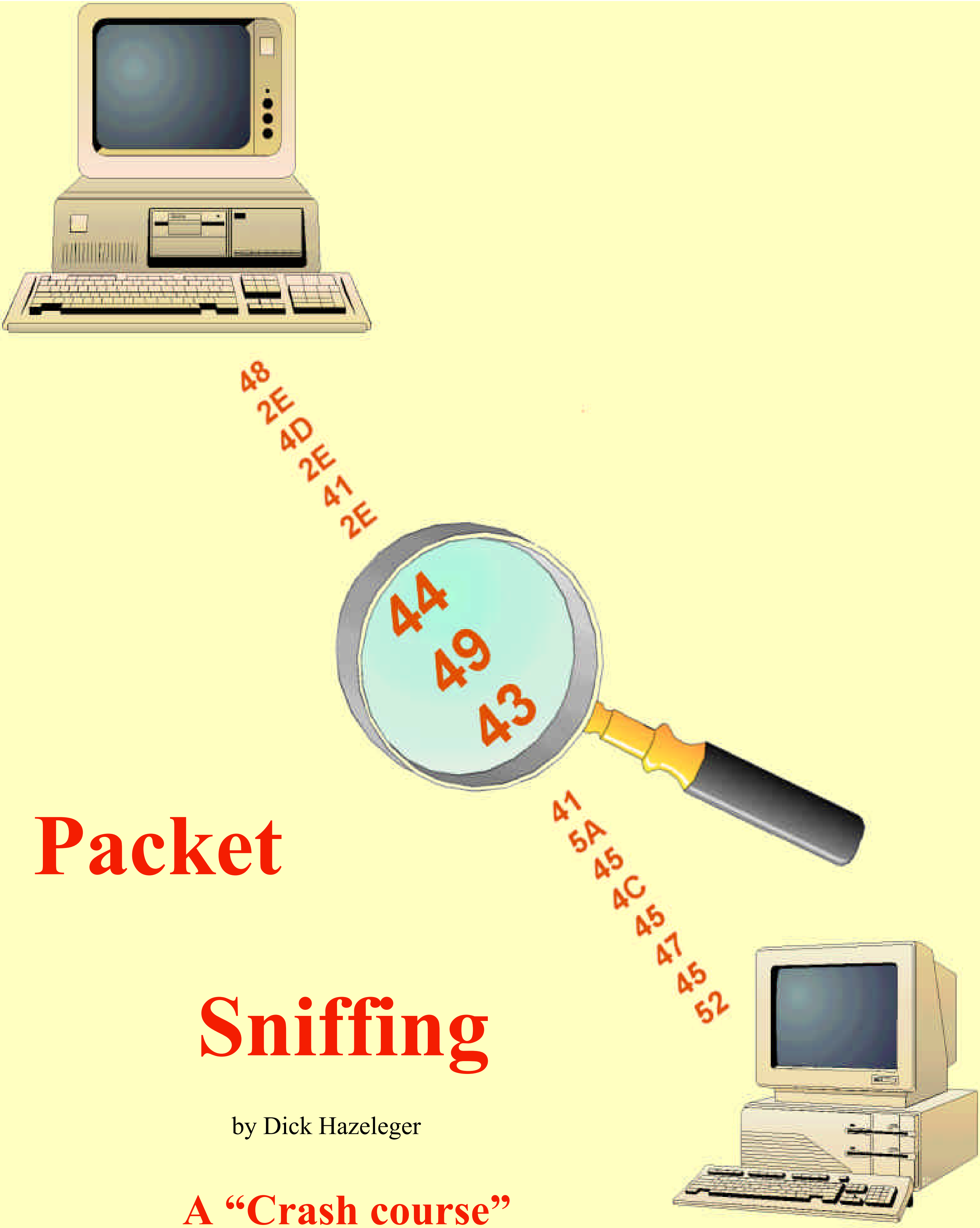


Packet Sniffing

by Dick Hazeleger

A “Crash course”

Credits

Realization of this document wouldn't have been possible without the direct or indirect help of:

The team at **http://ethereal.zing.org** for writing their excellent Ethereal Packet Sniffer program.
The team at **http://netgroup-serv.polito.it/winpcap/** for writing the WinPCap driver.
Cigar for proofreading this document so patiently.

Table of Contents

Publication date	Release	Remarks
03-03-2001	1.00	Initial publication

1. Introduction

I think you will agree that a lot can be done by almost anyone with today's computer technology. Unfortunately - and this seems to be a "human failing"- these possibilities aren't always used with the interest of a community, smaller groups or even individuals in mind, and nowhere is that more true than on the Internet.

For those who have access to the Internet threats are numerous: Viruses, Trojan Horses, Hacker attacks on the system in use and - more and more as it seems - the threat of having to give up privacy (be it willingly or unwillingly) while on the Internet.

For at least a year the computer user's privacy has been threatened by software that is either free and marked as "Adware" or sometimes even as "Freeware" without any indication that the software has a different price for you to pay"; or is installed from a CD that came with a magazine; or software that has been pre-installed by the computer manufacturer. Some of these programs install a delivery system for the ads that will be displayed when the program is executed.

In the last year it has been proved that the programs with cycling, ever changing, ads, which require an Internet connection to function, also keep tag of which ads are clicked, how long the user stayed on the page; one or two of these ad delivery systems even track the surfing habits of their users. This all is done largely without prior consent of the computer **owners**… YOU! They act the way spies work, hence the name "spyware" was given.

Without entering into a discussion over the "right or wrong" of spyware, when you're like me, you want to know what's going on in your system and even better, you want to know what's going out as well!! Due to popularity and ease of use, people tend to use port monitors to detect possible "spyware". This won't work: the why will be explained in the next chapter. The other, less popular and more difficult technique that actually **can** prove that a program (or even a Website) is spying on you is "Packet Sniffing". It's the scope of this "Crash Course" to explain the necessary technical information, what programs can be used and how to read the captured data.

I hope this course will be of use to anyone who wants to prevent his or her privacy from being "general knowledge"!!

For questions or remarks you can contact the author at: info@hazeleger.net

2. Port Monitoring versus Packet Sniffing

Besides "Packet Sniffing" another technique is very popular to detect possible spyware programs: Port Monitoring.

A Port Monitor is a program that is functioning in the background while the other programs are executed. If a program accesses the communication devices (COMM: devices) such as a Comm-port (with a modem connected to it), then this access is noted and logged.

Since several programs may have very legitimate reasons for accessing these COMM: devices without the knowledge of the user, this could end up with the user saying: "I have to get rid of this or that program, because it's spyware". On the other hand, a program accessing a COMM: device, doesn't mean it will broadcast information over an open connection, it's very possible that programs could access the COMM: devices to check the status of the device, or to send internal commands to the device that won't leave the system.

For example I tested a Port Monitor early this year and I used a general terminal program from a well-known manufacturer. When I executed the program, it showed the normal "terminal prompt" and I could access my modem, sending commands to it and reading the status from it. When I looked in the Port Monitor program, it looked like it was in an endless loop: the information kept coming in! Now this is easy to explain, the terminal program keeps checking the COMM: device (modem in this case) and these "read" accesses were noted and logged by the program (Note: there was NO open connection at that time).

So, we may conclude from this that a Port Monitor might be handy to have a first look; however it **never** can give you the definite proof that a program is actually sending out data about you (and whereto…).

Using "Packet Sniffing" as a detection tool however can give you the necessary proof, not only that data has been sent but also what data and to which destination (note: data may be encrypted in some way) and eventually what responses came back to your system.

Packet sniffing can be done on **any** network and every type of connection to the Internet (DUN, ISDN, ASDL, Cable), while Port Monitoring will only work for those connections that go through COMM: devices (Only DUN always does this).

Since we are going to install and use Packet Sniffing on our systems to find spying software with a "calling home" capability through the Internet, we will be mostly dealing with TCP/IP. In the next chapter I will give a very brief introduction in TCP/IP (there are very good books available for those who want to know the exact facts in detail). text moved

3. TCP/IP - Heartbeat of the Internet

TCP/IP (an acronym that stands for **T**ransport **C**ontrol **P**rotocol / **I**nternet **P**rotocol) is used on many networks **and** the Internet to transport data from one computer to another and back. During a session on the Internet your computer isn't in constant contact with the server (or another computer). To give each system on a server an equal chance to have a connection with this server, the server divides the time it has to spend on communication with other computers into little parts.

To achieve this, the data from the server and from the remote computer must be fit in a package that can be accepted in a time slice. Such a package is called "**a packet**". With only one computer on a server this is simple: the packet either comes from "A" and goes to "B" or it comes from "B" and goes to "A"; yet on the Internet (with its millions of servers) this isn't that easy, the servers need to know which computer asked for what and how to find it and send it back to just **that computer**!

This is where the IP-address comes into the picture. The IP-address is a long number (up to twelve digits) in the format: **###.###.###.###**, where every block a number in the range 0 - 255. A valid IP-address could be: 123.2.44.60 or 207.121.23.15., In order for this to work, every computer on the Internet must have a **unique** IP-address by which it can be identified.

If you have a dial up modem connection (DUN) your Internet service provider (ISP) will probably assign you, or "dynamically allocate", an IP from their block, or blocks of addresses on each connection. You will probably always have a similar IP address (for example, 123.123.123.xxx, where only the xxx varies from one connection to the next. On high-speed connections that are always on, your IP address is probably fixed and changes only rarely.

Now, when you issue a command to a server (for example: load web page www.xyz.com for me), the command will be sent to a server belonging to your ISP who will pass the command through to the appropriate computer. The IP-address of the source (your computer) and the destination (www.xyz.com) is in the packet containing the command, so every computer on the Internet knows where this packet originated from and where it has been sent to, just like when you send a letter: there has to be an address on it for delivery and a return address on it so the recipient can reply.

I can hear you think…. **HOLD IT!** I didn't type in any IP-address for that computer at www.xyz.com… and right you are. Now, at the time when you installed your Internet connection you had to enter information about your provider, amongst this information there were probably two IP-addresses, for the Primary DNS and for the Secondary DNS (DNS stands for **D**ynamic **N**ame **S**erver).

Every time you type in a WWW address (e.g. www.microsoft.com) and press the [Return] button, this address is sent to the Primary DNS, or if this server down or too busy, to the Secondary DNS. The DNS server translates the WWW address into an IP-address by looking it up in a database. This way we can use those, easy to remember, web-addresses instead of those long numbers. So, that is why you type in a "human readable" address and you end-up finding an IP-address.

By now we know about how an IP-address looks, how it's used and how names are translated back to IP-addresses by use of Dynamic Name Servers for the retrieval of information.

Back to our "packets"! Not all information (send or received) will have to fit into one packet. Therefore provisions are made within the protocols to allow information span multiple packets. These packets will show up in the sniffer file (the file containing the captured data) as "Continuation". Please note that packets may vary in length.

