

[Lab] 網路封包分析 器(Network Packet Analyzer)

異質多網多媒體服務



國立臺北科技大學電子工程系
授課教師：李昭賢 副教授
電子郵件：chlee@ntut.edu.tw
校內分機：2288





學習目標 Outline

■ Packet Analyzer / Sniffer

- Wireshark

■ HTTP

- Chrome Browser
- Postman

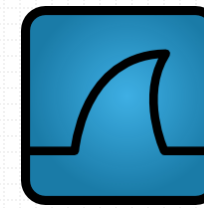


網路封包分析器(Network Packet Analyzer)

- 封包分析(Analysis)/監聽(Sniffing)
 - 當資料流經網路時，將其擷取並解譯。
 - 呈現特定封包的結構與內容。
 - 掌握網路資源使用情況。
 - 何人、何事佔用頻寬
 - 識別網路攻擊或惡意行為
- 封包分析器(Analyzer)/監聽器(Sniffer)
 - 封包擷取、流量統計之工具
 - TCPdump、Wireshark (Ethereal)、MS Message Analyzer (Network Monitor)
 - [Library/API] libpcap (Linux), WinPcap, Npcap (MS Windows)



Wireshark (Ethereal)



- 免費開源的網路封包分析軟體
 - 1998年Gerald Combs開發Ethereal
 - 2006年更名為Wireshark
- 目前全世界最廣泛的網路封包分析軟體
 - 官方網站：
<https://www.wireshark.org/>
 - 官方說明：
<https://www.wireshark.org/docs/>



NEWS Get Acquainted ▾ Get Help ▾ Develop ▾

Wireshark Training

Wireshark University

Co-founded by Laura Chappell, inspirational instructor, consultant, and Wireshark expert, provides training, Network Analyst Certification, and resources for all levels of Wireshark users.

Visit www.wiresharktraining.com.

Wireshark Network Analysis

The Official Wireshark Certified Network Analyst Study Guide is now available. Get your copy today!

Wireshark Certified Network Analyst: Official Exam Prep Guide

Want to become a Wireshark Certified Network Analyst? This book gives you 300 practice questions along with an accompanying practice CD.



User Documentation

User's Guide

The Wireshark User's Guide is available in several formats:

Videos and Presentations

Sharkfest Presentations

Sharkfest features presentations from a variety of knowledgeable, informative speakers.

- Sharkfest '15
- Sharkfest '14
- Sharkfest '13
- Sharkfest '12
- Sharkfest '11
- Sharkfest '10
- Sharkfest '09
- Sharkfest '08

Videos

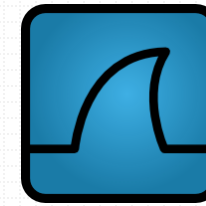


Hands on with Wireshark

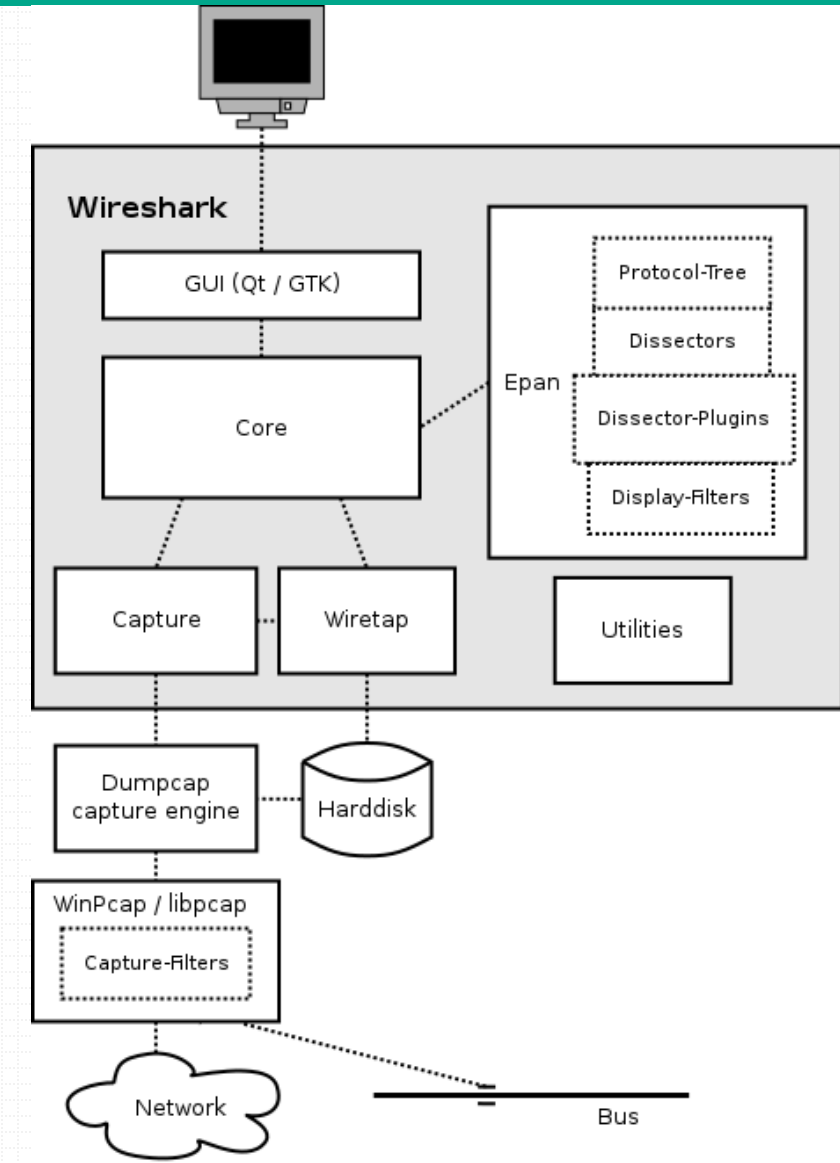
Hansang Bae shows you tips and tricks used by insiders and veterans. First in a series.

11m 43s

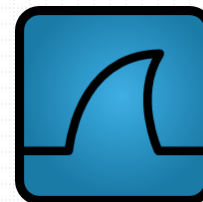
Wireshark (Ethereal)



- GUI
 - Handling of all user input/output.
- Core
 - Main "glue code" that holds the other blocks together.
- Epan
 - Enhanced Packet Analyzer
- Wiretap
 - The wiretap library is used to read and write capture files in libpcap, pcapng, and many other file formats.
- Capture
 - The interface with the capture engine.



Wireshark (Ethereal)



Main Menu

Display Filter

Packet List

Packet Details

Packet Bytes

tv-netflix-problems-2011-07-06.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/> Expression...

No.	Time	Source	Destination	Protocol	Length	Info
343	65.142415	192.168.0.21	174.129.249.228	TCP	66	40555 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519346 TSecr=551811827
344	65.142715	192.168.0.21	174.129.249.228	HTTP	253	GET /clients/netflix/flash/application.swf?flash_version=flash_lite_2.1&v=1.5&n...
345	65.230738	174.129.249.228	192.168.0.21	TCP	66	80 → 40555 [ACK] Seq=1 Ack=188 Win=6864 Len=0 TSval=551811850 TSecr=491519347
346	65.240742	174.129.249.228	192.168.0.21	HTTP	828	HTTP/1.1 302 Moved Temporarily
347	65.241592	192.168.0.21	174.129.249.228	TCP	66	40555 → 80 [ACK] Seq=188 Ack=763 Win=7424 Len=0 TSval=491519446 TSecr=551811852
348	65.242532	192.168.0.21	192.168.0.1	DNS	77	Standard query 0x2188 A cdn-0.nflximg.com
349	65.276870	192.168.0.1	192.168.0.21	DNS	489	Standard query response 0x2188 A cdn-0.nflximg.com CNAME images.netflix.com.edge...
350	65.277992	192.168.0.21	63.80.242.48	TCP	74	37063 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=491519482 TSecr=...
351	65.297757	63.80.242.48	192.168.0.21	TCP	74	80 → 37063 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=3295...
352	65.298396	192.168.0.21	63.80.242.48	TCP	66	37063 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=491519502 TSecr=3295534130
353	65.298687	192.168.0.21	63.80.242.48	HTTP	153	GET /us/nrd/clients/flash/814540.bun HTTP/1.1
354	65.318730	63.80.242.48	192.168.0.21	TCP	66	80 → 37063 [ACK] Seq=1 Ack=88 Win=5792 Len=0 TSval=3295534151 TSecr=491519503
355	65.321733	63.80.242.48	192.168.0.21	TCP	1514	[TCP segment of a reassembled PDU]

> Frame 349: 489 bytes on wire (3912 bits), 489 bytes captured (3912 bits)

> Ethernet II, Src: Globalsc_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio_14:8a:e1 (00:19:9d:14:8a:e1)

> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21

> User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)

▼ Domain Name System (response)

 [Request In: 348]

 [Time: 0.034338000 seconds]

 Transaction ID: 0x2188

 > Flags: 0x8180 Standard query response, No error

 Questions: 1

 Answer RRs: 4

 Authority RRs: 9

 Additional RRs: 9

 ▼ Queries

 > cdn-0.nflximg.com: type A, class IN

 > Answers

 > Authoritative nameservers

0020 00 15 00 35 84 f4 01 c7 83 3f 21 88 81 80 00 01 ...5.... ?[.....

0030 00 04 00 09 00 09 05 63 64 6e 2d 30 07 6e 66 6cc dn-0.nfl

0040 78 69 6d 67 03 63 6f 6d 00 00 01 00 01 c0 0c 00 ximg.com

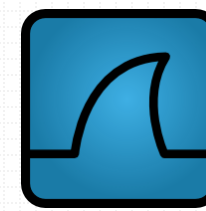
0050 05 00 01 00 00 05 29 00 22 06 69 6d 61 67 65 73). ".images

0060 07 6e 65 74 66 6c 69 78 03 63 6f 6d 09 65 64 67 .netflix .com.edg

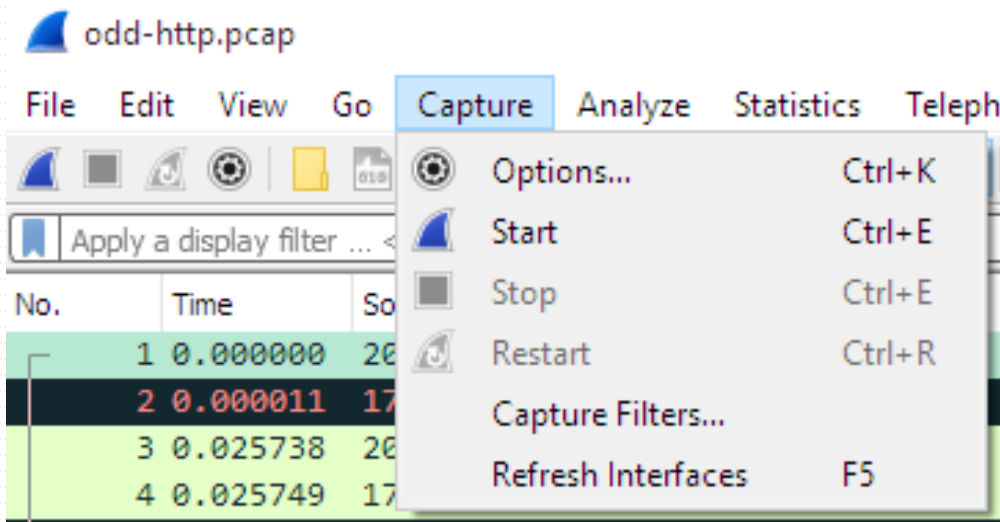
0070 65 73 75 69 74 65 03 6e 65 74 00 c0 2f 00 05 00 esuite.n et../...

