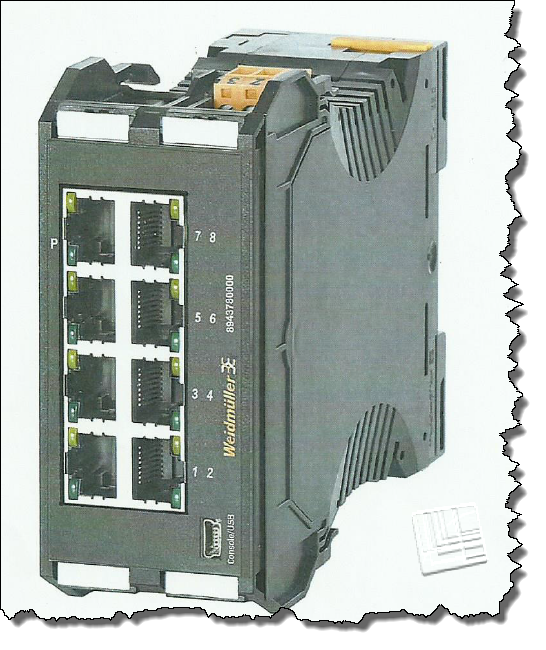
|  |  |
| --- | --- |
| **Exercise 38:** | **Managed Switch Configuration** |
| **Remote Lab PC:** | **Labs 1** |
| **Switch:** | **Weidmuller IE-SW-M Wave** |
| **MIB Browser:** | **ManageEngine MibBrowser - rfc mibs#1213, 1493, 1573,1642** |
| **Remote Lab Type:** | **A** |

**Objective**

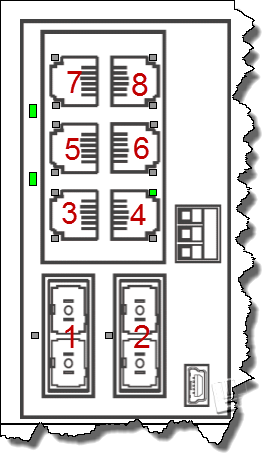
The objective of this exercise is to perform basic configuration and diagnostics on a Weidmuller IE-SW-M-Wave Industrial Ethernet managed switch.

**Background**

The Weidmueller IE-SW-M-Wave is an 8-port managed switch with port trunking, VLAN, Rapid Ring and other capabilities. See para 1.2 on p8 of the IE-SW-M-Wave.pdf manual (Annexure A).



For this paricular exercise we will use a version with 2 fiber ports and 6 copper ports as per the following figure. The Remote Lab 1 computer is connected to port 4, which will serve as our management port.



Also refer to the picture on the next page. You will notice the Cat5e flylead connecting the switch with the Remote Lab computer. On one side of the switch you will see the 24V power supply (with the bright green LED), and on the other side is a Weidueller IE-ARM-E router, not used for this particular exercise. The router is the device on the left, and the Cat5 cables to the Ethernet ports on the router were temporarily removed before the photo was taken.

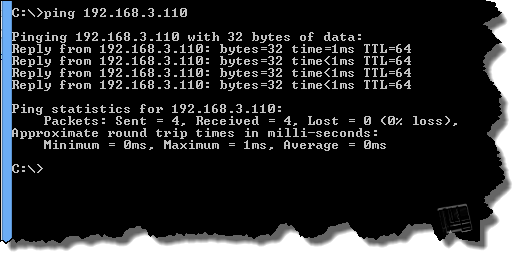
The power supply is on the right.



**Instructions**

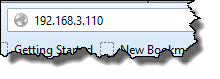
Log into the Electromeet, open remote Lab 1.

Ping the switch to ensure that it is present on the lab network. Note the IP address of the switch, below. The factory default IP address is 192.168.1.110, but this has been changed to 192.168.3.110 to fit in with the Lab IP addressing scheme.