4. What programs to use?

There is a wide variety of packet sniffing programs. Some of these programs are "cheap" (US$ 100 or less), while others are "expensive" (US$ 100 or more). However there is a third category: FREE!! Yes, that is right: FREE!!

Although the choice is limited (of course), there are two programs that are actually full functioning packet sniffers while being free at the same time (AND: No Spying modules included!!):

The first one is **Windump**: as very simple packet sniffer that runs in an MS-DOS Window under Windows 95 or 98. The program doesn't feature logging to file or scroll back, so its only use is to detect and spot things while an observer is present at the time the program is executed. This program can be downloaded (together with the necessary packet driver) from **http://netgroup-serv.polito.it/**

The second program is **Ethereal**, a very complete packet sniffer that can be downloaded from

http://Ethereal.zing.org

Ethereal is available for the following platforms:

- Linux (2.0.x, 2.1.x, 2.2.x, 2.3.x, 2.4.x)
- Solaris (2.5.1, 2.6, 7)
- FreeBSD (2.2.5, 2.2.6, 3.1, 3.2, 3.3)
- Sequent PTX v4.4.5
- Tru64 UNIX (formerly Digital UNIX) (3.2, 4.0)
- Irix (6.5)
- AIX (4.3.2, with a bit of work)
- Win32 (NT, 98)

So, be sure you download the right distribution files for the platform you use.

Note: *The download page mentions certain libraries to be downloaded too, please be sure to download them as well, they are needed while executing Ethereal!!*

Ethereal uses the same packet-capturing driver as WinDump. On Ethereal's download page a link to the page where this driver can be downloaded is given or simply jump to:

http://netgroup-serv.polito.it/winpcap/.

Be sure to download it too.

By now we should have the following:

- Ethereal executables for your platform (I'll assume Win32 (NT/98) in this course
- Necessary libraries
- Packet capturing driver

This means that we are ready to install the whole thing, so we can use it. Before we do so, however I want to point you towards a program we will use in Chapter 9, when we will be tracking down more usable information from IP- addresses by using Sam Spade. For now it's enough to know that this is a multi-functional tracking program, which can be downloaded from **http://www.samspade.org .**

5. Installing the packet-driver

A packet sniffer only can function when there is a connection between the network or DUN-connection and the program. A driver that, of course, needs to be installed first provides this connection.

In this chapter we will install the WinpCap packet driver, which will enable Ethereal to do its job properly. Please note that you probably will need the Windows installation medium (e.g. CD) in the process.

Assuming that you have downloaded the driver's zip file to a download directory, please execute the following steps:

1. Unpack/Unzip the archive file to a directory of its own.
2. Now {Right-click} on the "Network Neighborhood" icon on your desktop.
3. Click on "Properties" in the little menu that will appear.
4. A new screen will appear. Now click on the [Add] button.
5. From the new window that will appear, select "Protocol" by clicking once on it, now click on the [Add] button in this window.
6. Again another window will appear, from this window select "Have Disk". Since the system doesn't know where to look, you'll get a window pointing towards A:\ with a [Browse] button to the right. Press this button.
7. Now find the directory to which you previously unpacked the packet driver archive. If you have found that directory and selected it, you'll see a name (packet.inf) appear, select it.
8. Now you can click on [OK], the top windows will disappear now.
9. Again press [OK] and the packet driver will be installed.
10. Windows may ask for its installation medium at this point, although cases have been reported where this wasn't asked; however when it asks (e.g. CD), insert it in the appropriate drive when asked for it. When all this is done, remove the installation medium, close all windows and finally
11. REBOOT your system (this is very important!)

After the system has been rebooted, we'll go back to the "Network Environment" to check if the installation was successful.

OK, now the system has been perform the following steps:

1. {Right-click} on the "Network Neighborhood" icon on your desktop.
2. Click on "Properties" in the little menu that will appear.
3. Scroll down the list.
4. You should see an entry called "Network Packet Driver for Win 95/98 v. #.##" (#.## is the actual version number).
5. Select this entry by clicking once on it. In the description field the text "Network Capture Packet Driver for NDIS-adapter" will appear. (*Note: It is possible that - depending on your Operating system – no contents in the description field will be shown*).

If all this is correct, then the packet driver is installed correctly and should be ready to do its job.

6. Installing Ethereal and a brief explanation

Installing Ethereal actually is a cinch. Although the program lacks an installation program, anyone who has a bit more than basic knowledge of the Windows Operating System should be able to install it within five minutes. For those, not so familiar with the Windows Operating System here are the necessary steps:

A. Creating the environment.

The "environment" meant here isn't much more than a set of folders (directories) in which Ethereal, its libraries and - in a later stage - the sniffer files will be stored. To make this environment proceed as follows:

- From [Start], {Programs} start Windows Explorer
- In the left pane select the place where you would like to install Ethereal (normally: **C:\Program Files**" or "**C:**\")
- Let's assume we will install it in C:\Program Files. Now open this folder by double-clicking on it.
- Now that the "Program Files" folder is opened, {Right-click} on an empty spot in the right pane. A small menu should appear.
- From this menu, select "New", followed by "Folder".
- The screen should jump to the bottom of the Explorer window now, showing the folder icon and the text (white on blue) "New Folder". Type now "Ethereal" and press the [Enter] or [Return] button on the keyboard.
- Open this new folder by double-clicking on it.
- Again {Right-click} in the right pane, select "New", followed by "Folder".
- Now, type in "Sniffer files" or "Captured data" or whatever name you would like to give to this location.

When this is done, the "environment" in which we will install Ethereal is ready to use.

B. Installing the libraries and the program.

"Installing" is actually a big word, the only thing that has to be done is that the archive-files will have to be unpacked to the "C:\Program Files\Ethereal" folder. This has to be done for the libraries as well as for the program itself. After this has been done, the program should be ready to use. However it's not convenient to start it from Windows Explorer every time we need it, so we must make a shortcut to the program which we can place in a location which is easier to reach… e.g. the Desktop.

To make a shortcut, from Windows Explorer go to the folder (directory) where you installed the Ethereal program. Depending on your settings you could see (amongst many others) a file called "Ethereal.exe" or just "Ethereal"; the file can be recognized by its icon, which is:

Be careful to select the right file, because there is another file called "tEthereal.exe" or "tEthereal" with the same icon in the folder!!!

After you've found the right file, {Right-click} on its name and select "Make shortcut" from the menu. A new file called "Shortcut to Ethereal.exe" now should appear. Press [F2] to edit the name… e.g. "Ethereal Sniffer" and press [Enter] or [Return] to finish or just leave the name what it is.

Now all we will have to do is place this shortcut somewhere we can access it easily, in this case I'll place it on the desktop. Select the shortcut by clicking once on it. Now press - simultaneously - [Ctrl] and [C]. You may close the Windows Explorer now or just minimize it. Now go back to the desktop and press - again simultaneously - [Ctrl] and [V]; the shortcut should be on the desktop now, ready for use!

On the following page(s) a brief explanation of Ethereal's main screen will be given, if you use a printed copy of this tutorial, it's best to start the program and have a look yourself.

C. Ethereal’s main screen

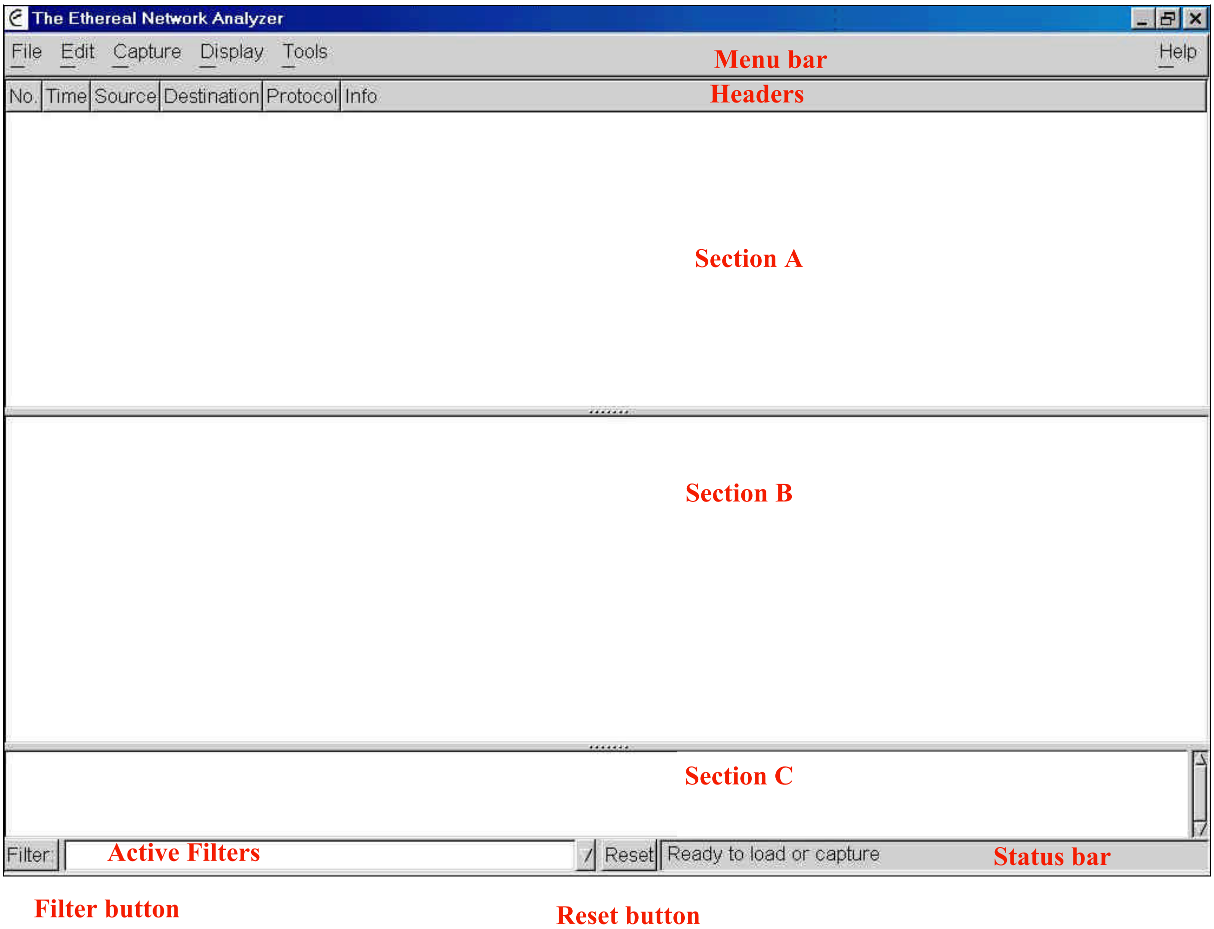


Fig. 1. Ethereal’s Main Screen

Once you’ve started Ethereal, you should see this screen.