Identification of transaction (dns.id), 2 bytes Packets: 10299 · Displayed: 10299 (100.0%) · Load time: 0:0.182 Profile: Default

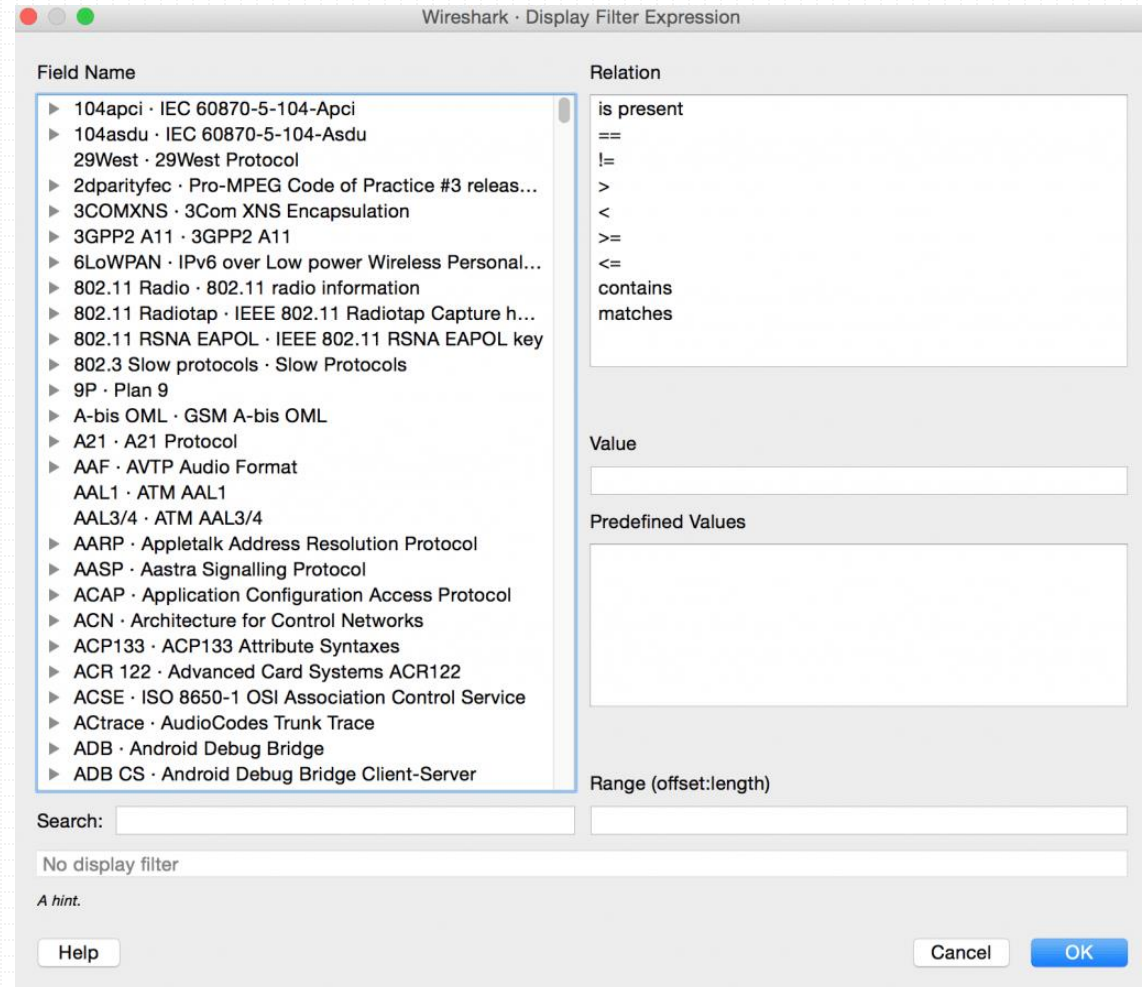
Wireshark (Ethereal)



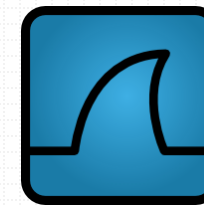
- Capture → Start



- Display Filter



Wireshark (Ethereal)



• Packet List

Apply a display filter ... <%%/>									Expression...	+
No.	Time	Source	Destination	Protocol	Length	Ethernet	Info			
1	2004-05-13 03:17:07.311224	145.254.160.237	65.208.228.223	TCP	62	Yes	3372 → 80 [SYN] Seq=0 Win=8760 Len=0 MSS=146...			
2	2004-05-13 03:17:08.222534	65.208.228.223	145.254.160.237	TCP	62	Yes	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 Le...			
3	2004-05-13 03:17:08.222534	145.254.160.237	65.208.228.223	TCP	54	Yes	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0			
4	2004-05-13 03:17:08.222534	145.254.160.237	65.208.228.223	HTTP	533	Yes	GET /download.html HTTP/1.1			
5	2004-05-13 03:17:08.783340	65.208.228.223	145.254.160.237	TCP	54	Yes	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len=0			
6	2004-05-13 03:17:08.993643	65.208.228.223	145.254.160.237	TCP	1434	Yes	[TCP segment of a reassembled PDU]			
7	2004-05-13 03:17:09.123830	145.254.160.237	65.208.228.223	TCP	54	Yes	3372 → 80 [ACK] Seq=480 Ack=1381 Win=9660 Le...			
8	2004-05-13 03:17:09.123830	65.208.228.223	145.254.160.237	TCP	1434	Yes	[TCP segment of a reassembled PDU]			
9	2004-05-13 03:17:09.324118	145.254.160.237	65.208.228.223	TCP	54	Yes	3372 → 80 [ACK] Seq=480 Ack=2761 Win=9660 Le...			
10	2004-05-13 03:17:09.754737	65.208.228.223	145.254.160.237	TCP	1434	Yes	[TCP segment of a reassembled PDU]			

• Packet Details

▶ Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)

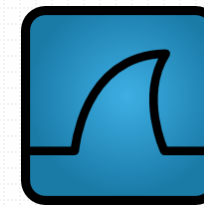
▶ Ethernet II, Src: Superlan_00:00:00 (00:00:01:00:00:00), Dst: fe:ff:20:00:01:00 (fe:ff:20:00:01:00)

▶ Internet Protocol Version 4, Src: 145.254.160.237, Dst: 65.208.228.223

▼ Transmission Control Protocol, Src Port: 3372, Dst Port: 80, Seq: 0, Len: 0

- Source Port: 3372
- Destination Port: 80
- [Stream index: 0]
- [TCP Segment Len: 0]
- Sequence number: 0 (relative sequence number)
- Acknowledgment number: 0
- Header Length: 28 bytes
- ▶ Flags: 0x002 (SYN)
- Window size value: 8760
- [Calculated window size: 8760]
- Checksum: 0xc30c [unverified]
- [Checksum Status: Unverified]

Wireshark (Ethereal)



- Packet Bytes

Apply a display filter ... <=>/>

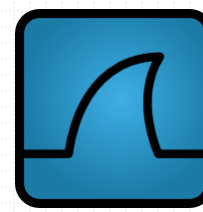
No.	Time	Source	Destination	Protocol	Length	Ethernet	Info
1	2004-05-13 03:17:07.311224	145.254.160.237	65.208.228.223	TCP	62	Yes	3372 → 80 [SYN] Seq=0 Win=8760 Len=0 MSS=1...
2	2004-05-13 03:17:08.222534	65.208.228.223	145.254.160.237	TCP	62	Yes	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 ...
3	2004-05-13 03:17:08.222534	145.254.160.237	65.208.228.223	TCP	54	Yes	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0
4	2004-05-13 03:17:08.222534	145.254.160.237	65.208.228.223	HTTP	533	Yes	GET /download.html HTTP/1.1
5	2004-05-13 03:17:08.783340	65.208.228.223	145.254.160.237	TCP	54	Yes	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len...
6	2004-05-13 03:17:08.993643	65.208.228.223	145.254.160.237	TCP	1434	Yes	[TCP segment of a reassembled PDU]
7	2004-05-13 03:17:09.123830	145.254.160.237	65.208.228.223	TCP	54	Yes	3372 → 80 [ACK] Seq=480 Ack=1381 Win=9660 ...
8	2004-05-13 03:17:09.123830	65.208.228.223	145.254.160.237	TCP	1434	Yes	[TCP segment of a reassembled PDU]

[Next sequence number: 480 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
Header Length: 20 bytes
▶ Flags: 0x018 (PSH, ACK)
Window size value: 9660
[Calculated window size: 9660]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0xa958 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
▶ [SEQ/ACK analysis]