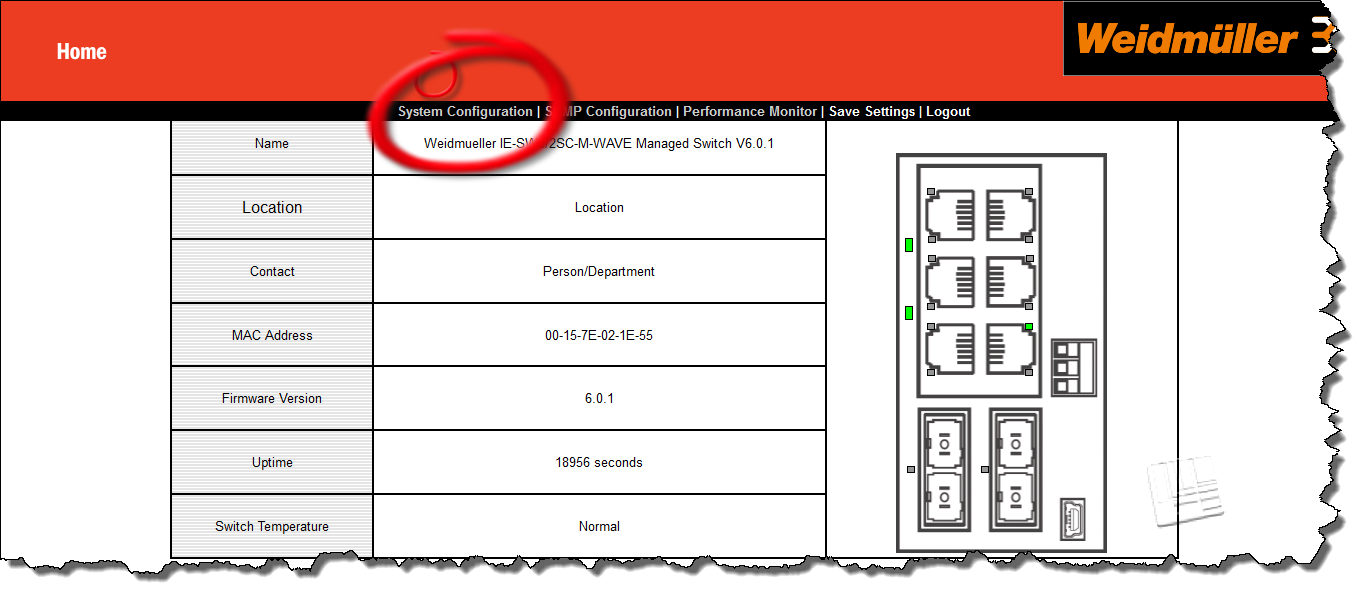


Note that, unlike the Cisco router in Remote Lab 4 (Exercise 39) that uses Telnet for configuration via its Ethernet port, this device uses a browser (as in the case of ADSL routers and most wireless Access Points). This switch can also be configured via its USB port connected to the serial port of the configuring PC (with a suitable cable and driver software), in which case the Web interface cannot be used.

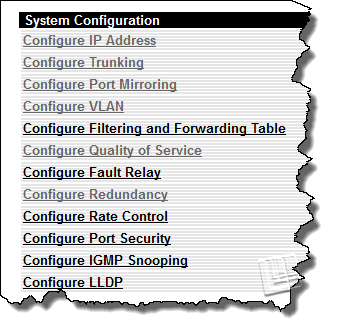
Now open (any) browser and type the switch IP address in the search bar. No need for ‘http://’



When prompted, enter username admin and password detmold. The home page of the firmware will open.



Now click on System Configuration.