Items typed in *Italic* are not always available, items typed in ***Bold/Italic*** are always available.

In the Menu bar we find the following options:

- **File**, containing the ***Open*** (capture) file, *Close*, *Save*, *Save as*, *Reload*, *Print*, *Print Packet* and ***Quit*** (program exit) commands.
- **Edit**, containing the *Find Frame, Go to Frame, Mark Frame, Mark All Frames, Unmark All Frames,* ***Preferences***, ***Filters*** and ***Protocols*** commands.
- **Capture**, consists of only one option: *Start*.
- **Display,** this menu item has the following sub items: ***Options***, *Match Selected, Colorize Display, Collapse All, Expand All* and *Show Packet in New Window*.
- **Tools**, which has the following items: ***Plugins***, *Follow TCP Stream* and *Summary*.
- **Help** (located at the right hand side of the screen) for the available help file and a box showing information about Ethereal.

The headers identify information that will be shown in Section A. By clicking on these headers you can sort the display on the information contained in each column.

Section A shows you the general frame information data, normally sorted by “No.”
In Section B Extensive information about the selected frame is given. Section C shows the exact contents of the frame, both in hex and in ASCII (if possible)

The [Filter] button on the lower left hand side of the screen enables the user to filter on various items for each protocol. The active filter is shown in the field next to the [Filter] button. If more filters are present, the user can select the appropriate filter by clicking on the downwards-pointing triangle at the right of the filter field. In this drop down list, an empty filter always exists; by selecting this one the user can easily disable all filtering.

The [Reset] button resets filtering and is needed when filtering messed things up. At the right of the [Reset] button you'll find the status bar; in this bar the program shows whether a live capture is going on or a capture file is loaded, and how many packet have been "dropped" (missed).

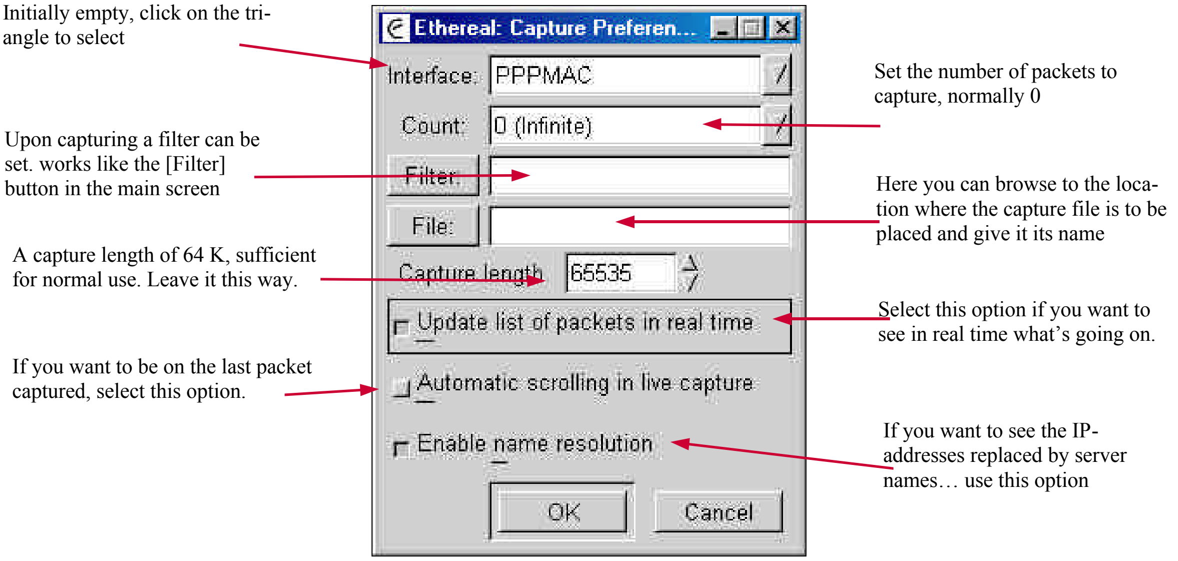
This concludes our preview of the program. Be sure to know where to find the commands. In the next chapter we will be setting up our first capture.

7. Setting up the first capture

Now that we have had all the introductions, it's about time to start capturing something. It doesn't matter to Ethereal when you start capturing, whether before any DUN-connection has been made (thus insuring us that everything will be captured), or after, somewhere during a connection.

If you do start capturing before a connection has been made, be aware that your login information (username, password) will be visible **in plain text** in the captured text, so be careful with whom you share this information.

Starting a capture can be done by hitting (simultaneously) [Ctrl] + [K] or by accessing the menu-item "Capture" through the menu bar. By using either, you'll get the following screen (or something alike):



Ethereal: Capture Preferen...

Interface: PPPMAC

Count: 0 (Infinite)

Filter:

File:

Capture length 65535

□ Update list of packets in real time

□ Automatic scrolling in live capture

□ Enable name resolution

Initially empty, click on the triangle to select

Upon capturing a filter can be set. works like the [Filter] button in the main screen

A capture length of 64 K, sufficient for normal use. Leave it this way.

If you want to be on the last packet captured, select this option.

Set the number of packets to capture, normally 0

Here you can browse to the location where the capture file is to be placed and give it its name

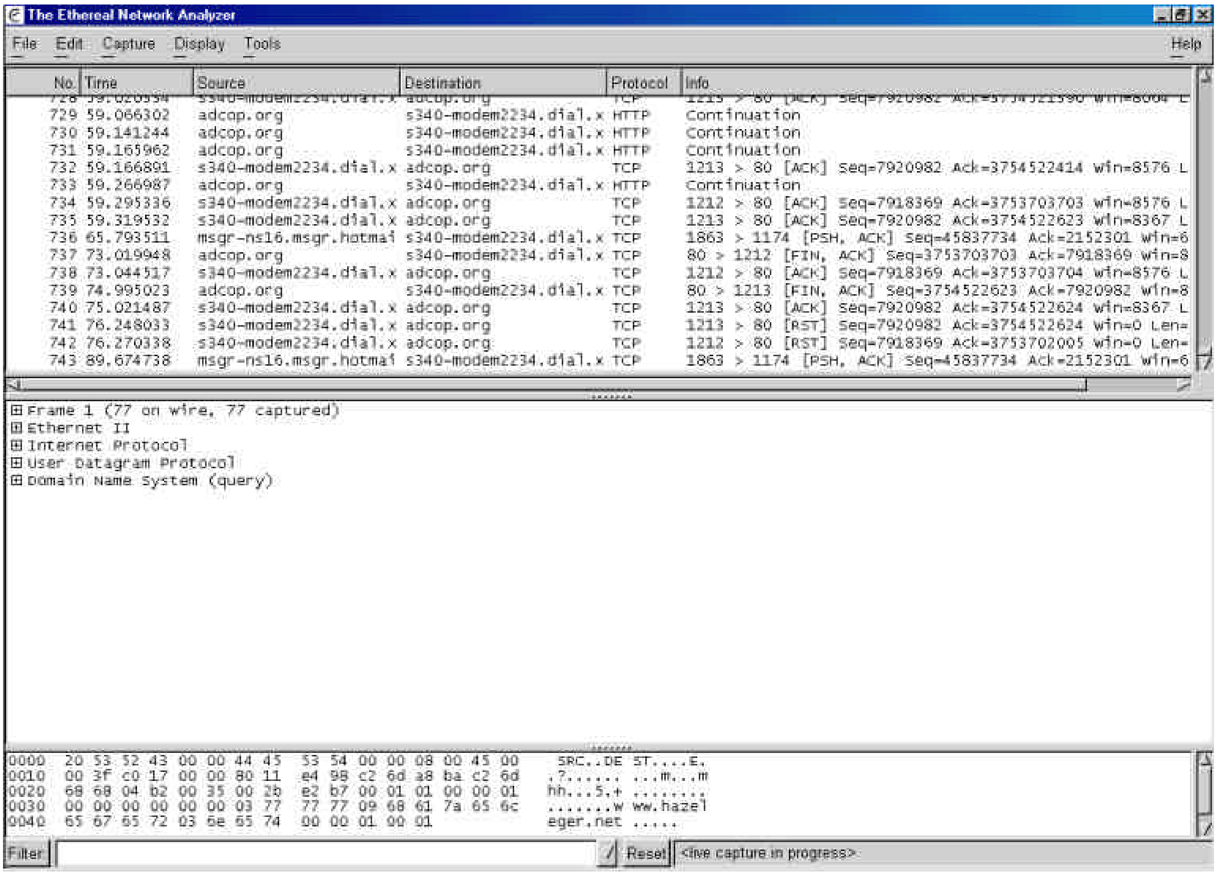
Select this option if you want to see in real time what's going on.

If you want to see the IP-addresses replaced by server names… use this option

Fig. 2. Setting up the capturing

Now that all settings have been set, we can start capturing. Click on the [OK] button in the window to do so. After you've clicked [OK], another window will appear showing the numbers of packets for the miscellaneous protocols that have been captured. This window can be minimized, but not closed (X), since this is the same as clicking on the [Stop] button: it would stop any capturing in progress.