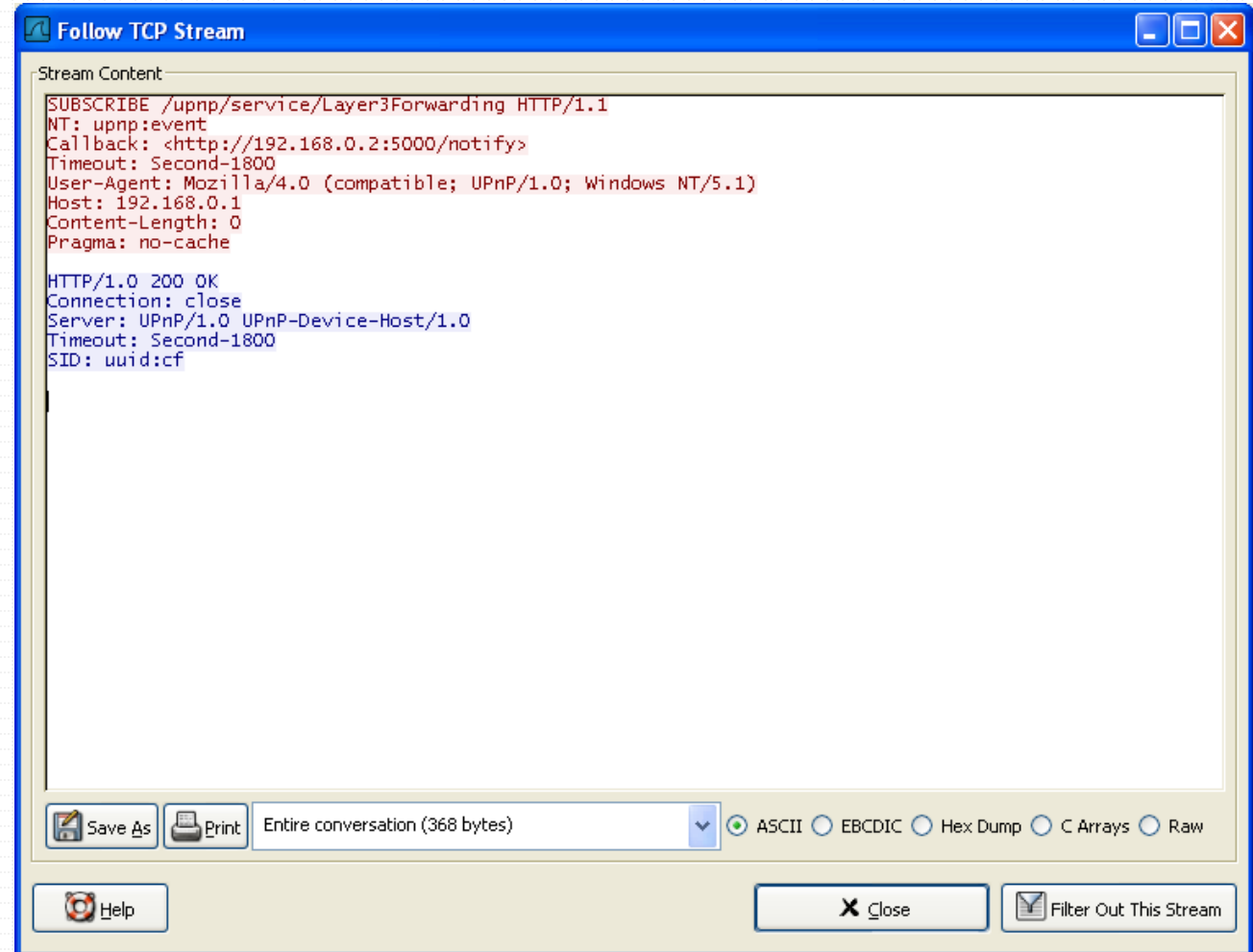
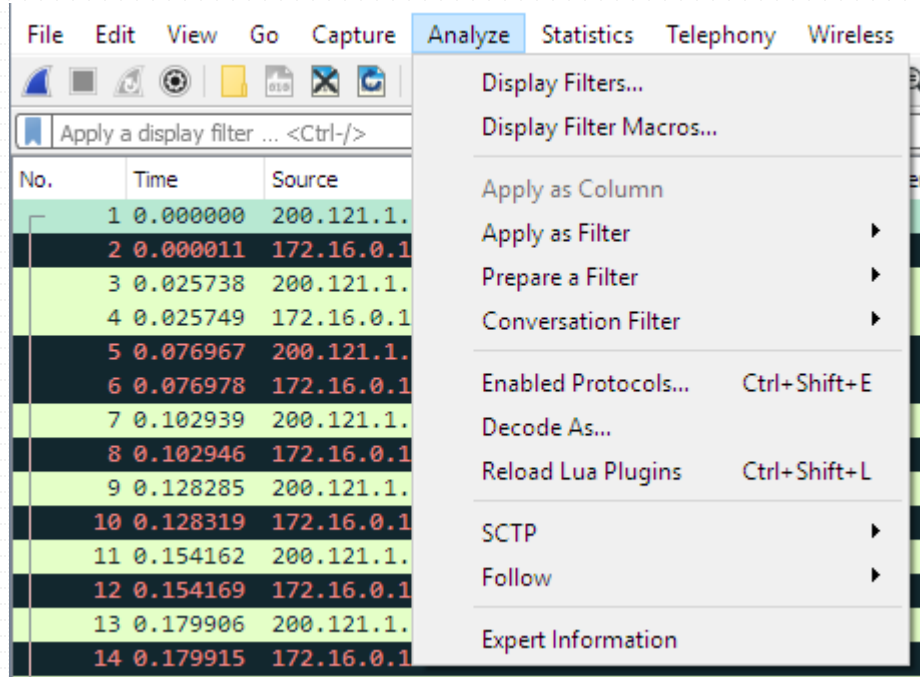
Hypertext Transfer Protocol

0030	25 bc a9 58 00 00 47 45	54 20 2f 64 6f 77 6e 6c	%..X..GE T /downl
0040	6f 61 64 2e 68 74 6d 6c	20 48 54 54 50 2f 31 2e	oad.html HTTP/1.
0050	31 0d 0a 48 6f 73 74 3a	20 77 77 77 2e 65 74 68	1..Host: www.eth
0060	65 72 65 61 6c 2e 63 6f	6d 0d 0a 55 73 65 72 2d	ereal.co m..User-
0070	41 67 65 6e 74 3a 20 4d	6f 7a 69 6c 6c 61 2f 35	Agent: M ozilla/5
0080	2e 30 20 28 57 69 6e 64	6f 77 73 3b 20 55 3b 20	.0 (Wind ows; U;
0090	57 69 6e 64 6f 77 73 20	4e 54 20 35 2e 31 3b 20	Windows NT 5.1;
00a0	65 6e 2d 55 53 3b 20 72	76 3a 31 2e 36 29 20 47	en-US; r v:1.6) G
00b0	65 63 6b 6f 2f 32 30 30	34 30 31 31 33 0d 0a 41	ecko/200 40113..A
00c0	63 63 65 70 74 3a 20 74	65 78 74 2f 78 6d 6c 2c	ccept: t ext/xml,
00d0	61 70 70 6c 69 63 61 74	69 6f 6e 2f 78 6d 6c 2c	applicat ion/xml,
00e0	61 70 70 6c 69 63 61 74	69 6f 6e 2f 78 68 74 6d	applicat ion/xhtm
00f0	6c 2b 78 6d 6c 2c 74 65	78 74 2f 68 74 6d 6c 3b	l+xml,te xt/html;

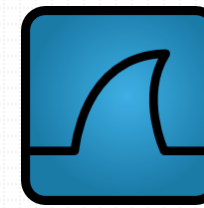
Wireshark (Ethereal)



• Analyze → Follow TCP Stream



Wireshark (Ethereal)



• Statistics → Protocol Hierarchy

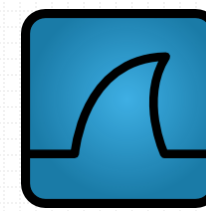
The image shows the Wireshark Statistics menu and a portion of the packet list. The Statistics menu is open, showing options like Capture File Properties, Resolved Addresses, Protocol Hierarchy, Conversations, Endpoints, Packet Lengths, I/O Graph, Service Response Time, DHCP (BOOTP) Statistics, ONC-RPC Programs, 29West, ANCP, BACnet, Collectd, DNS, Flow Graph, HART-IP, HPFEEDS, HTTP, HTTP2, Sametime, TCP Stream Graphs, UDP Multicast Streams, IPv4 Statistics, and IPv6 Statistics. The packet list shows 14 packets with details for Frame 1: 1454 bytes on wire (1163).

No.	Time	Source
1	0.000000	200.121.1.131
2	0.000011	172.16.0.122
3	0.025738	200.121.1.131
4	0.025749	172.16.0.122
5	0.076967	200.121.1.131
6	0.076978	172.16.0.122
7	0.102939	200.121.1.131
8	0.102946	172.16.0.122
9	0.128285	200.121.1.131
10	0.128319	172.16.0.122
11	0.154162	200.121.1.131
12	0.154169	172.16.0.122
13	0.179906	200.121.1.131
14	0.179915	172.16.0.122

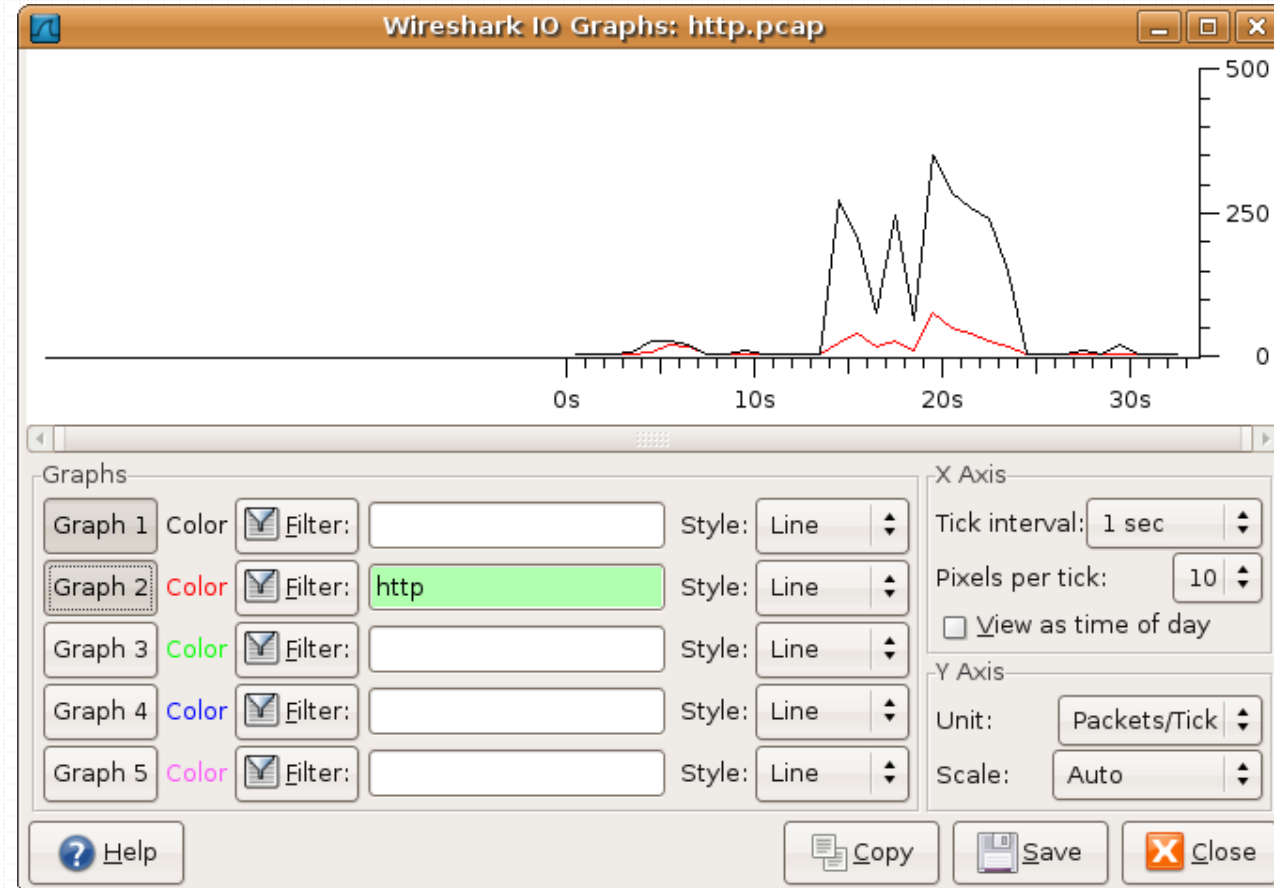
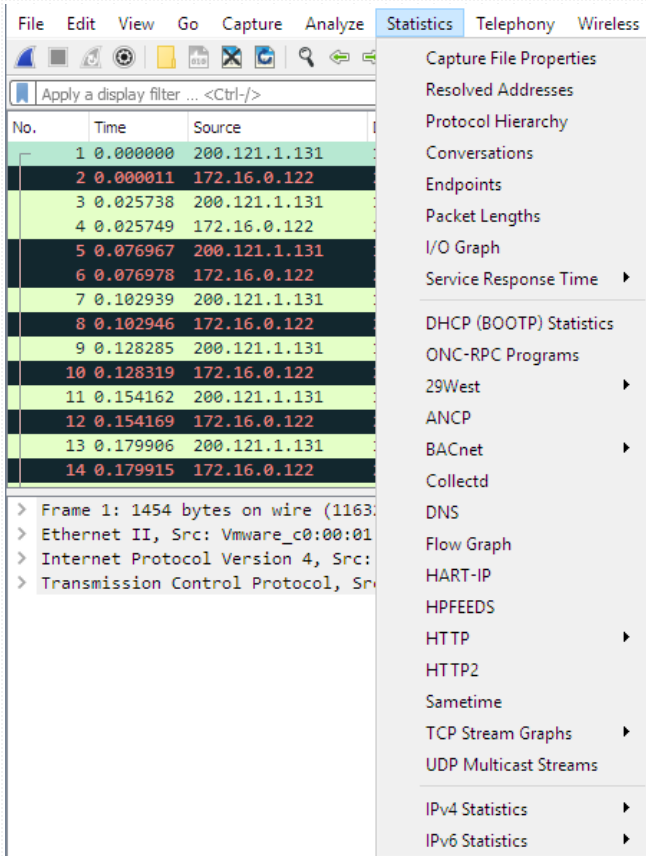
The image shows the Wireshark Protocol Hierarchy Statistics window for 'google-v4+v6'. It displays a tree view of protocols with corresponding statistics for Percent Packets, Packets, Percent Bytes, Bytes, Bits/s, End Packets, End Bytes, and End Bits/s.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	445	100.0	238413	78 k	0	0	0
Ethernet	100.0	445	2.6	6230	2,062	0	0	0
Internet Protocol Version 6	40.0	178	3.0	7120	2,356	0	0	0
Transmission Control Protocol	39.1	174	46.3	110299	36 k	148	88513	29 k
Hypertext Transfer Protocol	5.8	26	43.9	104611	34 k	14	15448	5,113
Portable Network Graphics	0.4	2	15.4	36720	12 k	2	37336	12 k
Media Type	0.2	1	0.5	1150	380	1	1464	484
Line-based text data	0.9	4	52.2	124486	41 k	4	44546	14 k
Compuserve GIF	1.1	5	1.7	3988	1,320	5	4299	1,423
Internet Control Message Protocol v6	0.9	4	0.1	128	42	4	128	42
Internet Protocol Version 4	60.0	267	2.2	5340	1,767	0	0	0
User Datagram Protocol	16.6	74	0.2	592	195	0	0	0
Dropbox LAN sync Discovery Protocol	0.4	2	0.1	208	68	2	208	68
Domain Name System	16.2	72	3.5	8414	2,785	72	8414	2,785
Transmission Control Protocol	42.2	188	41.9	99902	33 k	144	72207	23 k
Secure Sockets Layer	4.7	21	2.9	6952	2,301	20	5331	1,764
Hypertext Transfer Protocol	5.4	24	37.0	88154	29 k	13	13622	4,508
Portable Network Graphics	0.2	1	3.1	7330	2,426	1	7641	2,529
Line-based text data	1.1	5	71.5	170403	56 k	5	61353	20 k
Compuserve GIF	1.1	5	1.7	3988	1,320	5	4299	1,423
Internet Control Message Protocol	1.1	5	0.1	180	59	5	180	59