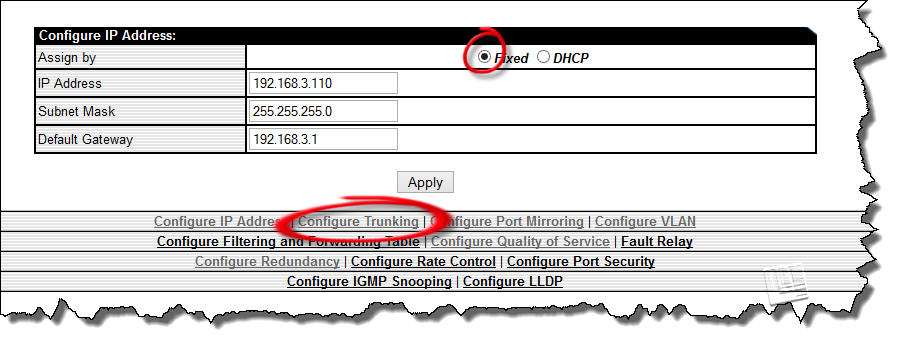
**IP ADDRESS CONFIGURATION**

**PLEASE DO NOT CHANGE THE IP ADDRESS**

We are not going to touch this setting, but there is no harm in having a quick look. Click on Configure IP Address (first item in the figure above). The IP configuration page will open.

In the screenshot below, note the fixed IP address of 192.168.3.110/24. **Once again; please do not change.**

**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER (SS1)**

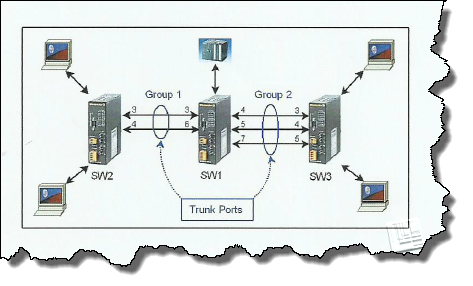


From here (see the red circle above) we can go to Port Trunking. Note that you do not have to go back to the main manu to access other functions, as the menu is repeated on every page, as in the screenshot above.

**PORT TRUNKING**

**PLEASE DO NOT SAVE ANY SETTINGS**

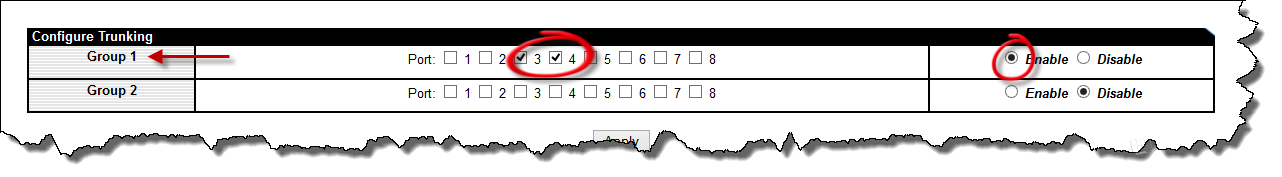
First read section 4.1 on pages 14, 15 and 16 in *IE-SW-M-Wave.pdf* (Annexure A). Refer to the figure on p14, reproduced here.



For the sake of simplicity we are going to assume that ‘our’ switch is SW2, hence we are only going to configure one group (Group 1) consisting of ports 3 and 4. That’s it! In real life we would now click ‘Apply’, and also proceed to configure trunking on SW3 and SW4. The creation of a trunk from SW2 to SW3 gives us a 200 Mbps data rate between the switches in Group 1. It also creates redundancy in case one of the links fail.

Create a ‘Group1’ for ports 3 and 4, and enable it.

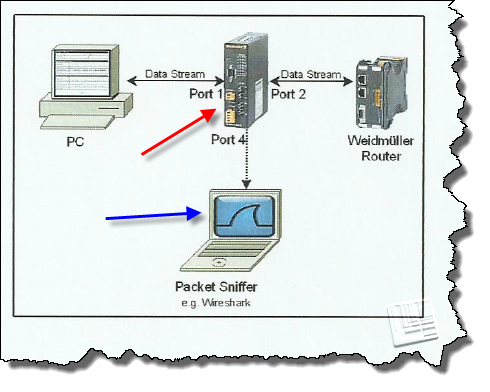
**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER (SS2)**



**PORT MIRRORING**

**PLEASE DO NOT SAVE ANY SETTINGS**

First read section 4.2 on pages 16 thru 19 in *Weidmuller Switch.pdf* (Annexure A). Refer to the following figure on p16.



In the figure above you are logged into the ‘Packet Sniffer’ (RL1) running Wireshark (blue arrow), and you are connected to the managed switch (Red arrow). The other PC and the router are not connected at present, but imagine that they are connected to ports 1 and 2 as per the figure.