If you haven't started your DUN-connection, now would be a good time to do so. The moment your system starts contacting to your provider's server you should see activity in Ethereal's main screen. A sample of this activity is shown on the next page. If you don't see any activity until you click on [Stop] you may have forgotten to select the "Update list of packets in real time" option!!



```
No. Time      Source                Destination           Protocol Info
728 59.020554 s340-modem2234.dial.x adcop.org             TCP      1213 > 80 [ACK] Seq=7920982 Ack=3754521590 win=8064 L
729 59.066302 adcop.org             s340-modem2234.dial.x HTTP     Continuation
730 59.141244 adcop.org             s340-modem2234.dial.x HTTP     Continuation
731 59.165962 adcop.org             s340-modem2234.dial.x HTTP     Continuation
732 59.166891 s340-modem2234.dial.x adcop.org             TCP      1213 > 80 [ACK] Seq=7920982 Ack=3754522414 win=8576 L
733 59.266987 adcop.org             s340-modem2234.dial.x HTTP     Continuation
734 59.295336 s340-modem2234.dial.x adcop.org             TCP      1212 > 80 [ACK] Seq=7918369 Ack=3753703703 win=8576 L
735 59.319532 s340-modem2234.dial.x adcop.org             TCP      1213 > 80 [ACK] Seq=7920982 Ack=3754522623 win=8367 L
736 65.793511 msgr-ns16.msgr.hotmai s340-modem2234.dial.x TCP      1863 > 1174 [PSH, ACK] Seq=45837734 Ack=2152301 win=6
737 73.019948 adcop.org             s340-modem2234.dial.x TCP      80 > 1212 [FIN, ACK] Seq=3753703703 Ack=7918369 win=8
738 73.044517 s340-modem2234.dial.x adcop.org             TCP      1212 > 80 [ACK] Seq=7918369 Ack=3753703704 win=8576 L
739 74.995023 adcop.org             s340-modem2234.dial.x TCP      80 > 1213 [FIN, ACK] Seq=3754522623 Ack=7920982 win=8
740 75.021487 s340-modem2234.dial.x adcop.org             TCP      1213 > 80 [ACK] Seq=7920982 Ack=3754522624 win=8367 L
741 76.248033 s340-modem2234.dial.x adcop.org             TCP      1213 > 80 [RST] Seq=7920982 Ack=3754522624 win=0 Len=
742 76.270338 s340-modem2234.dial.x adcop.org             TCP      1212 > 80 [RST] Seq=7918369 Ack=3753702005 win=0 Len=
743 89.674738 msgr-ns16.msgr.hotmai s340-modem2234.dial.x TCP      1863 > 1174 [PSH, ACK] Seq=45837734 Ack=2152301 win=6
```

```
⊞ Frame 1 (77 on wire, 77 captured)
⊞ Ethernet II
⊞ Internet Protocol
⊞ User Datagram Protocol
⊞ Domain Name System (query)
```

```
0000  20 53 52 43 00 00 44 45  53 54 00 00 08 00 45 00    SRC..DE ST....E.
0010  00 3f c0 17 00 00 80 11  e4 98 c2 6d a8 ba c2 6d   .?...... ...m...m
0020  68 68 04 b2 00 35 00 2b  e2 b7 00 01 01 00 00 01   hh...5.+ ........
0030  00 00 00 00 00 00 03 77  77 77 09 68 61 7a 65 6c   .......w ww.hazel
0040  65 67 65 72 03 6e 65 74  00 00 01 00 01            eger.net .....
```

Fig. 3. Capturing activity

The picture above shows, partial capturing results of a session started after my login to the Internet and loading - from my homepage - of the site at **http://adcop.org.** This capture file is available as: ***PS-CC File 01.zip*** (241 kB uncompressed size), which will be mailed on request. (Note: Usernames and passwords have been altered into “username” and “password”).

8. Reading the Captured DATA

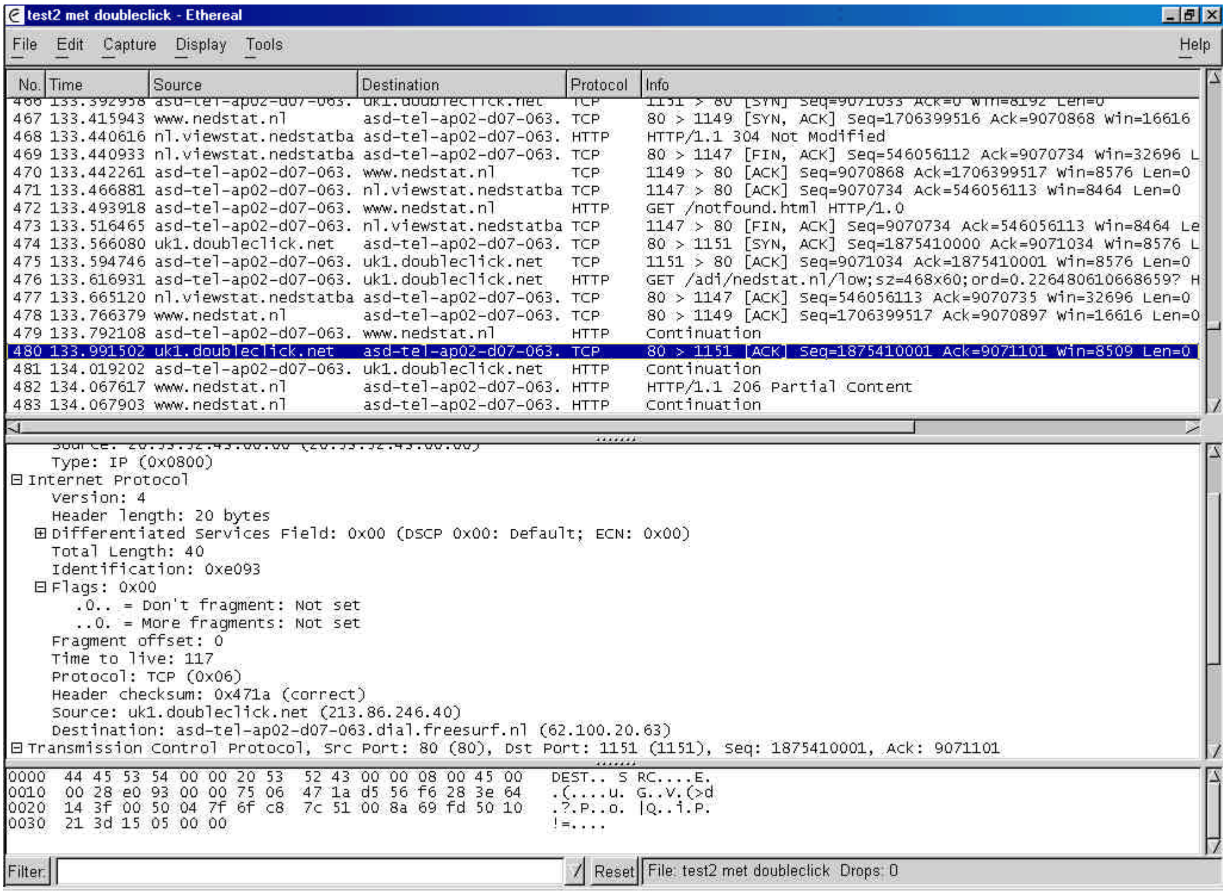
After reading the last chapter, you might have wondered why I didn’t skip the “normal situation” in the first place; why I didn’t went straight ahead for the captures which show “problems”… The simple explanation is: “How would you know the difference between a capture showing “normal” traffic and one showing “abnormal” traffic, when the “normal” version wasn’t shown before?”

To be able to understand what is happening we must dissect some frames from the PS-CC File 1 capture file. (If you don’t have the capture files, send an eMail message to info@hazeleger.net requesting them, they will be mailed to your eMail address in a ZIP-file).

PS-CC File 1

Now start the Ethereal program and load the PS-CC File 1 capture file. We will start our explanation at frame (packet) 001. Take care that section B is at least $\frac{1}{3}$ of the available screen, the upper part (section A) should be large enough to give room for at least 10 lines (frames), the remainder is for section C.

Note: *You can resize each section by hovering with your cursor above the divider, where the five dots are. Your cursor should now change into a ⇳. Now press the left mouse-button and drag the line (invisible) to its new position; release the mouse-button and the divider will be relocated at the desired position.*


test2 met doubleclick - Ethereal
File
Edit
Capture
Display
Tools
Help
Filter:
Reset
File: test2 met doubleclick Drops: 0

```
No. Time       Source               Destination          Protocol Info
466 133.392958 asd-tel-ap02-d07-063. uk1.doubleclick.net  TCP     1151 > 80 [SYN] Seq=9071033 Ack=0 Win=8192 Len=0
467 133.415943 www.nedstat.nl       asd-tel-ap02-d07-063. TCP     80 > 1149 [SYN, ACK] Seq=1706399516 Ack=9070868 Win=16616
468 133.440616 nl.viewstat.nedstatba asd-tel-ap02-d07-063. HTTP   HTTP/1.1 304 Not Modified
469 133.440933 nl.viewstat.nedstatba asd-tel-ap02-d07-063. TCP    80 > 1147 [FIN, ACK] Seq=546056112 Ack=9070734 Win=32696 L
470 133.442261 asd-tel-ap02-d07-063. www.nedstat.nl       TCP     1149 > 80 [ACK] Seq=9070868 Ack=1706399517 Win=8576 Len=0
471 133.466881 asd-tel-ap02-d07-063. nl.viewstat.nedstatba TCP    1147 > 80 [ACK] Seq=9070734 Ack=546056113 Win=8464 Len=0
472 133.493918 asd-tel-ap02-d07-063. www.nedstat.nl       HTTP    GET /notfound.html HTTP/1.0
473 133.516465 asd-tel-ap02-d07-063. nl.viewstat.nedstatba TCP    1147 > 80 [FIN, ACK] Seq=9070734 Ack=546056113 Win=8464 Le
474 133.566080 uk1.doubleclick.net  asd-tel-ap02-d07-063. TCP     80 > 1151 [SYN, ACK] Seq=1875410000 Ack=9071034 Win=8576 L
475 133.594746 asd-tel-ap02-d07-063. uk1.doubleclick.net  TCP     1151 > 80 [ACK] Seq=9071034 Ack=1875410001 Win=8576 Len=0
476 133.616931 asd-tel-ap02-d07-063. uk1.doubleclick.net  HTTP    GET /adi/nedstat.nl/low;sz=468x60;ord=0.226480610668659? H
477 133.665120 nl.viewstat.nedstatba asd-tel-ap02-d07-063. TCP    80 > 1147 [ACK] Seq=546056113 Ack=9070735 Win=32696 Len=0
478 133.766379 www.nedstat.nl       asd-tel-ap02-d07-063. TCP     80 > 1149 [ACK] Seq=1706399517 Ack=9070897 Win=16616 Len=0
479 133.792108 asd-tel-ap02-d07-063. www.nedstat.nl       HTTP    Continuation
480 133.991502 uk1.doubleclick.net  asd-tel-ap02-d07-063. TCP     80 > 1151 [ACK] Seq=1875410001 Ack=9071101 Win=8509 Len=0
481 134.019202 asd-tel-ap02-d07-063. uk1.doubleclick.net  HTTP    Continuation
482 134.067617 www.nedstat.nl       asd-tel-ap02-d07-063. HTTP    HTTP/1.1 206 Partial Content
483 134.067903 www.nedstat.nl       asd-tel-ap02-d07-063. HTTP    Continuation
```

```
    Source: 20:53:52:43:00:00 (20:53:52:43:00:00)
    Type: IP (0x0800)
⊟ Internet Protocol
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 40
    Identification: 0xe093
  ⊟ Flags: 0x00
      .0.. = Don't fragment: Not set
      ..0. = More fragments: Not set
    Fragment offset: 0
    Time to live: 117
    Protocol: TCP (0x06)
    Header checksum: 0x471a (correct)
    Source: uk1.doubleclick.net (213.86.246.40)
    Destination: asd-tel-ap02-d07-063.dial.freesurf.nl (62.100.20.63)
⊟ Transmission Control Protocol, Src Port: 80 (80), Dst Port: 1151 (1151), Seq: 1875410001, Ack: 9071101
```

```
0000  44 45 53 54 00 00 20 53  52 43 00 00 08 00 45 00   DEST.. S RC....E.
0010  00 28 e0 93 00 00 75 06  47 1a d5 56 f6 28 3e 64   .(....u. G..V.(>d
0020  14 3f 00 50 04 7f 6f c8  7c 51 00 8a 69 fd 50 10   .?.P..o. |Q..i.P.
0030  21 3d 15 05 00 00                                  !=....
```

Fig. 4. Capturing activity showing DoubleClick traffic

Now that the screen has been setup, we can have a look at what was happening during this session on the Internet. If not selected, select packet number 1. (Remember, I'd already logged onto the Internet when capturing started). After one has logged onto the Internet, the very next thing your browser will do is request the standard homepage. This is an address in the form http://www.xyz.com. As I have mentioned before, this address will be sent to the DNS for translation into an IP-address. Therefore the first thing "Internet Protocol" (IP) will do is to issue a request for a standard query to the DNS to ask what the address is for www.hazeleger.net.