Wireshark (Ethereal)



- Statistics → I/O Graph (throughput)
- How about loss, delay, and jitter?

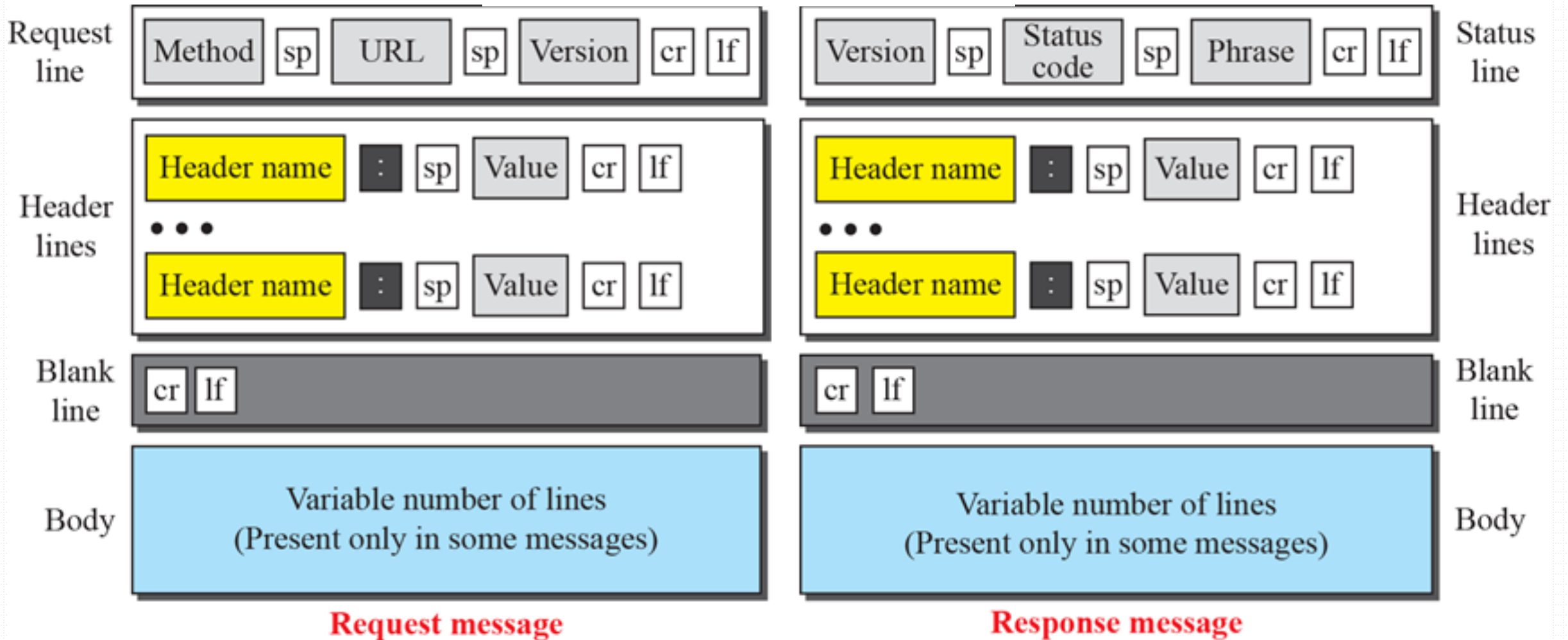


HTTP Packet Format



圖片來源：Computer Networks - A Top Down Approach，McGraw-Hill出版

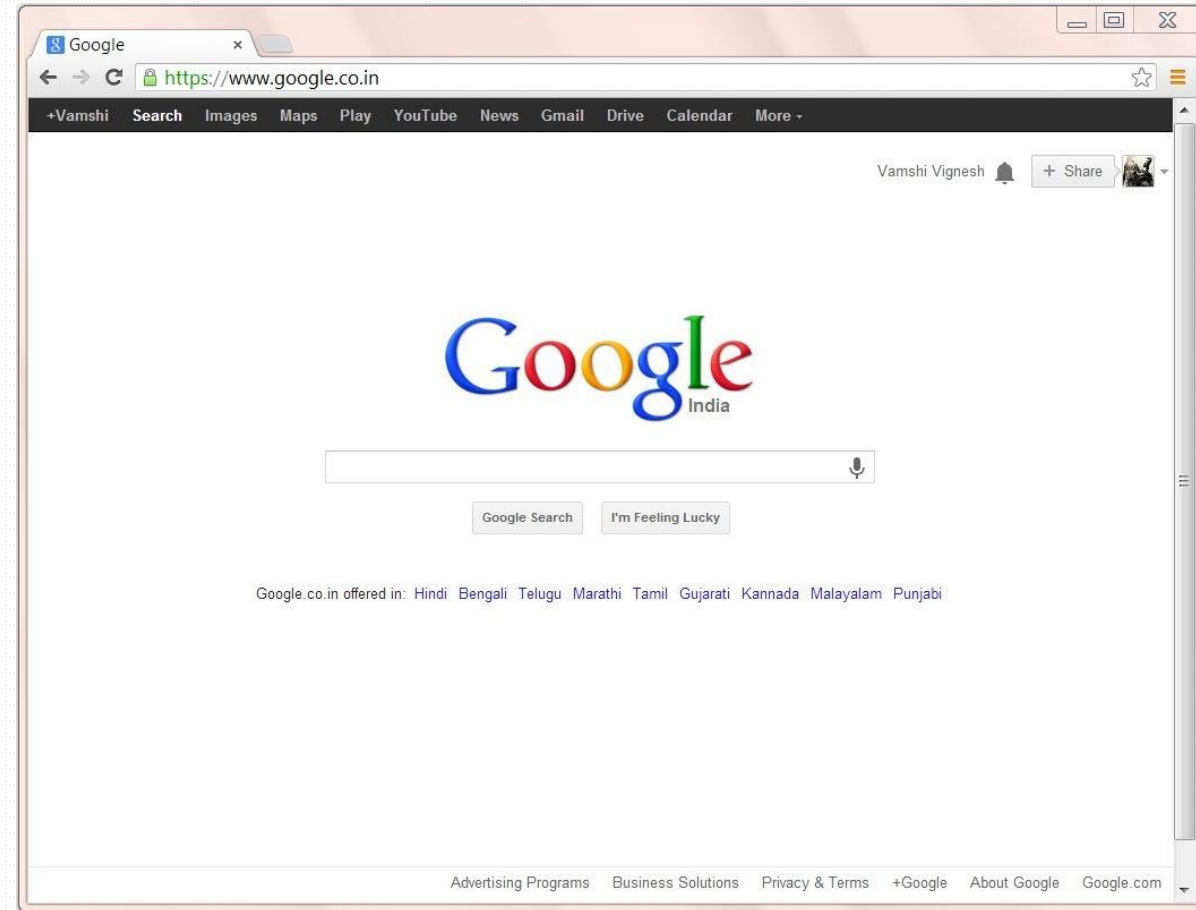
sp: Space cr: Carriage Return lf: Line Feed



Google Chrome Browser



- 由Google開發的免費網頁瀏覽器
 - 對應之開放原始碼計劃名為Chromium
 - Google Chrome本身是非自由軟體，未開放原始碼
- 官方網站：
<https://www.google.com.tw/chrome/>