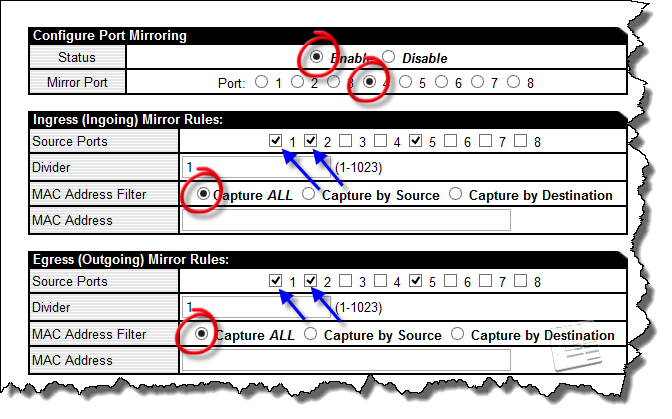
The problem is that Wireshark would not ‘see’ any traffic between the PC and the router, because of the way in which a switch operates. The solution is to mirror ports 1 and 2 to port 4.

Note the following:

* We can mirror any ports we want to
* We can set rules for both traffic going into the switch (ingress) and traffic going out of it (egress)
* We can capture all packets, or filter on source/destination MAC address (entered in hex without colons or dashes)
* In cases of very heavy traffic we can set a ‘divider’ so we do ot capure all packets and overload the switch in the process.

In the following example we are mirroring ports 1 and 2 onto port 4, with no filters.

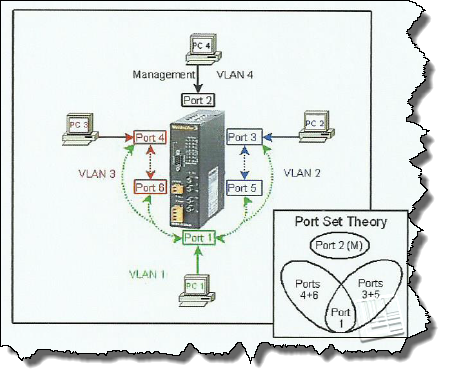
**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER (SS3)**



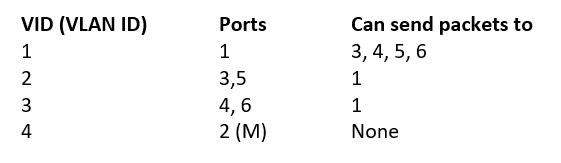
**VLAN**

**PLEASE DO NOT SAVE ANY SETTINGS**

First read section 4.2 on pages 19 thru 23 in *IE-SW-M-Wave.pdf* (Annexure A). Refer to the following figure on p21. You may want to read it several times to ‘get the hang of it’.