We now move on to packet 2. As you can see, the question asked in packet 1 is answered here. Now move in section B. the screen contents in such a way that the item "Domain Name System (response)" moves upwards. Now click on the "+" sign, the information tree will unfold now. You'll see entries in it, most of them again with a "+" sign in front of them. The "Flags" item gives information about the kind of transaction and its status; the "Queries" item will show us the queries the system is answering (in this case: what is the IP-address of 'www.hazeleger.net'?); the "Answer" item in this tree shows us what the IP-address is: 195.11.243.19. "Authoritative Nameservers" tells the protocol where to look for the nameservers for this domain (hazeleger.net), two are given: a "primary" and a "secondary". Of course these names have to be translated into IP-addresses, so the item "Additional record" shows the translation for these nameservers.

Next, the protocol will issue an "ARP" (Address Resolution Protocol) request to whom it concerns. This must be explained. TCP uses IP, as well as physical addresses. Now if an IP-address is known, ARP will translate this into a physical address (also known as the "MAC"-address = Media Control Address). Those with a real network interface card would see here a long and unique number; for Windows 9x users with a DUN-connection this address always is 44:45:53:54:00:00 (data in hexadecimal representation, as you can see from the data in the data section of the screen).

In packet 4 we see, under the "Ethernet II" item, that a physical address has been returned (20:53:52:43:00:00). Now that the addresses are known (the computers have introduced themselves to each other) the sequence numbers must be synchronized; this is done by setting the SYN flag to 1 and sending this to the remote computer (where the page is to be downloaded from). This is done in packet 4. in the "Internet Protocol" item in the in-

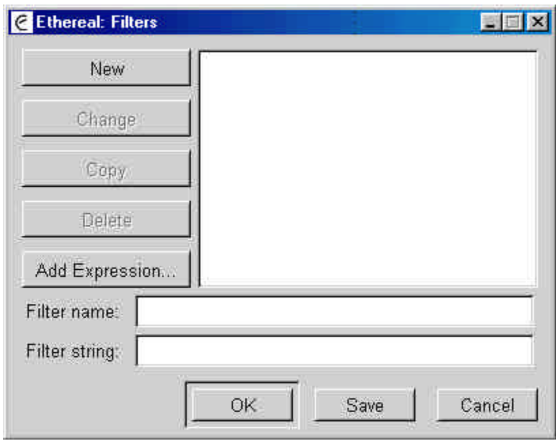


Fig. 5. Filter Set up (1st screen)

formation tree. Here the SYN flag is set (see under "Flags") and the sequence number is set by the local system (my computer) to a value of 7887652. In packet 5. one (1) is added and this value is sent back by the remote computer as acknowledgement together with the request to synchronize to this value. This is done in packet 6.

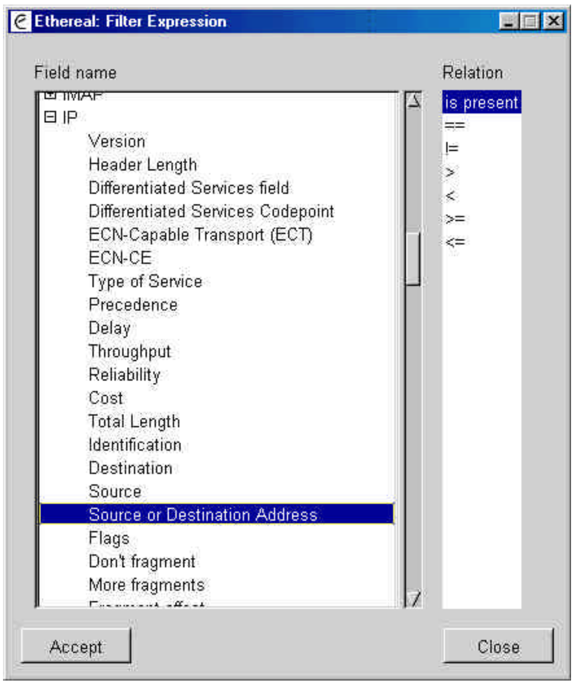


Fig. 6. Filter Set up (2nd screen)

and they now can communicate happily…

Finally in packet 7; they start with the things we, the users, could see: Getting the HTML page by issuing a HTTP (Hyper Text Transfer Protocol) command "GET". Now, besides the download of the page contents, a few other things happen; first of all we see a "detour" to "demon.co.uk" (packet 25 is the start of this). Now if you look at the next packets from this source you'll see that a "counter.cgi" script is executed from there; this was a counter that used to be on my page; the next "detour" brings us to "nl.nedstatbasic.net" (starting in packet 37), this is a "web statistics page"; but a bit of an intrusive one, clicking the icon on my page most certainly will deliver you a "doubleclick" cookie on your system when you have a look at my page statistics (in Dutch).

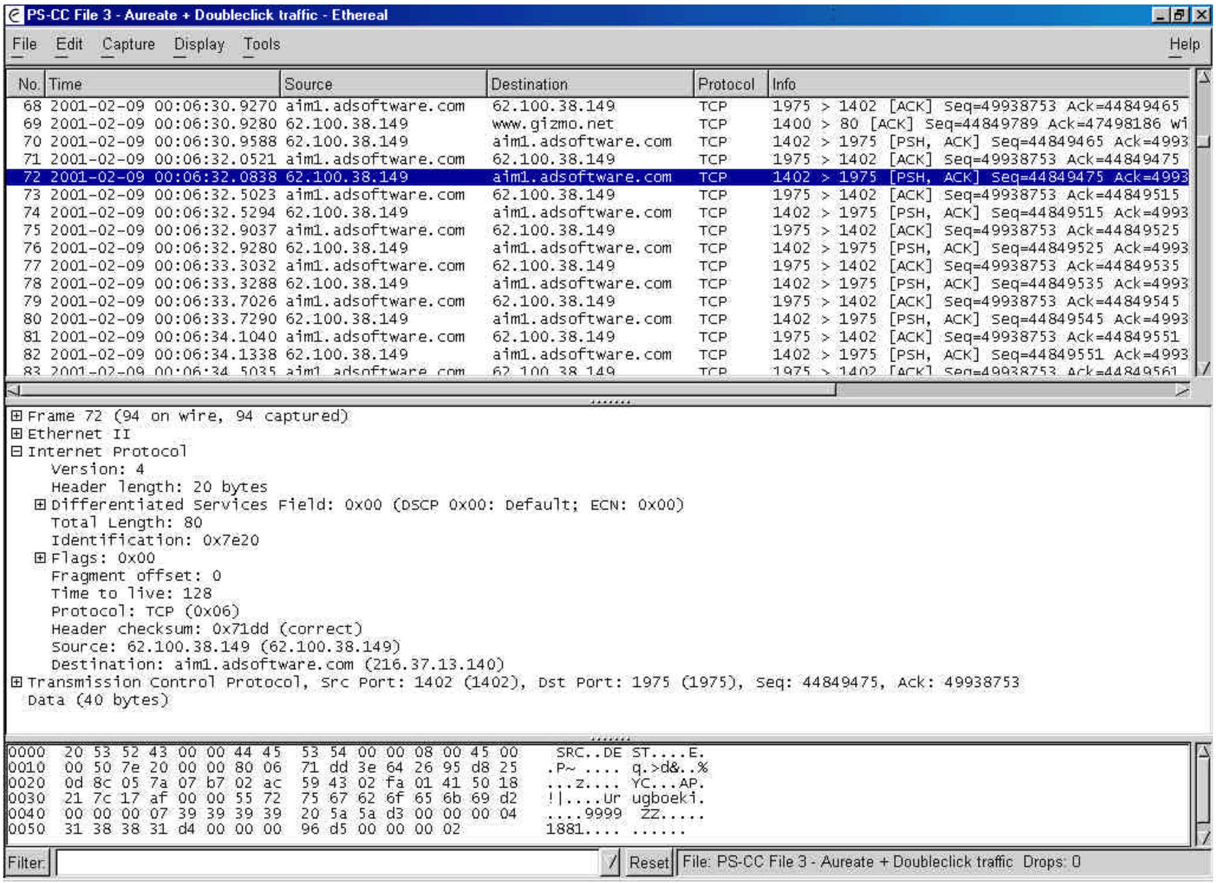
Now, I can hear you think: "OK, fine; but I haven't seen a single line of HTML, I've only seen a few pictures being loaded and counter and statistics scripts being executed so far… So where is the HTML?". The answer is easy. My page on startup is of course my own page. Therefore the page itself will be in my cache and when it hasn't changed it doesn't need to be reloaded, for some reason pictures are stored as links and reloaded each time you connect to a page.

So, in order to see HTML being loaded, we must watch the traffic from a site that wasn't in my cache at capture time, such as… adcop.org! Now select packet 83 (here the start of the actual downloading of the page is initiated) and {Right-click} on the item, select the first option from the menu that will be shown then ("Follow TCP Stream"). Now a new window is opened and the contents of that stream are shown and there is the HTML we were looking for!). This ends with package 742, the final packet in this capture.

PS – CC File 2.

Now that we have had a look at the normal traffic during a session on the Internet, we will proceed with a visit to a site that redirects to an ad banner-placing server.