Google Chrome Browser



- Chrome 開發人員工具

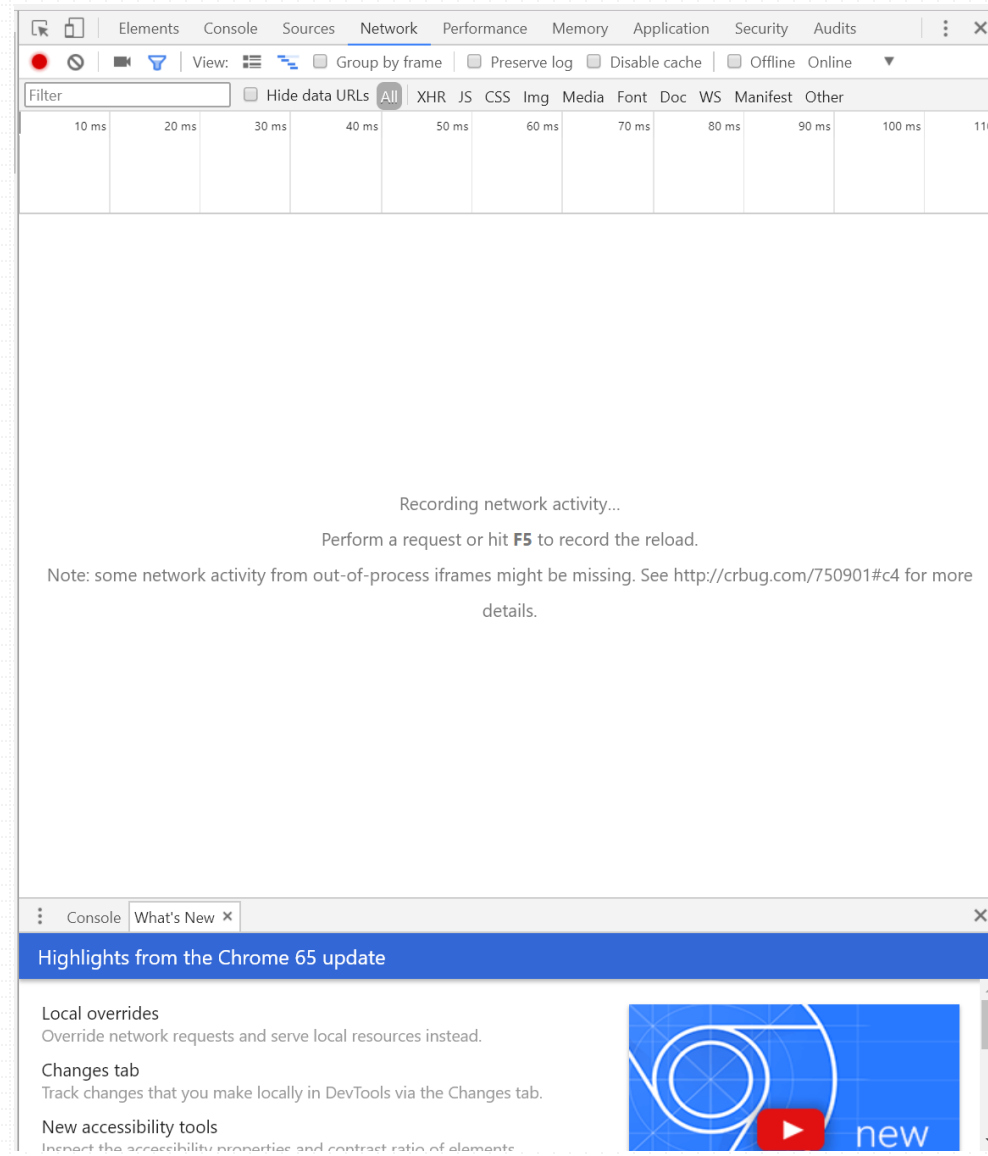
- 更多工具 → 開發者工具

- Network

- 瞭解請求和下載的資源文件並優化網頁加載性能

- 官方說明

- <https://developers.google.com/web/tools/chrome-devtools/network-performance/resource-loading?hl=zh-tw>
 - <https://developers.google.com/web/tools/chrome-devtools/network-performance/understanding-resource-timing?hl=zh-tw>
 - <https://developers.google.com/web/tools/chrome-devtools/network-performance/network-conditions?hl=zh-tw>



Google Chrome Browser



- Network 面板由五個窗格組成：

1. Controls

- 使用這些選項可以控制 Network 面板的外觀和功能。

2. Filters

- 使用這些選項可以控制在 Requests Table 中顯示哪些資源。提示：按住 Cmd (Mac) 或 Ctrl (Windows/Linux) 並點擊過濾器可以同時選擇多個過濾器。

3. Overview

- 此圖表顯示了資源檢索時間的時間線。如果您看到多條豎線堆疊在一起，則說明這些資源被同時檢索。

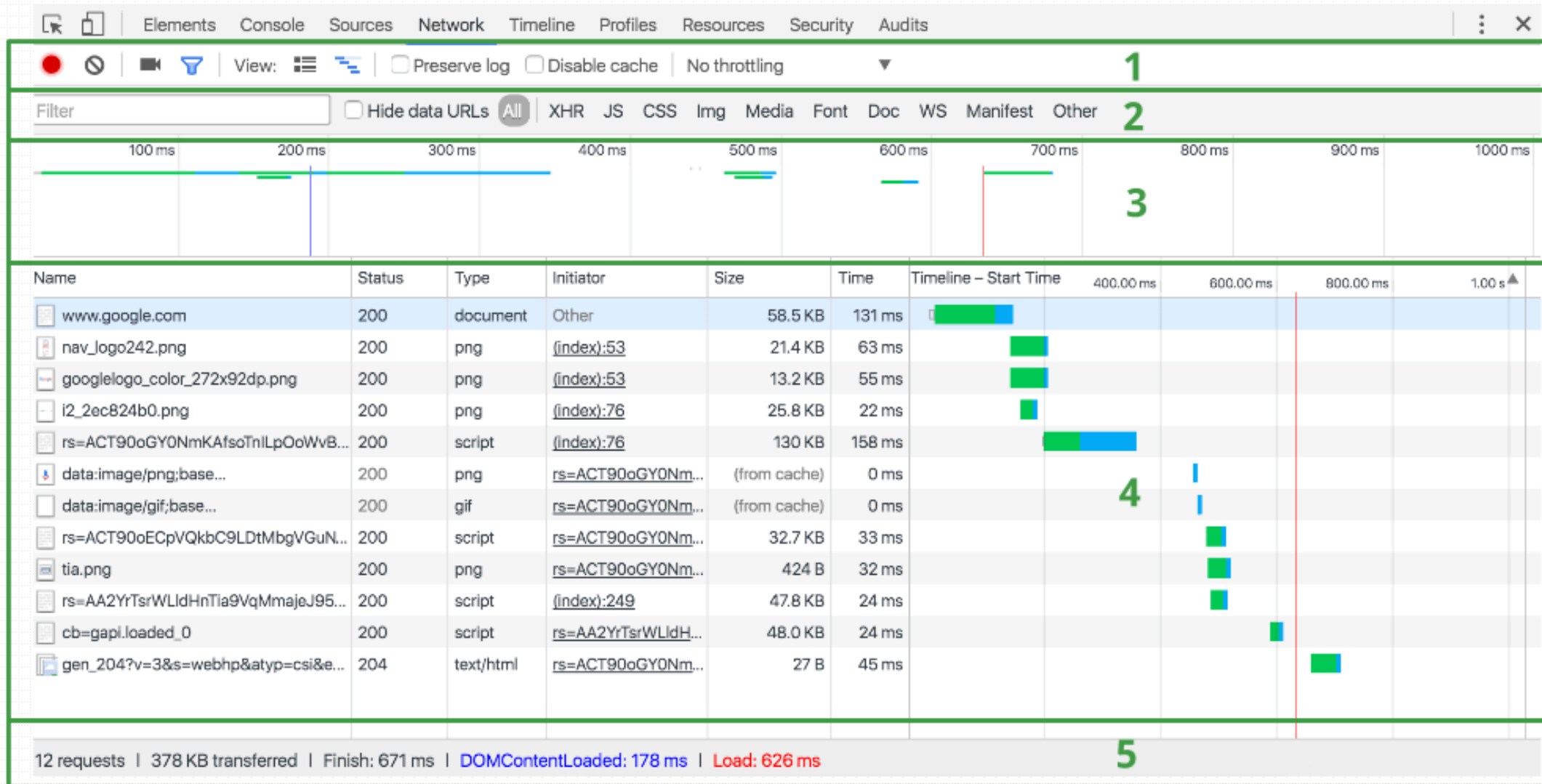
4. Requests Table

- 此表格列出了檢索的每一個資源。默認情況下，此表格按時間順序排序，最早的資源在頂部。點擊資源的名稱可以顯示更多信息。提示：右鍵點擊 Timeline 以外的任何一個表格標題可以添加或移除信息列。

5. Summary

- 此窗格可以一目瞭然地告訴您請求總數、傳輸的數據量和加載時間。

Google Chrome Browser



圖片來源：<https://developers.google.com/web/tools/chrome-devtools/network-performance/resource-loading?hl=zh-tw>

Google Chrome Browser



Name	× Headers Preview Response Timing
tools.css	<div>▼ General</div> <div>Request URL: <code>https://developers.google.com/web/ugging.png</code></div> <div>Request Method: GET</div> <div>Status Code: ● 200</div> <div>Remote Address: <code>[2607:f8b0:4005:800::100e]:443</code></div> <div>▼ Response Headers</div> <div>alt-svc: <code>quic=":443"; ma=2592000; v="30,29,28,27</code></div> <div>alternate-protocol: <code>443:quic,p=1</code></div> <div>cache-control: <code>must_revalidate, public, max-age=</code></div> <div>content-language: <code>en</code></div> <div>content-type: <code>image/png</code></div> <div>date: <code>Tue, 23 Feb 2016 17:50:23 GMT</code></div> <div>expires: <code>Tue, 23 Feb 2016 18:50:23 GMT</code></div> <div>last-modified: <code>Wed. 14 Oct 2015 23:42:52 GMT</code></div>
developers-logo-no-brackets.svg	
github-mark.svg	
feedback.svg	
chrome_devtools.svg	
settings.png	
remotedebugging.png	
devicemode.png	
elements.png	
console.png	
network.png	
38 requests 657 KB transferred ...	

查核點：網頁還原



- 目標

- 請任選一個網站，使用**Chrome**開啟該網站過程中，利用**Wireshark**擷取**HTTP**封包後，從中還原出下載的網頁檔案。

- 檢查項目

- 使用**Chrome**開發人員工具顯示網頁下載的**HTTP Header**
- 使用**Wireshark**顯示網頁下載的**HTTP Header**
- 顯示利用**Wireshark**還原的網頁檔案

查核點：影片還原



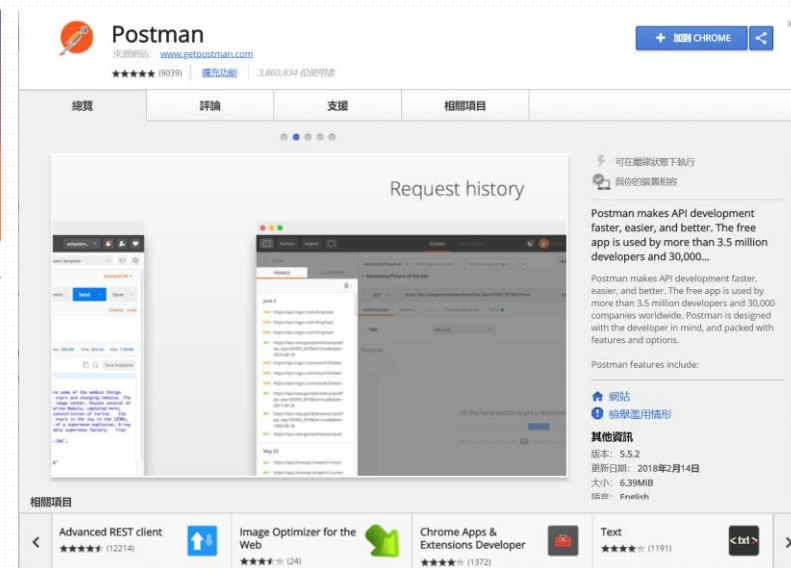
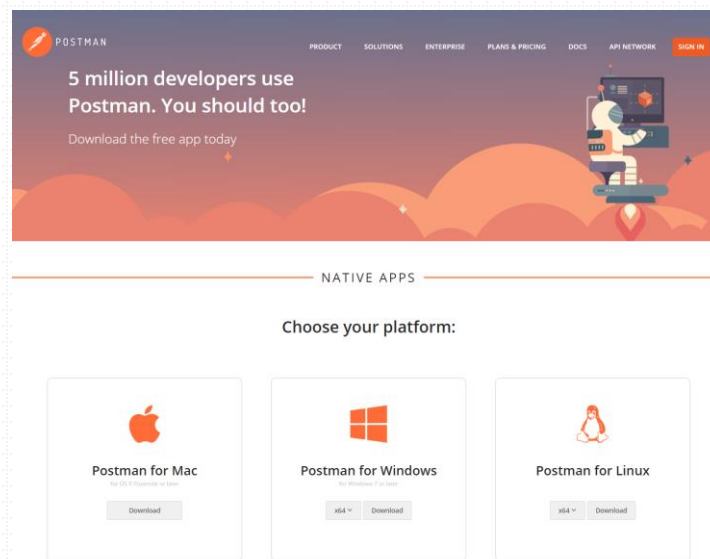
- 目標：
 - 請任選一個具有URL的視訊檔案
 - 如：<http://140.124.70.120/course/video.html> 內有多個video
 - 使用Chrome開啟視訊檔案的URL，利用Wireshark擷取HTTP封包後，從中還原出下載的視訊檔案。
- 檢查項目：
 - 播放還原的視訊檔案