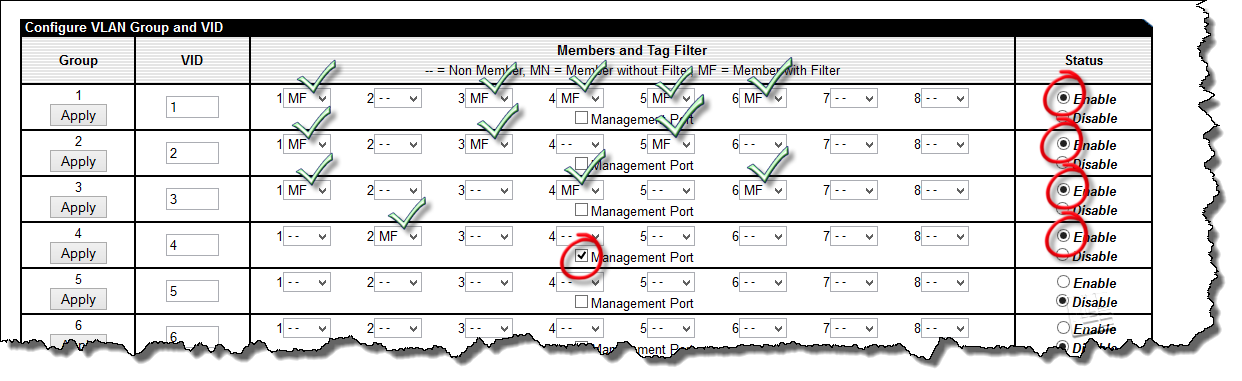


In order to keep things simple we are going to follow the example in the IE-SW-M-Wave manual, and set up four VLANs as follows. Port 2 is the management port. VLAN 1 allows poer 1 to send packets tpo 3, 4, 5 and 6, and VLANs 2/3 create return paths for 3,5 and 4,6 respectively.



First we have to create the VLANs.

**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER (SS4)**

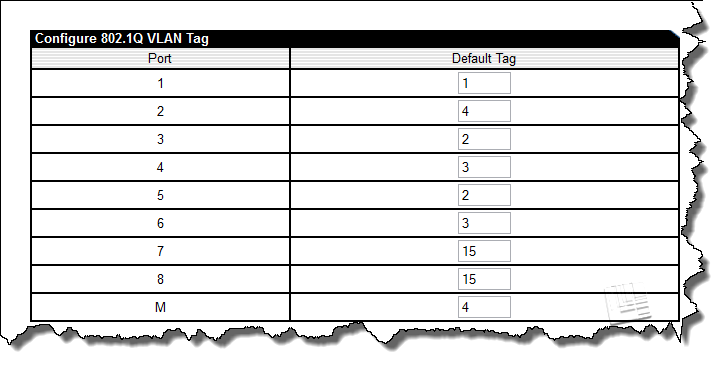


Continue to page 2 of the VLAN settings.



Then we have to set the IEEE802.1Q VLAN Tag or ‘VID’ for each port as per the following example, so that they match our previous allocation.

**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER (SS5)**



Note that port 2 appears twice, because it is also the management port. We are not allocating ports 7 and 8 to any VLAN.

**SNMP**

You may want to do some reading on the topic; here are a few links:

* <http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol>
* <http://oreilly.com/perl/excerpts/system-admin-with-perl/twenty-minute-snmp-tutorial.html>
* <http://www.manageengine.com/network-monitoring/what-is-snmp.html>

Unfortunately SNMP (Simple Network Management) is anything but simple (for beginners, at least) and often its implementation on a specific device is poorly documented as well. This switch is no exception. However, let’s see what information we can extract from the switch.

The following is a snippet from the IE-SW-M-Wave MIB documentation.

*ieswSwitchFaultStatus OBJECT-TYPE*

*SYNTAX INTEGER {*

*fault(1),*

*noFault(2)*

*}*

*ACCESS read-only*

*STATUS mandatory*

*DESCRIPTION*

*"A value which indicates the fault status of the switch.*

*The meanings of the values are:*

*fault(1) - fault has occured.*

*noFault(2) - no fault has occured."*

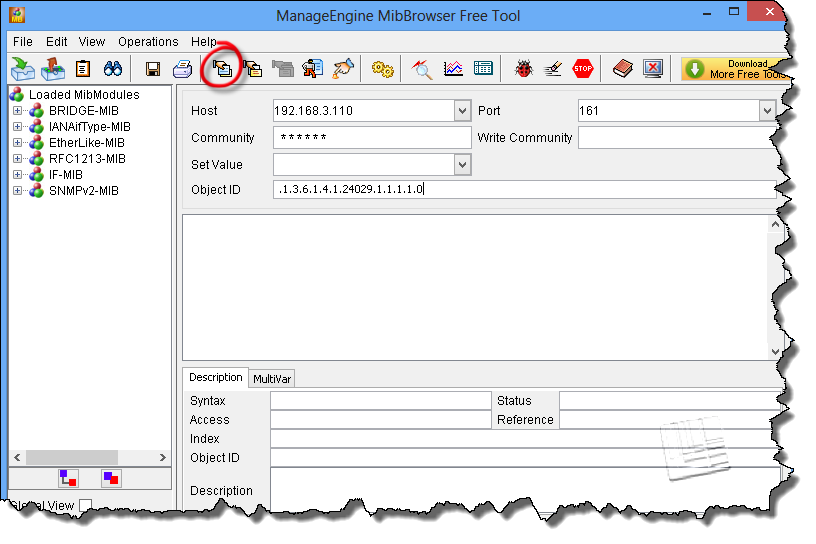
*::= { ieswFault 1 }*

Note that 1= fault, and 2=no fault.