The first difference with the previous capture file that you might notice is the presence of a lot of "Ethernet II"


PS-CC File 3 - Aureate + Doubleclick traffic - Ethereal
File
Edit
Capture
Display
Tools
Help
Filter:
Reset
File: PS-CC File 3 - Aureate + Doubleclick traffic Drops: 0

```
No. Time                    Source              Destination         Protocol Info
 68 2001-02-09 00:06:30.9270 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849465
 69 2001-02-09 00:06:30.9280 62.100.38.149       www.gizmo.net       TCP      1400 > 80 [ACK] Seq=44849789 Ack=47498186 Wi
 70 2001-02-09 00:06:30.9588 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849465 Ack=4993
 71 2001-02-09 00:06:32.0521 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849475
 72 2001-02-09 00:06:32.0838 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849475 Ack=4993
 73 2001-02-09 00:06:32.5023 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849515
 74 2001-02-09 00:06:32.5294 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849515 Ack=4993
 75 2001-02-09 00:06:32.9037 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849525
 76 2001-02-09 00:06:32.9280 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849525 Ack=4993
 77 2001-02-09 00:06:33.3032 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849535
 78 2001-02-09 00:06:33.3288 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849535 Ack=4993
 79 2001-02-09 00:06:33.7026 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849545
 80 2001-02-09 00:06:33.7290 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849545 Ack=4993
 81 2001-02-09 00:06:34.1040 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849551
 82 2001-02-09 00:06:34.1338 62.100.38.149       aim1.adsoftware.com TCP      1402 > 1975 [PSH, ACK] Seq=44849551 Ack=4993
 83 2001-02-09 00:06:34.5035 aim1.adsoftware.com 62.100.38.149       TCP      1975 > 1402 [ACK] Seq=49938753 Ack=44849561
```

```
⊞ Frame 72 (94 on wire, 94 captured)
⊞ Ethernet II
⊟ Internet Protocol
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 80
    Identification: 0x7e20
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 128
    Protocol: TCP (0x06)
    Header checksum: 0x71dd (correct)
    Source: 62.100.38.149 (62.100.38.149)
    Destination: aim1.adsoftware.com (216.37.13.140)
⊞ Transmission Control Protocol, Src Port: 1402 (1402), Dst Port: 1975 (1975), Seq: 44849475, Ack: 49938753
  Data (40 bytes)
```

```
0000  20 53 52 43 00 00 44 45  53 54 00 00 08 00 45 00    SRC..DE ST....E.
0010  00 50 7e 20 00 00 80 06  71 dd 3e 64 26 95 d8 25   .P~ .... q.>d&..%
0020  0d 8c 05 7a 07 b7 02 ac  59 43 02 fa 01 41 50 18   ...z.... YC...AP.
0030  21 7c 17 af 00 00 55 72  75 67 62 6f 65 6b 69 d2   !|....Ur ugboeki.
0040  00 00 00 07 39 39 39 39  20 5a 5a d3 00 00 00 04   ....9999  ZZ.....
0050  31 38 38 31 d4 00 00 00  96 d5 00 00 00 02         1881.... ......
```

Fig. 7. Capturing Aureate / Doubleclick traffic

packets. These 21 packets together form the "Logon sequence". In the first nine packets the remote system asks what my computer wants and my computer tells the remote system that it wants to logon to the Internet. In packet 10; my username and password are sent (there where it says "username" and "password" in the data section). Then the remote system checks if I'm allowed to connect to the Internet through that system and sends back the results of that check. Finally, in the final Ethernet II packets the TCP/IP connection is prepared.

In packet 22 my computer sends out an ARP to find out if IP-address 62.100.20.63 is in use. Since this doesn't seem to be the case, this address is assigned to be mine for this session. Then – I'm using the same browser as before – my own homepage is loaded and shown. Since my homepage is cached on my system (see explanation earlier in this chapter), you won't find any traces of it being downloaded. Instead you see that a connection to nl.nedstatbasic is being setup (a site statistics page in Dutch).

Note: *Since I knew this site was going to redirect me to "DoubleClick" from previous captures, it wasn't that difficult to get this data collected; in your case…. it may be somewhat more difficult and you could easily end up wading through megabytes of captured data before finding anything (or nothing); I was recently testing a program for its Internet behavior, the test took 43 minutes to perform and resulted in over 11,000 packets in a capture file over 3.6 Mb; quite a lot of data to examine. While capturing it is of the utmost importance that you log in some way where you have been during this session, after a longer period of time has passed this record could become very important!*

Fortunately, Ethereal can help us sort through all these packets for the ones we want. As written before, you can click on the headers of either "Source" or "Destination" to group all the sources or destination addresses together and then browse quickly through the data by using the slide-bar at the right of the screen, to undo a sort like this, just click on the "No." header and all should be back to normal. Another option is applying a filter.

To apply a filter we **must** know some basic details about what we want to see. Now, using the capture file above, I could apply a filter on "DoubleClick" to see how many times there has been an exchange between my system and their server during this session (and what data was transferred). To achieve this I would proceed as follows (assuming the capture file is loaded or an active capture is present):

1. I'd lookup the first appearance of this communication between my system and DoubleClick and select it.
2. Then I would expand the "Internet Protocol" header in section "B" (See page 8)
3. I would then see information like the information shown in ***figure 4***.
4. Now I would write down the IP-address (because I need it for the actual filtering)
5. Next, I would click on the [Filter] button in the lower left hand side of the Ethereal Main Screen. A screen as shown in figure 5 (next page) should appear.
6. Since Ethereal always starts this screen with an empty filter, we can give it a name in the "Filter name" field.
7. The next step I would take then would be to let Ethereal know what information I would like to filter. This is done by clicking on the [Add Expression] button. Now a screen with all the protocols appears.
8. Now, I would select the IP (Internet Protocol) and click on the "+" sign.
9. The screen in front of me should look like the one in fig. ##.
10. Next, I would find me the "Source or Destination Address" entry in that list and select it, the screen now should match the one shown in fig. ##1 exactly.
11. Now we will have to enter the actual filtering equation by clicking one of the equations in the right hand column of this screen, from top to bottom they are: "Is present" (obvious); "==" (Is exact equal to); "! =" (Not equal to); ">" (Greater than / Coming after); "<" (Smaller than / Coming before); ">=" (Equal to and greater than / coming after) and "<=" (Equal to and smaller than / coming before). Select one of these by clicking on it (I will use "==" here).
12. Now - at the right of the equation - a new field should appear, in this field you enter the IP-address you wrote down earlier.
13. Now click on the [Accept] button, the screen will close and you will be back in the "Filter setup" screen, where now a new button ([Apply]) should have appeared. Click on this button. The output now will be filtered and the filter expression will be visible in the field next to the [Filter] button.
14. To undo, there are two options, the first one is to click on the [Reset] button, which immediately will execute a filter setting of "" (meaning: No filtering); or you could click on the downwards pointing triangle at the right hand side of the filter expression field, select the empty filter and press [Enter] or [Return] to execute.

OK, now we know this, we can continue with our search through the data captured while loading "Nedstat".
Downloading NedStat's page is quite straightforward, no spectacular things happen during the first 334 packets; but then my system asks through ARP who owns the IP-address 213.86.246.40; a request that is answered by a DoubleClick server. DoubleClick is an advertising service and data mining company and is connected with several of the other "spyware" producing and distributing companies.

If you now move the selection bar to packet 336 and you {Right-Click} on the item and select "Follow TCP Stream" then you will see what this "detour" exactly does: It places a cookie on your system with a Time To Live of almost 30 YEARS!!! Also a banner ad for "www.reizenmarkt.nl" is shown and we're directed back to our point of origin.

Those who examine this “TCP Stream” screen more closely, might see a few strange things near User – Agent: e.g. SpaceBison/0.01??? Windows 67??? x??? Shonenknife??? The explanation for this is that I use Proxomitron on this connection and this is the kind of information Proxomitron provides about my system when asked by a “nosy” site!

OK, now I know that this “DoubleClick” places cookies, but a lot of sites do that and I think it won’t harm me… Yes? NO!!! Wrong!!! DoubleClick cookies are “persistant”; every time you visit a site that has this “DoubleClick” detour; or every time when you click on such a tempting “Click here” offer; information is added to the cookie and read whenever you’re on the DoubleClick server. By doing so, they can compile quite a nice profile about your surfing habits. In connection with the keyword registering capabilities of some of these nice “Browser plugins” they can store quite an impressive and detailed profile about you or the other members of your household… And remember: this happens without you knowledge or approval!!!

OK, we talked enough about “DoubleClick”, didn’t we? Well ehhm, Yes and No. As I pointed out earlier “DoubleClick” works in close cooperation with certain “spyware distributors”. One of them being Aureate (AKA Radiate). In the next paragraphs we will see what happens if one installs a spyware program and fills the installation questionnaire with data.

PS – CC 3.

As promised before, I installed an older and known spyware program, a download manager called Go!Zilla, filled the “questionnaire” with bogus information and accessed the Internet. The only program running at the time of the connection was the installed spyware program. Why? Because, if you want to do a controlled experiment change only one variable at a time (meaning in this case: run the spyware program when connected to the Internet for the first time, so no other programs will interfere by adding packets to the traffic); it also makes sense to do so if you want a controlled capture; if you won’t do so, you will end up with a garbage pile of unrelated packets compared to the examples I gave you. As expected, upon connecting a lot of traffic occurred. The first result of this traffic was a harmless update notice, which I escaped (meaning that no update took place...). Then even more traffic occurred and, as the resulting capture file later showed, then the data I entered into the “questionnaire” was transferred to the advertisement server. (Note: In this file the login information has been changed into “username” and “password”).

Analyzing the contents of this capture file is a bit more difficult than the previous ones, though not as difficult as reading Chinese without learning it first. Let’s start at the beginning of the file. The first 24 packets (Those with “Ethernet II” in the Info column) are known from the last capture file. They represent the logon sequence. The username and password are transmitted to the remote computer in packet 12. Then this information is checked on the logon server and the result is transferred back to my system. Then the normal setup of TCP/IP continues and my system gets IP-address 62.100.38.149 assigned to it.

After the normal setup of TCP/IP, in packet 35 the DNS information about the Aureate server (my connection at that time) is requested from the Freesurf DNS; in packet 36 the DNS info about another site, www.gizmo.net is requested; www.gizmo.net is the Go!Zilla homepage address. In packet 38 the answer to the DNS-query for the Aureate server is given. In packet 45 the Aureate server signals a PSH (Push); this means that TCP must “Push” all data to the receiving application (in this case this is the remainder of the Go!Zilla installation program).

After the usual synchronization finally, in packet 48 the transfer of information between www.gizmo.net and my system starts. My system sends a standard header (generated by Go!Zilla) which contains also the version of the program. In packets 49 and 50 the Aureate server sends it SYN reply, which in turn is acknowledged by my system. In packets 51 and 52 my system and the Gizmo server again synchronize. The “Aureate “Push” is repeated in packet 53 (at this point an active new ad is already shown). In packet 56 the server at Gizmo.net starts uploading the update notification to my system (you check this out by {Right-Clicking} this item and select “Follow TCP Stream; don’t forget to hit the [Reset] button if you did check this out!). This communication is broken into several packets which we see as HTTP” in the “Protocol” and “Continuation” in the “Info” columns. I will skip this communication in the remainder of this text.