Postman



- 用來測試 Web Service 的方便工具，可快速產生複雜之 HTTP Requests

- 可選安裝可執行檔，或Chrome的擴充元件



- 官方網站：<https://www.getpostman.com/>
- 官方說明：<https://www.getpostman.com/docs/v6/>

Postman



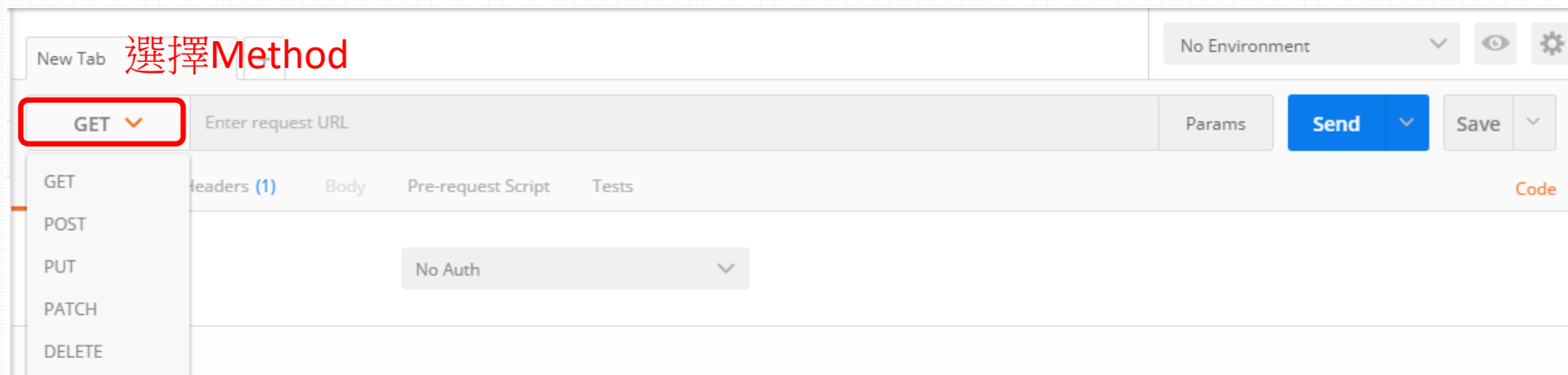
The screenshot shows the Postman application interface. The top bar includes tabs for Runner, Import, and Builder, along with a Team Library link and a status indicator 'IN SYNC'. The left sidebar features a 'Filter' search bar, a 'History' tab, and a 'Collections' tab. The 'History' tab is highlighted with a red box and the word 'History' in red text. It lists several recent requests, including GET and POST requests to various URLs. The main workspace is divided into two sections. The top section, labeled 'input' in red text, shows the request configuration for a GET request to 'http://127.0.0.1:1880/test'. It includes tabs for Authorization, Headers (1), Body, Pre-request Script, and Tests. The 'Headers' tab is selected, showing a single header: 'Content-Type' with the value 'application/json'. The bottom section, labeled 'output' in red text, shows the response configuration for the same request. It includes tabs for Body, Cookies, Headers (6), and Tests. The 'Body' tab is selected, showing the response in 'Pretty' format:

```
{ "type": "string", "unit": "fake data" }
```

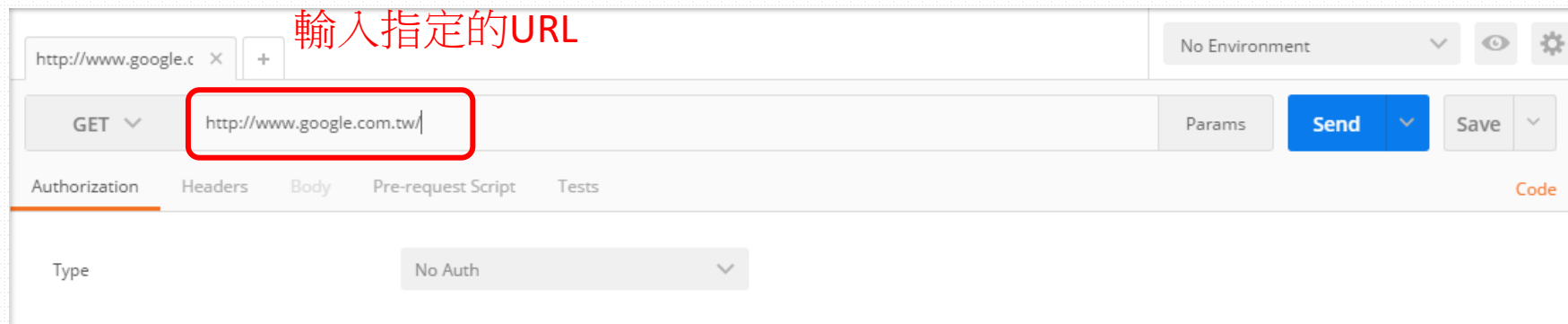
. The status bar at the bottom right indicates 'Status: 200 OK' and 'Time: 21 ms'.