Now look at p41 of the switch manual, under OID (Object Identifier). The OID for Switchfault status is

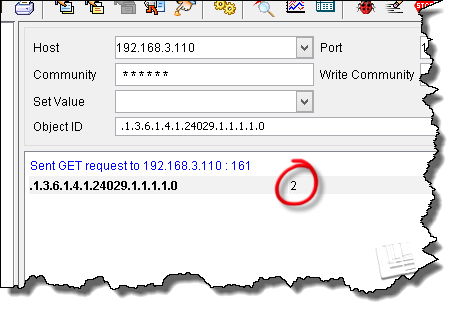
.1.3.6.1.4.1.24029.1.1.1.1.0.

Start the ManageEngine Free MIB Browser Tool.



Enter the IP address of the switch (‘Host’), as well as the Object ID (‘OID’) for Switchfault status.

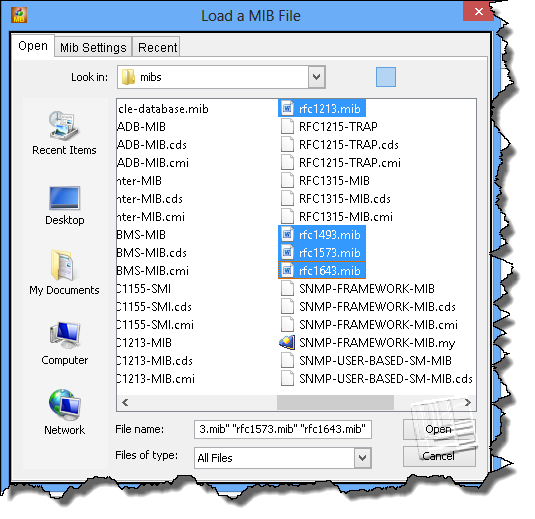
Click on the icon for ‘GET SNMP Variable’ (circled in previous screenshot). Note the returned value of 2 (= no fault).



Now this is all fair and well, but because we do not have the appropriate MIBs loaded, hence the information we can retrieve is rather sketchy and in numerical format only.

We need to load four MIBs, viz. rfc1213.mib, rfc1493mib, rfc1573.mib, and 1643.mib as supplied by Weidmueller.

Click File->Load MIB and select the four MIBs. You could, of course, take the easy way out and ‘Load All MIBs’ (unless they already appear in the ‘mibs’ folder as shown below).



Now click ‘Open’. The dialog box will indicate successful loading of the MIBs.

Start with RFC1213-MIB and ‘drill down’ as follows:

->org

->dod

->internet

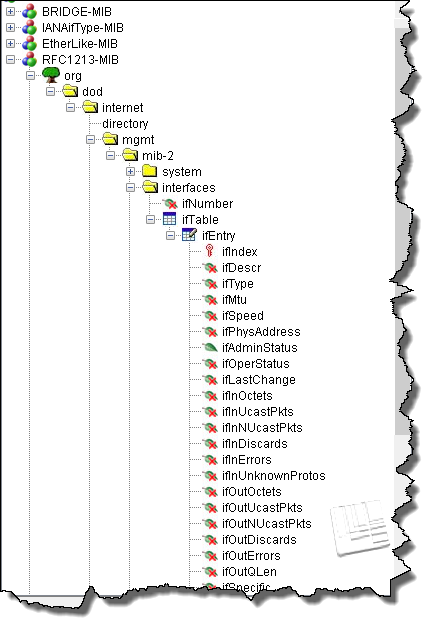
->mgmt

->mib-2

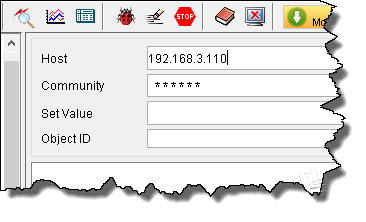
->interfaces

->ifTable

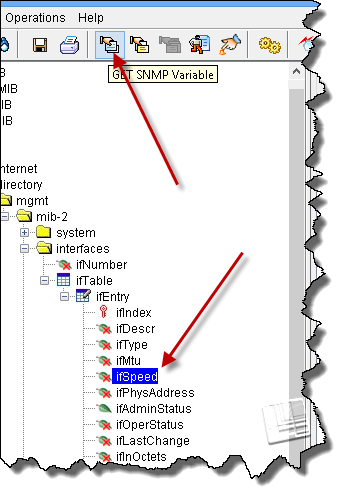
->ifEntry



Do not forget to set the Host address to the switch’s IP address. It defaults to’localhost’ and if you fotget to change this, you will be trying to GET values from the Remote Lab computer and not the switch.



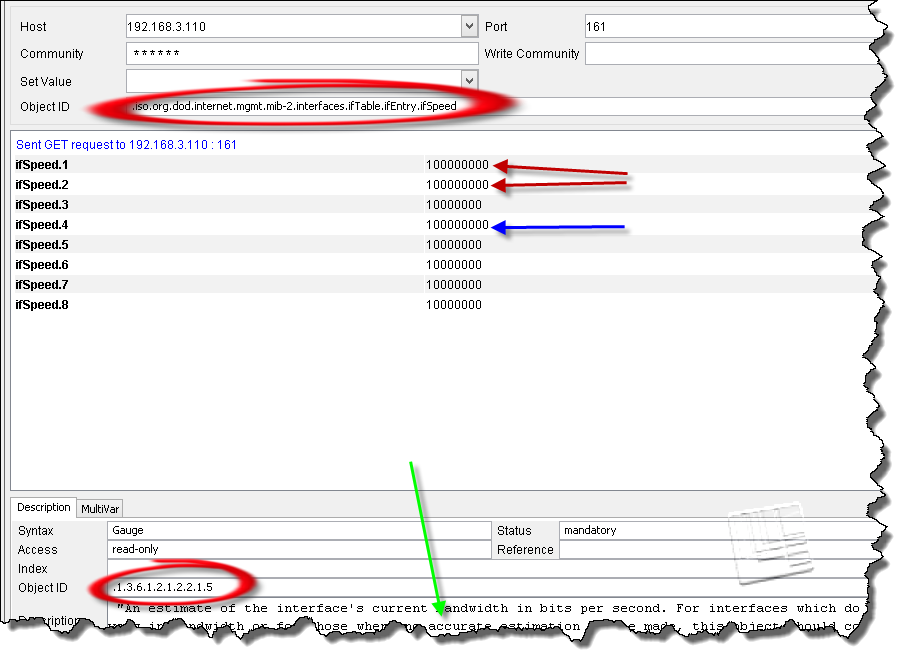
Select one of the variables ( say ‘ifSpeed’ ) and click on ‘Get SNMP Variable’. This will get the speeds of the eight switch ports.



The following results are obtained. Note the description (green arrow) and the OID in numerical format as well as text (circled).

Ports 1 and 2, being fiber (100BaseFX) ports, cannot auto-negotiate and are always running at 100Mbps (red arrows). The copper ports, being able to auto-negotiate, default to 10Mbps since they are not connected. The exception is the management port (#4), connected to the Remote Lab PC, which has connected at 100 Mbps in this case.

**TAKE A SCREENSHOT OF THIS IMAGE ON YOUR COMPUTER, and any additional ones if requested in your assignment.**



**EIT wishes to acknowledge the contribution of Weidmuller Australia in donating the quipment for this exercise.**

Weidmuller Pty Ltd

43 Huntingwood Drive

NSW, 2148 Huntingwood

PO Box 6944

Blacktown, NSW 2148

+ 61 (0)2 9671 9999 Tel

+ 61 (0)2 9671 9900 Fax