In packet 72 the “installation program” starts transferring the data I entered earlier – during installation - in the “questionnaire”. You might not recognize it as such, but it is the data I entered. “Urugboek, this is a new (non existing) “province” of The Netherlands; “9999 ZZ” is a faked postcode (ZIP-code); “1881” should have been my ‘Year of birth’ (Yes, I’m a ‘hale and hearty’ old fellow!). If you would {Right-Click} on the item and select “Follow TCP Stream” you would see that also some numeric data was transferred; these numbers represent some choices I made from drop down menus; at the time when I filled the questionnaire.

The screenshot of the capture file shows packet 72 here, showing the data, as described above, being transferred.

While the communication with the Aureate server continues, the server at Gizmo.net signals in packet 108 a TCP “FIN” flag. This flag is used to signal that no data from that system or server has to be transferred and that this connection may be closed. From this point on we’re at the “mercy” of the Aureate server… After quite a bunch of “Pushes” and “ACK’s” a change of scene is announced in packet 122, where we are redirected yet another time. This time the detour leads us to… DoubleClick. Yes indeed, the same one we saw earlier. After yet another 17 packets of pushes and acknowledgements my system queries the DNS for that server (ad. doubleclick.net) which query is answered in packet 141 (have a look at the list of Nameservers in this response…). During this communication yet another of DoubleClick’s cookies (with a Time To Live of more than 29 YEARS) is set. During the communication that follows also the ads are refreshed.

Note: *Although my name wasn’t transferred to the Aureate and / or Doubleclick server, the other information I entered was, including the postal or ZIP-code. Now I don’t know all systems used by the national postal organizations in the world, but I* ***do*** *know that in some countries giving the postal code is as much the same as giving your name; thus allowing the Aureate’s and Doubleclick’s to link the profile they make to you and make it less anonym than you might think and they would let you to believe! So:* ***be careful when entering information in questionnaires like this!!!***

After this the traffic didn’t stop, the first set of “fresh” ads was then downloaded to my system. During this time I did absolutely nothing but watch what data came in and got out. I acted this way because I knew that this was a spyware program; normally you, when you installed this program (a download-manager), would be experimenting with the program, generating your own traffic so this extra traffic wouldn’t be noticed. At the same time however this would be the beginning of a very fruitful cooperation… fruitful for them that is, since you would provide them with varied information about your entire household, which then could be sold to marketing companies.

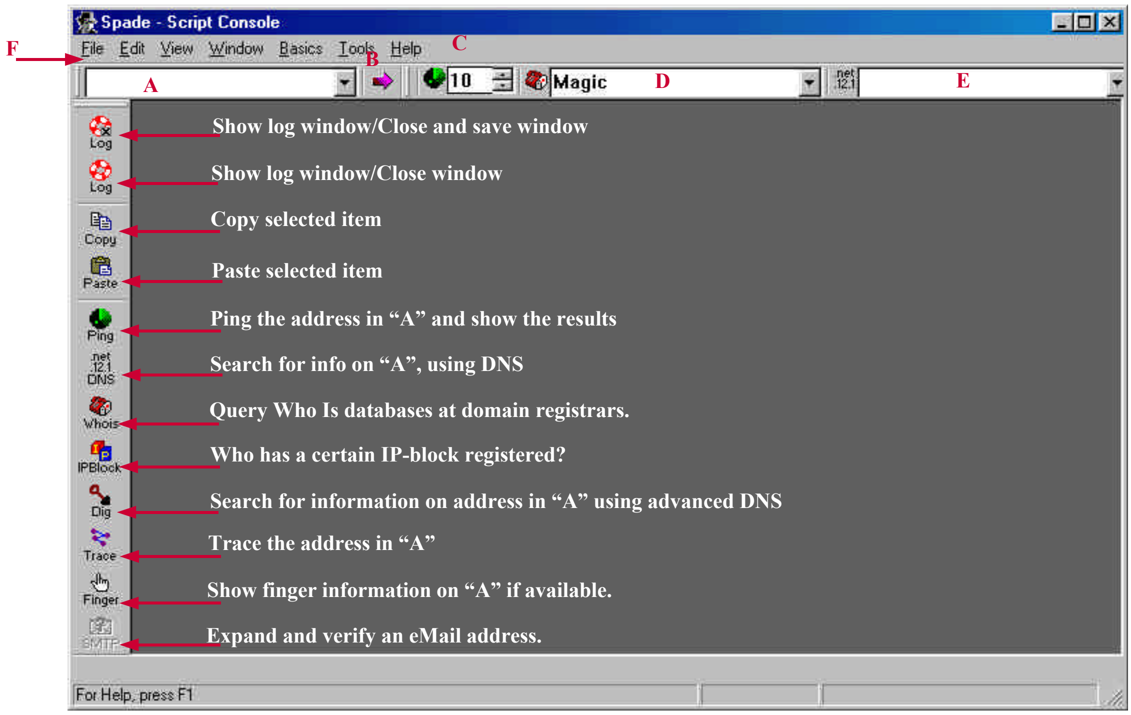
For now, we have enough information to continue with our next task, getting information from either a servername or an IP-address; we will achieve this by using the Windows 32 bit program “Sam Spade”.

9. Checking out destination IP-addresses (Sam Spade)

So far we have used Ethereal's name resolution to retrieve information about these advertising servers, the companies that own them etc. Now we will use another program called Sam Spade to learn more. (**Note:** *Sam Spade is free and widely used for tracking and tracing websites and servers. It is a good example of what to expect in such a program and how to use them, but in this chapter it's just that: An example of what you may expect from a program like this!)*

Sam Spade originally started as a web based secure ping and trace utility used for fighting "Spam". Later a browser plug-in was developed to access the web-based service. Today Sam Spade is a normal 32 bits Windows application. Since "ping" (ICMP) and "trace" are commands that originate normally from your computer, the resulting packets would contain your IP-address; exposing it to the computer you tried to ping or trace. The use of Sam Spade avoids this, since the originating IP-address is that of Sam Spade and not yours.

Sam Spade, is available at http://samspade.org/ssw/dl.html: it installs itself, creates a shortcut in the [Start] "Programs" folder and then is ready to use. Now connect to the Internet and start Sam Spade.



Show log window/Close and save window

Show log window/Close window

Copy selected item

Paste selected item

Ping the address in "A" and show the results

Search for info on "A", using DNS

Query Who Is databases at domain registrars.

Who has a certain IP-block registered?

Search for information on address in "A" using advanced DNS

Trace the address in "A"

Show finger information on "A" if available.

Expand and verify an eMail address.

Fig. 8. Sam Spade screenshot.

Now, let's go back to the previous page. In frame 82 "aim1.adsoftware.com" is mentioned as the destination address. Now let's try and get some more information on this server. Here is what I did:

After executing "Sam Spade" I entered the "aim1.adsoftware.com" address in the left hand field and the first assignment I gave Sam was to perform a DNS search. Though useful, the information returned here was limited: It consisted of the name of that server (which we already knew) and its IP-address (which we could have found too, if we had opened the Internet Protocol information tree in Ethereal). (**Note:** *If this DNS search won't work for you, look up the IP-address of your ISP's primary name server DNS* (for an easy way to find this information, please have a look at the information in Chapter 13: "Appendix B. How to determine your IP-address"), *write it down (if necessary), then go to Sam Spades options screen (Edit | Options) and fill in this information in the first large field on the "Basics" tab. Then this function should work.*

The second assignment for "Sam Spade" was a "Who Is". Just like a real private eye (Wasn't Sam Spade the name of a private eye Humphrey Bogart played in one of his movies?) the program searches for the domain name in several databases belonging to domain registrars (if “Magic” was selected in the drop-down field, marked “D” in the screen shot on the previous page) . From this search I got more information, telling me that the domain belongs to “Radiate Inc.” plus the address where they can be contacted, their billing address at “Aureate Development Inc.” and what the names and IP-addresses of their name servers (DNS) are.

But I wasn’t finished yet. Next, I assigned Sam to look for information on the IP-block this IP-address belongs to. By doing so, you can find out who is the provider of a specific IP-address; this information may be vital when complaining to an ISP, e.g. in case of Spamming or port probing. This information was returned instantly.

Then I let Sam Spade do some “digging”, this returned some information about the servers: unfortunately, this provided no new information not in an earlier report.

Finally, I tried to extract the “finger” information from both the aim1 and the nameservers at aureate. It seems they’ve switched off this feature (on the server-side), so no information but a mere “failed, couldn't connect to host” was returned.

All these actions took about five minutes and what did it bring us? First it showed us the name, the address and the contact information of the registered user of that server, a well-known developer and distributor of spyware modules. Secondly, we found out who their ISP is. The remainder of the information found was either a confirmation of information found earlier, or no information at all. Yet, the result is quite remarkable.

10. I have found something odd, now what?

So, you **have** discovered something going on your system you don’t like? What now?

Well first of all, the offending program(s) should be identified. On the Internet several lists of spyware or suspected spyware programs exist, you could check the programs you’ve installed against one (or more) of these lists. Please refer to the “Useful links” section.

When the offending program have been found, the next question is: “Do you want to keep it?” If so, you could leave it the way it is, knowing what is going on; you could consider buying the program (and even then check this program out), or look for an alternative that won’t “spy” on you.

If you decide to remove the program from your computer, then do so either by using the “Uninstall” program that came with the program or by using Window’s built in uninstaller that can be reached through [Start] | “Configuration Panel” | “Software.

The next thing that should be done is to run a spyware checker since a number of spyware modules may remain active even after the “host program” was removed from the system. Any occurrences of spyware should be removed then (if you find more, different, modules here… be careful you could render another program unusable…). The best way to check for and, eventually remove, the remaining spyware is by using “Ad-aware” (see … Useful links).

Now you may want to contact the author of the program and tell him / her that you removed his / her program from your system and why. It could be possible that, when he author gets enough of these kind of messages, he or she would reconsider the way this program is distributed (cases are known where this happened). The author also could get you involved in some fruitless discussion about the “paranoid” behavior of the Internet community towards “Adware” programs; it’s a risk you would have to take!!

11. More information on “Packet Sniffing”

In the previous chapters I gave a brief introduction into “Packet Sniffing”. The knowledge provided here should be sufficient to start the search for unwanted Internet traffic. For those who want to know more and want to get the information from the Internet, below are some links on the topics “Packet Sniffing”, “Intrusion detection by using a packet sniffer”, “TCP/IP and other protocols” and “Packet Sniffers” are listed.