- 設定HTTP Request - Method

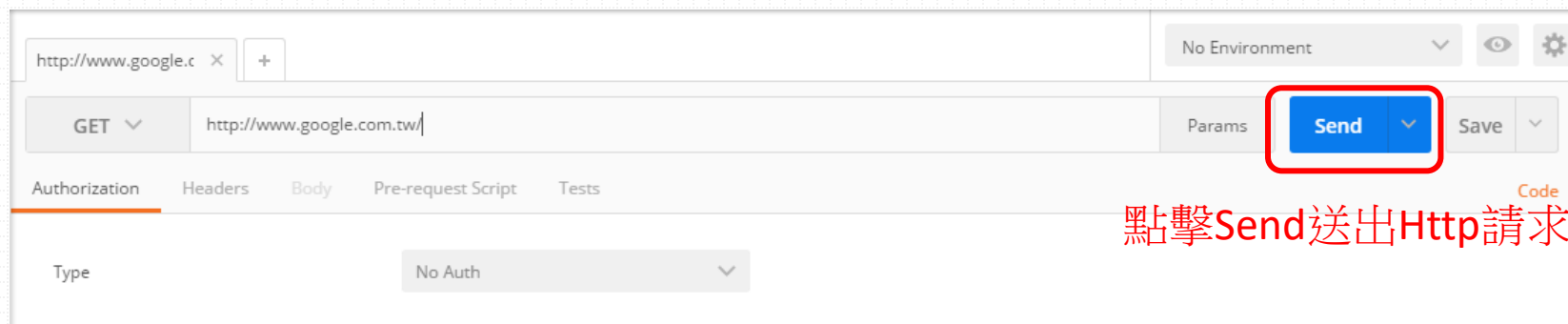


- 設定HTTP Request - URL

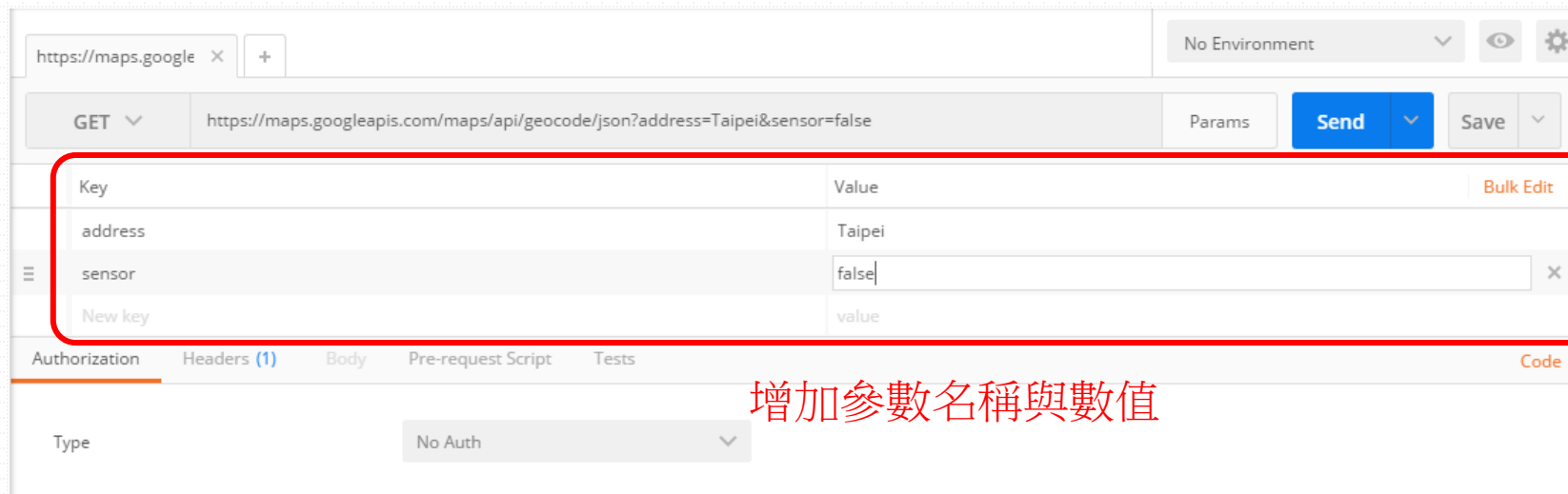




- 送出HTTP Request



- [若有需求] 設定HTTP Request - 參數





• 察看HTTP Response - Body

選擇Body

HTTP狀態碼、響應時間

Body Cookies Headers (10) Tests

Status: 200 OK Time: 360 ms

Pretty Raw Preview HTML

HTTP response (Body)

```
1 <!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="zh-TW"><head><meta content="/images/branding/google/1x
  /googleleg_standard_color_128dp.png" itemprop="image"><link href="/images/branding/product/ico/googleleg_lodp.ico" rel="shortcut icon"><meta content
  ="origin" id="mref" name="referrer"><title>Google</title><script>(function(){window.google={kEI:'bef8WIHxN4ja8QXqw4KwCA',kEXPI:'201761,1351176
  ,1352864,1352996,1353038,1353096,1353098,1353108,1353476,3700297,3700347,3700405,4029815,4031109,4032677,4036527,4038214,4038394,4039268,4041776
  ,4043492,4045096,4045293,4045841,4046904,4047140,4047454,4048347,4048980,4050750,4051887,4056126,4056682,4058016,4061666,4061980,4062724,4063220
  ,4064468,4064796,4065787,4069829,4071757,4072270,4072364,4072777,4073111,4076096,4076999,4078430,4078588,4079081,4079423,4080760,4081038,4081164
  ,4082131,4082230,4083044,4083458,4085336,4090550,4090553,4090806,4090877,4092934,4093314,4093497,4094252,4094544,4094837,4095558,4095910,4095998
  ,4096323,4097153,4097922,4097929,4098458,4098733,4098740,4098752,4100171,4100379,4100682,4100714,4100828,4101376,4101429,4101684,4101750,4102717
  ,4103469,4103475,4103849,4103999,4104202,4104723,4105085,4105317,4105470,4105556,4105649,4106605,4106949,4107424,4107428,4107437,4107450,4107454
  ,4107555,4107628,4107866,4107899,4107968,4107989,4108380,4108417,4108538,4108540,4108870,4108888,8300509,8503585,8508229,8508707,8508931,8509037
  ,8509373,10200083,10201957,10202317,16200026,19001732,19001735,19002007,19002038,41027342',authuser:0,j:{en:1,bv:24,u:'c9c918f0',qbp:0},kscs
  : 'c9c918f0_24'};google.kHL='zh-TW'}});(function(){google.lc=[];google.li=0;google.getEI=function(a){for(var b;a&&(!a.getAttribute)||!(b=a
  .getAttribute("eid")));a=a.parentNode;return b}|google.kEI};google.getLEI=function(a){for(var b=null;a&&(!a.getAttribute)||!(b=a.getAttribute("leid"
  )));a=a.parentNode;return b};google.https=function(){return"https://"+window.location.protocol};google.ml=function(){return null};google.wl=function
  (a,b){try{google.ml(Error(a),!1,b)}catch(c){};google.time=function(){return(new Date).getTime();google.log=function(a,b,c,d,g){a=google.logUrl(a,b
  ,c,d,g);if("!"=a){b=new Image;var e=google.lc,f=google.li;e[f]=b;b.onerror=b.onload=b.onabort=function(){delete e[f];window.google&&window.google
  .vel&&window.google.vel.lu&&window.google.vel.lu(a);b.src=a;google.li=f+1};google.logUrl=function(a,b,c,d,g){var e="",f=google.ls|"";c||-1!=b
  .search("&ei=")||!(e="&ei="+google.getEI(d),-1==b.search("&lei=")&&(d=google.getLEI(d)&&(e+="&lei="+d));a=c||"/"+(g||"gen_204")+"?atyp=i&ct="+a
  +"&cad="+b+e+f+"&z="+google.time();/^http/i.test(a)&&google.https()}&&(google.ml(Error("a"),!1,{src:a,glmm:1}),a="");return a};google.y={};google.x
  =function(a,b){google.y[a.id]=[a,b];return!1};google.lq=[];google.load=function(a,b,c){google.lq.push([a,b,c]);google.loadAll=function(a,b
  )(google.lq.push([a,b]));}).call(this);
2 google.j.b=(!location.hash&&!location.hash.match('[#&]((q|fp)=|tbs=rimg|tbs=simg|tbs=sbi)'))
3 |(google.j.qbp==1);(function(){google.hs={h:true,pa:true,q:false;})();(function(){google.c={c:{a:true,d:false,i:false,m:true,n:false};google.sn
  ='webhp';(function(){var e=function(a,b,c){a.addEventListener?a.removeEventListener(b,c,!1):a.attachEvent&&a.detachEvent("on"+b,c);g=function(a,b,c
  ){f.push({o:a,v:b,w:c});a.addEventListener?a.addEventListener(b,c,!1):a.attachEvent&&a.attachEvent("on"+b,c);f=[];google.timers={};google.startTick
  =function(a,b){var c=b&&google.timers[b].t;google.timers[b].t.start:google.time();google.timers[a]={t:{start:c,e:{},it:{},m:{}};(c=window
  .performance)&&c.now&&(google.timers[a].wsrt=Math.floor(c.now()));google.tick=function(a,b,c){google.timers[a]|google.startTick(a);c=c||google
  .time();b instanceof Array||(b=[b]);for(var d=0;d<b.length;++d)google.timers[a].t[b[d]]=c;google.c.e=function(a,b,c){google.timers[a].e[b]=c
```



• 察看HTTP Response - Header

選擇Headers

Body Cookies **Headers (10)** Tests Status: 200 OK Time: 360 ms

alt-svc → quic=":443"; ma=2592000; v="37,36,35"

cache-control → private, max-age=0

content-encoding → gzip

content-type → text/html; charset=UTF-8 Body資料格式為html，編碼為utf-8

date → Sun, 23 Apr 2017 17:42:05 GMT

expires → -1

server → gws

status → 200 狀態碼為200(OK)

x-frame-options → SAMEORIGIN

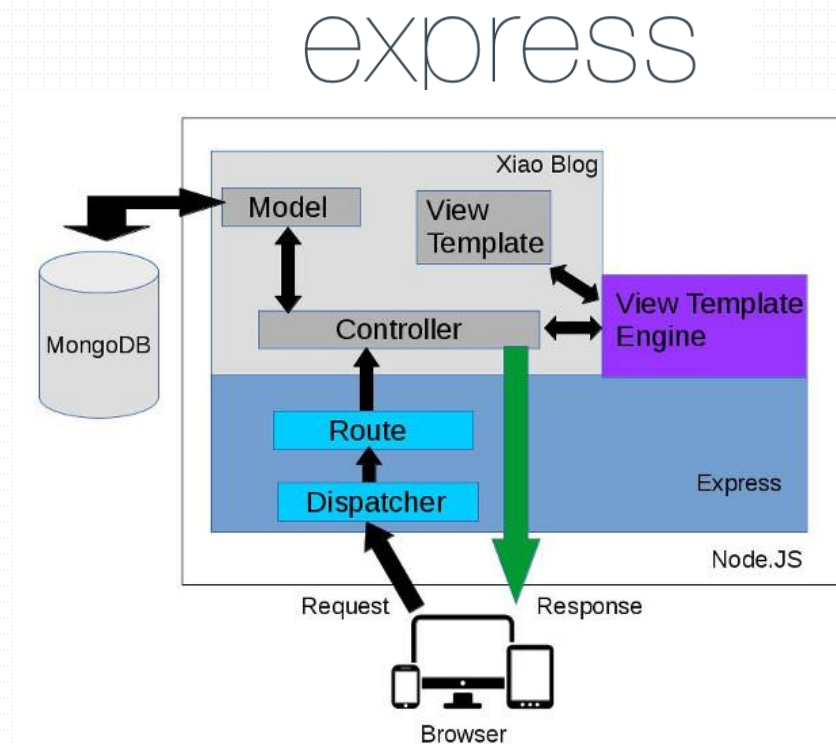
x-xss-protection → 1; mode=block

HTTP response (Headers)

Express



- Express是最小又靈活的Node.js Web應用程式架構，為Web與行動式應用程式提供一組健全的特性。
 - <http://expressjs.com/>
- 大量的HTTP公用程式方法與中介軟體供您支配，能夠快速又輕鬆的建立完整的API。

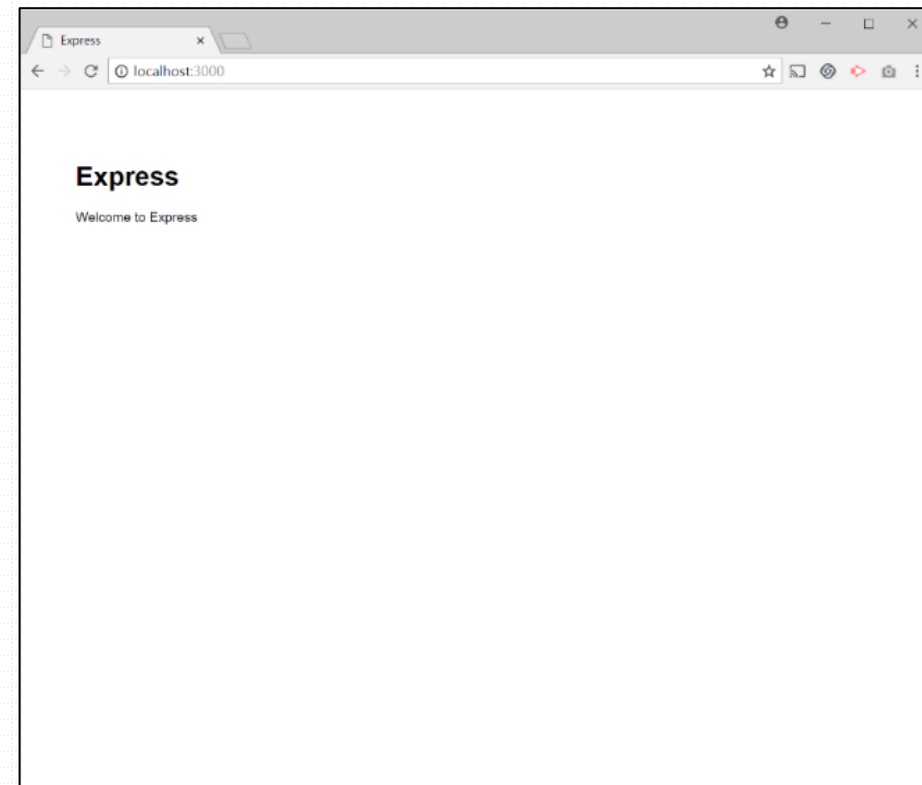


Express-Generator



- Express提供快速建立應用程式架構的工具

1. `npm install express-generator -g`
2. `express myExpressApp`
3. `cd myExpressApp`
4. `npm install`
5. `npm start`
6. 用瀏覽器打開<http://localhost:3000>
7. Ctrl+C可以中斷指令執行



基礎路由(Basic Routing)



- <http://expressjs.com/en/starter/basic-routing.html>
- 基礎語法：app.METHOD(PATH, HANDLER)
 - app是express的實例(instance)。
 - METHOD是HTTP要求方法，如：get、post、put、delete。
 - PATH是伺服器上的路徑。
 - HANDLER是當路由相符時要執行的函數。
- 示範：

```
app.get('/', function (req, res)
{
  res.send('Hello World!');
});
```

路由器層次中介軟體(Router-level Middleware)



- <http://expressjs.com/en/guide/using-middleware.html>
- Express 中介軟體(Middleware)
 - 代表有權存取(1)要求物件(req)、(2)回應物件(res)以及(3)呼叫下一個中介軟體函數的函數。

```
var app = express();
var router = express.Router();
// a middleware function with no mount path.
// This code is executed for every request to the router
router.use(function (req, res, next) {
  console.log('Time:', Date.now());
  next();
});
```

靜態檔案(Static Files)



- <http://expressjs.com/en/starter/static-files.html>
- 使用內建函式：`express.static(root, [options])`
 - 示範：`app.use(express.static('public'));`
 - 代表可以載入位於public目錄中的檔案
 - `http://localhost:3000/images/kitten.jpg`
 - `http://localhost:3000/css/style.css`
 - `http://localhost:3000/js/app.js`
 - `http://localhost:3000/images/bg.png`
 - `http://localhost:3000/hello.html`



- <https://flaviocopes.com/express-request-parameters/>
- 若使用GET，參數夾帶在URL中，則使用req.query物件

GET /test?name=fred&tel=0926xxx572

```
app.get('/test', function(req, res) {  
  console.log(req.query.name);  
  console.log(req.query.tel);  
});
```



- <https://flaviocopes.com/express-request-parameters/>
- 若使用POST，參數夾帶在HTTP封包中，則使用req.body物件

```
<form action='/test' method='post'>  
  <input type='text' name='name' value='fred'>  
  <input type='text' name='tel' value='0926xxx572'>  
  <input type='submit' value='Submit'>  
</form>
```

```
app.post('/test', function(req, res) {  
  console.log(req.query.id);  
  console.log(req.body.name);  
  console.log(req.body.tel);  
});
```

查核點：協定追蹤



- 目標：
 - 利用Express設計出自己的REST/RESTful API (可接收表單參數)
 - 利用Postman送出HTTP Request 附帶表單參數
 - 利用Wireshark擷取封包後，找出Express回傳的資料
- 查核項目：
 - 使用Wireshark顯示回傳的資料內容

自行探索



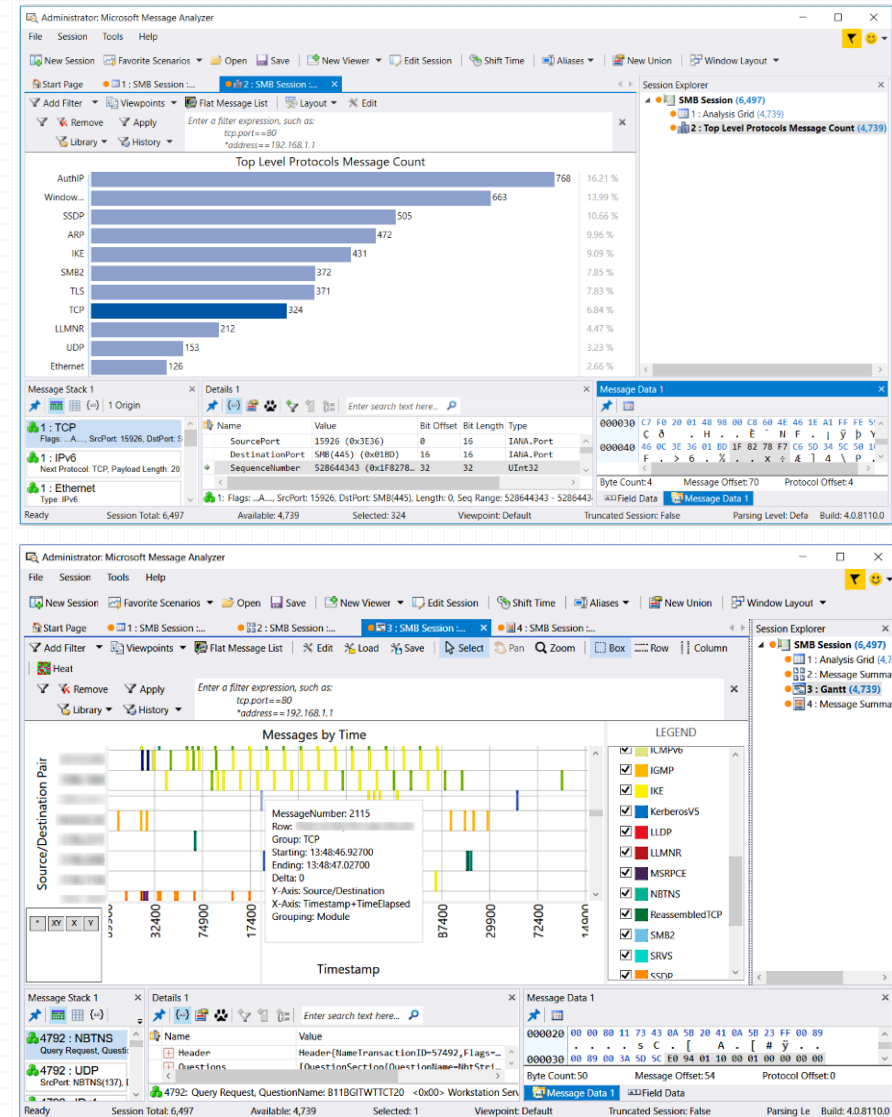
NTUT NESL



MS Message Analyzer (Network Monitor)



- A new tool for capturing, displaying, and analyzing protocol messaging traffic, events, and other system or application messages in network troubleshooting and other diagnostic scenarios.
 - It is the successor to Microsoft Network Monitor 3.4 and is a key component in the Protocol Engineering Framework (PEF).
- 下載網址：<https://www.microsoft.com/en-us/download/details.aspx?id=44226>
- 官方說明：<https://docs.microsoft.com/en-us/message-analyzer/microsoft-message-analyzer-operating-guide>



其他網路封包分析器



◆	Creator ◆	Latest release ◆	User interface ◆	Software license ◆	Cost ◆
Analyze This	Comoe Networks		Web GUI	N/A	?
Cain and Abel	Massimiliano Montoro	4.9.56 / April 7, 2014	GUI	Freeware	Free
Capsa	Colasoft	10.0 / July 26, 2017 ^[1]	GUI	Proprietary	\$0-\$995, depending on version ^[2]
Carnivore	Federal Bureau of Investigation	?	?	N/A	?
Charles Web Debugging Proxy	Karl van Randow	4.1.4 / July 10, 2017	GUI	?	\$30-\$50 (Free Trial)
Clarified Analyzer	Clarified Networks		GUI	Proprietary	Non-free
Clusterpoint Network Traffic Surveillance System	Clusterpoint		web GUI	Proprietary	?
CommView	TamoSoft	6.5	GUI	Proprietary	\$299-\$599, \$149 1 year subscription
Debookee	iwaxx	6.0.0b2 (2278) / July 21, 2017 ^[3]	GUI	Proprietary	\$29.90-\$69.90
dSniff	Dug Song	2.3 / December 17, 2000 ^[4]	CLI	BSD License	Free
EtherApe	Juan Toledo	0.9.14 / February 6, 2016 ^[5]	GUI	GNU General Public License	Free
Ettercap	ALoR and NaGA	0.8.2-Ferri / March 14, 2015 ^[6]	Both	GNU General Public License	Free
Fiddler	Eric Lawrence	4.6.3.50306 / 9 December 2016	GUI	Freeware	Free
justniffer	The Justniffer team	0.5.15 / March 21, 2016 ^[7]	CLI	GNU General Public License	Free
Kismet	Mike Kershaw (dragorn)	2016-01-R1 / January 31, 2016 ^[8]	CLI	GNU General Public License	Free
Microsoft Message Analyzer	Microsoft	1.4 / October 28, 2016 ^[9]	GUI	Proprietary	Free
Microsoft Network Monitor	Microsoft	3.4 / June 24, 2010	GUI	Proprietary	Free
netsniff-ng	Daniel Borkmann	0.6.2 / November 7, 2016	CLI	GNU General Public License	Free
ngrep	Jordan Ritter	1.45 (11/18/06)	CLI	BSD-style	Free
Observer	Viavi Solutions (formerly Network Instruments)		GUI	Proprietary	Price on request
OmniPeek (formerly AiroPeek, EtherPeek)	Savvius (formerly WildPackets)	11.1 / November, 2017	GUI	Proprietary	\$1194-\$5994, depending on version ^[10]
SteelCentral Transaction Analyzer	OPNET Technologies/Riverbed Technology	17.0.T-PL1 / June 9, 2014 ^[11]	GUI	Proprietary	Non-free
snoop	Sun Microsystems	Solaris 10 / December 11, 2006	CLI	CDDL	Free
tcpdump	The Tcpdump team	4.8.1 / October 25, 2016 ^[12]	CLI	BSD License	Free
Tranalyzer	The Tranalyzer team	0.7.5 / February 10, 2018 ^[13]	CLI	GNU General Public License	Free
Wireshark (formerly Ethereal)	The Wireshark team	2.4.5 / February 23, 2018 ^[14]	Both	GNU General Public License	Free
Xplico	The Xplico team	1.2.0 / February 1, 2017 ^[15]	Both	GNU General Public License	Free

圖片來源：https://en.wikipedia.org/wiki/Comparison_of_packet_analyzers



- SIPp

- <http://sipp.sourceforge.net/index.html>
- <http://sipp-wip.readthedocs.io/en/latest/>

```

ocadmin@vista:~/sipp
----- Scenario Screen ----- [1-4]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
    10 cps(0 ms)   5061      4.01 s         40  127.0.0.1:5060(UDP)

10 new calls during 1.000 s period      16 ms scheduler resolution
0 concurrent calls (limit 30)           Peak was 1 calls, after 0 s
0 out-of-call msg (discarded)
1 open sockets

      Messages  Retrans  Timeout  Unexpected-Msg
INVITE ----->      40      0      0
    100 <-----      0      0      0
    180 <-----      40      0      0
    200 <----- E-RTD  40      0      0
    ACK ----->      40      0
      [    0 ms]
    BYE ----->      40      0      0
    200 <-----      40      0      0

----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----

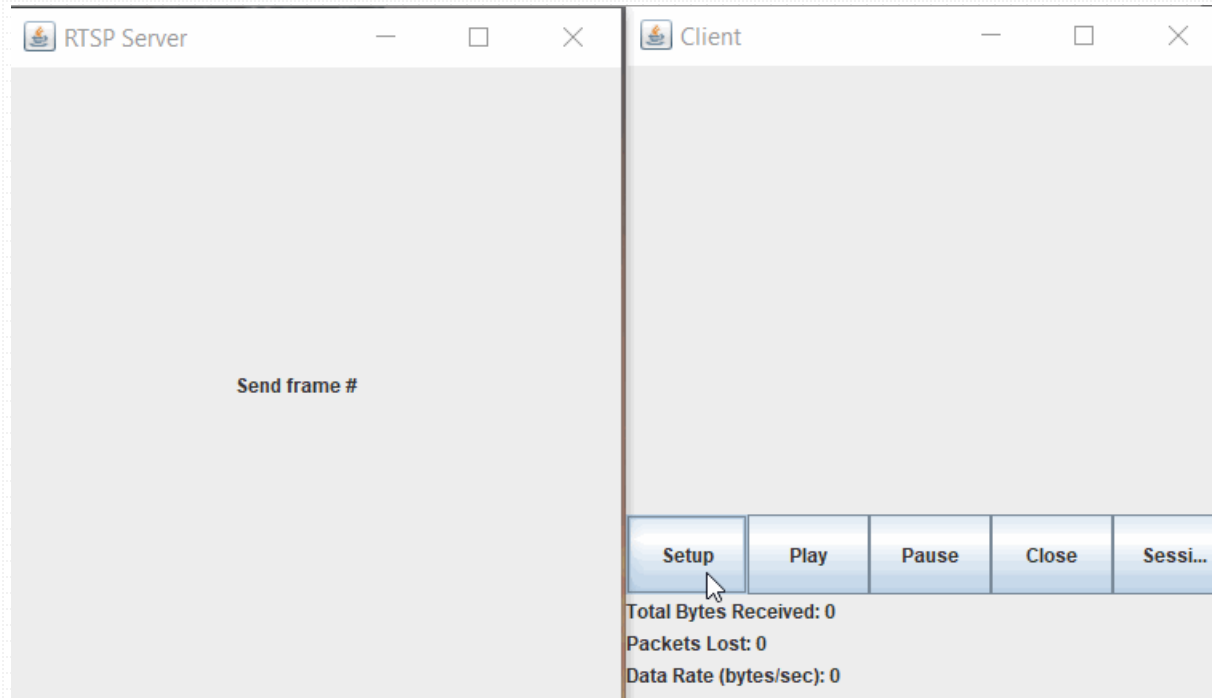
```

RTP/RTCP/RTSP



- RTSP Client & Server

- <https://www.csee.umbc.edu/~pmundur/courses/CMSC691C/lab5-kurose-ross.html>
- <https://github.com/mutaphore/RTSP-Client-Server>





總結



- 網路封包分析器(如：Wireshark)可幫助具體了解網路封包傳遞與網路資料交換行為。
 - 封包標頭(Header)解析
 - 封包交換先後順序
 - 傳輸效能分析與統計