Clicking on a **blue** link will start your browser and bring you to this page directly.

http://www.decodes.com	➔ Resource for Network Protocol Analysis
http://www.protocols.com	➔ The name says it all!
http://www.radcom-inc.com	➔ Protocols
http://www.robertgraham.com	➔ FAQ on sniffing and intrusion detection
http://www.acacia-net.com/Clarinet/protocol.htm	➔ Online book on Protocols
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ppp.htm	➔ Info on PPP
http://www.cyberport.com/~tangent/programming/winsock/resources/debugging.html	➔ Comparison of sniffers
http://www.3com.com/nsc/501302.html	➔ Understanding IP Addressing
http://www.charm.net/ppp.html	➔ Lots of info on TCP/IP and PPP
http://www.doc.ic.ac.uk/~ih/doc/pc_conn/	➔ Lots of info on IP and Winsock
http://packetstorm.securify.com/sniffers/	➔ Information and download of sniffers

12. Appendix A. Useful links

A. Information about spyware

Information on the subject of spyware can be found on one of the following websites (if you have a site that gives information on the subject and that isn't related in any way with the developers or distributors of spyware, please mail me the address: it will be listed in the next release then. My eMail address can be found **here**.

Christophe Wolfs page on spyware at **http://home.tvd.be/ws36178/security.html**
Easy Life Help pages at **http://www.eazylife.net/HELP/spies.htm#update**
John Fitzsimons page at **http://www.alphalink.com.au/~johnf/spyware.html**
My own page (for the ACF spyware list) at **http://www.hazeleger.net/**
Spychecker on line reference database at **http://spychecker.com/**
Steve Gibson's homepage on spyware and the like at **http://grc.com/optout.htm**
VNU Belgium on spyware (French) at **http://www.vnunet.be/compmag_fr/index_spyware.asp**
VNU Belgium on spyware (Dutch/Flemish) at **http://www.vnunet.be/compmag/index_spyware.asp**
Voice of the public on spyware at **http://www.voiceofthepublic.com/IsYourFreewareSpyware.html**

B. Detection and removal of spyware

At present only one spyware removal and detection tool exists. It is called **Ad-Aware** and is available at: **http://www.lavasoft.de/aaw/index.html** for free. This program makes use of "signatures", just as an Anti-Virus program does, so you should visit this site at regular intervals to update your signature file.

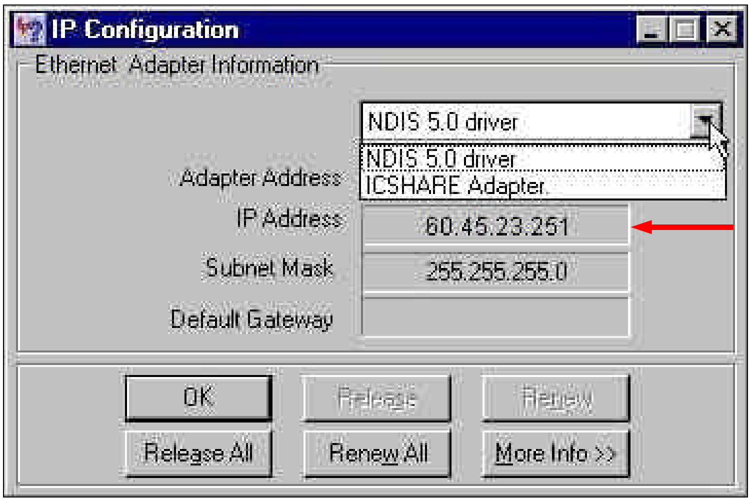
13. Appendix B. How to determine your IP-address and DNS

When packet sniffing, it may be useful to know one's own IP-address on beforehand. Luckily the Windows 95 and Windows 98 Operating Systems have a small, yet undocumented, utility program to just do so.

This program is called: "**Winipcfg**"

(In the next paragraphs I assume you're connected to the Internet)

To start this program, click on [Start], "Run" and then type in "winipcfg.exe" (without the quotes). Then either click on [OK] or press [Enter] or [Return]. After the program has been started you should a screen like the one below:



IP Configuration

Ethernet Adapter Information

NDIS 5.0 driver

NDIS 5.0 driver
ICSHARE Adapter

Adapter Address

IP Address 60.45.23.251

Subnet Mask 255.255.255.0

Default Gateway

Fig. 9. Winipcfg (basic information)

Now; if there isn't anything shown, or in the white field at the top your adapter type isn't mentioned; click on the downwards pointing triangle at the right of the adapter type (where the white "cursor" arrow is pointing to) and select the right type (for most DUN connections this would be the "PPP Adapter).

Now you should see your IP-address at the place where the red arrow is pointing at.

Another piece of information that's been given here is the **Subnet** Mask. The Subnet Mask is a 32 bits pattern that is used to determine what part of the IP-address is the Host part and what part is the Network part. The Subnet Mask is used in two computations, one computation generates a result with the IP address of the "target computer", the other generates a result with the IP-address of the "source computer". If the results are equal, then the IP-protocol "knows" that both computers are in the same Network and that they can communicate directly. If the results don't match, then the IP-protocol knows the computers are located in different segments of the network - or even an different network - and that the packets will have to be send to a router to be able to leave this network.

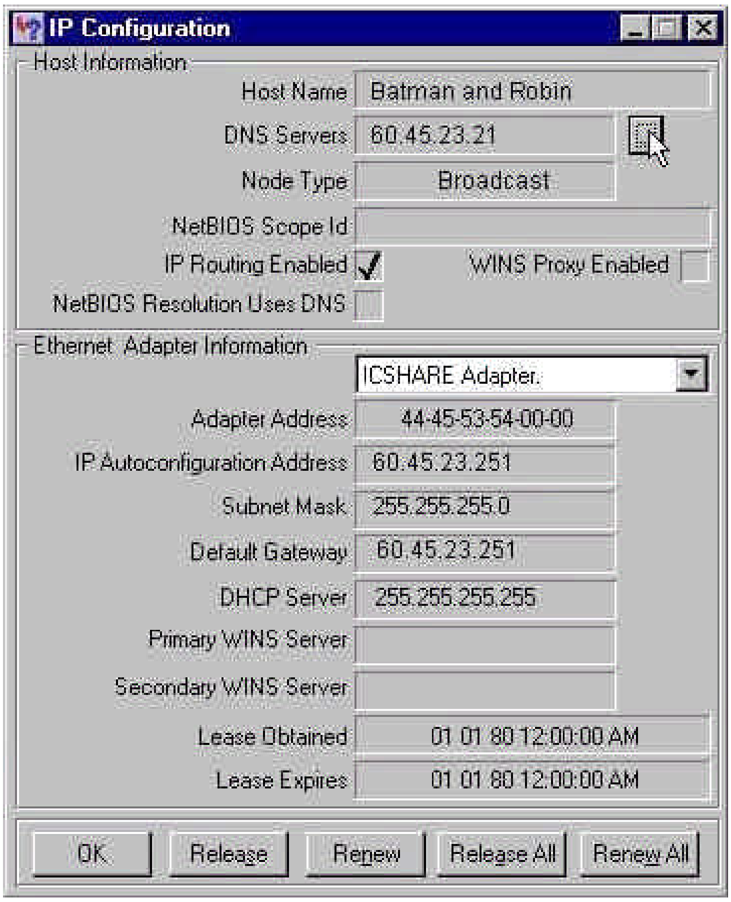
In a Network, the Subnet Mask also gives information about the "Class" the Network belongs to, without explanation, these classes are shown on the next page in "Table 1. Network Classes."

If you find yourself interested in TCP/IP, Subnet Masks and the way these provide us with fast means of transportation of our (Inter)Net traffic, the very last chapter (**14. Appendix C. Bibliography**) will provide the names of some really good books on the matter.

Of course you've noticed the [**More Info** >>] button in the lower right hand side of the illustration. Now, if you want to see the more information about the settings for this connection, click on [**More Info** >>] and the screen as shown in Figure 10 will be shown. This screen shows you information about:

Class	Subnet Mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

Table 1. Classes and Subnet Masks



IP Configuration

Host Information

Host Name: Batman and Robin

DNS Servers: 60.45.23.21

Node Type: Broadcast

NetBIOS Scope Id:

IP Routing Enabled ☑ WINS Proxy Enabled □

NetBIOS Resolution Uses DNS □

Ethernet Adapter Information

ICSHARE Adapter.

Adapter Address: 44-45-53-54-00-00

IP Autoconfiguration Address: 60.45.23.251

Subnet Mask: 255.255.255.0

Default Gateway: 60.45.23.251

DHCP Server: 255.255.255.255

Primary WINS Server:

Secondary WINS Server:

Lease Obtained: 01 01 80 12:00:00 AM

Lease Expires: 01 01 80 12:00:00 AM

Fig. 10 Winipcgf ([More Info >>] pressed); the pointer points at the button to change the DNS used.

- Your computer's name on the network.
- What DNS is currently selected as the active one (if you need your DNS for some program, e.g. Sam Spade, this is the place to find it).
- Whether IP routing is enabled or disabled (in Fig. 10 it's enabled, but it could be disabled on your connection).
- Whether WINS proxy is enabled or disabled (in Fig. 10 it's disabled, but it could be enabled on your connection).
- Whether NetBIOS Resolution uses DNS. In the above figure it's disabled and since NetBIOS is dangerous (from a safety perspective) to use on the Internet, it **should** be disabled!
- Again the information, we saw earlier in in the screen as shown in Figure 9, is shown here too.
- Default Gateway, this is an optional 32-bits address that will be used to tell the IP-protocol the address of a router. Packets that are intended to be sent to another Network, will be send to this address and will be, from there, routed the right way to their destination.

- DHCP Server. (Dynamic Host Control Protocol Server). This protocol enables computers to get their TCP/IP settings assigned automatically. The computer which receives the settings is called the “client”, the computer which provides the TCP/IP settings is called the “DHCP-Server”.
- WINS Servers. WINS (Windows Internet Name Service) can be installed on Windows NT-systems. It was developed to be a solution for certain limitations provided by other Name Services (e.g. Lan Manager Host - in short LMHost -).

You may close any of these screens during a connection by clicking on either the [OK] button or the “X” mark in the right hand corner of the screen, this won’t close the connection in progress.

14. Appendix C: Bibliography

For those who want to have detailed information about the TCP/IP protocol, below a list of good books on the subject is given:

Title	Author	ISBN
TCP/IP in 24 Hours	Hayden, Matt	0672312484
TCP/IP Clearly explained	Loshin, Peter	0124558267
TCP/IP for Dummies	Leiden / Wilensky	0764507265
TCP/IP Illustrated vol. 1: The Protocols	Stevens, W. Richard	0201633469
TCP/IP Illustrated vol. 2: The Implementation	Wright / Stevens	020163354X
TCP/IP Illustrated vol. 3: TCP for transactions, HTTP, NNTP and the UNIX Domain Protocols	Stevens, W. Richard	0201634953

Table 2. Books on the subject of TCP/IP

Of course, this is just a small selection of books available on the subject. A book not being mentioned here can be just as good as the ones I've list here. If you have information about other, good, books on the subject; feel free to eMail me the information I need to include that book on this list in a future release.

This page was intentionally left blank