

# PROGETTO LEZIONE N5

Come premessa per lo svolgimento dell'esercizio veniva richiesta l'assegnazione di specifici indirizzi IP e DNS server sulla macchina Kali (server) e Windows 7 (client)

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\vboxuser>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::8508:a43:cbb5:5453%11
    IPv4 Address. . . . . : 192.168.32.101
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.32.1

Tunnel adapter isatap.{45E6035F-8DF4-4472-83E4-3BE108E298CB}:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 

C:\Users\vboxuser>
```

```
francesco@kali: ~
File Actions Edit View Help

TX packets 559 bytes 55812 (54.5 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(francesco@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.32.100 netmask 255.255.255.0 broadcast 192.168.32.255
    inet6 fe80::a00:27ff:feb1:88e2 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:b1:88:e2 txqueuelen 1000 (Ethernet)
    RX packets 1197 bytes 248728 (242.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 768 bytes 59664 (58.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 559 bytes 55812 (54.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 559 bytes 55812 (54.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(francesco@kali)-[~]
$
```

e lo stato di attivato sui servizi HTTP/HTTPS e DNS server con IP della macchina Kali

```
francesco@kali: ~  
File Actions Edit View Help  
GNU nano 7.2 /etc/inetsim/inetsim.conf  
  
#####  
Service DNS  
#####  
  
# dns_bind_port  
#  
# Port number to bind DNS service to  
# Syntax: dns_bind_port <port number>  
#  
# Default: 53  
#  
#dns_bind_port 53  
  
#####  
# dns_default_ip  
#  
# Default IP address to return with DNS replies  
# Syntax: dns_default_ip <IP address>  
#  
# Default: 192.168.32.100  
#  
dns_default_ip 192.168.32.100  
  
"the quieter you become, the more you are able to hear"  
#####  
# dns_default_hostname  
#  
# Default hostname to return with DNS replies  
# Syntax: dns_default_hostname <hostname>  
  
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute  
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

```
francesco@kali: ~  
File Actions Edit View Help  
GNU nano 7.2 /etc/inetsim/inetsim.conf *  
  
# The services to start  
#  
# Syntax: start_service <service name> sim/report  
#  
# Default: none  
#  
# Available service names are:  
# dns, http, smtp, pop3, tftp, ftp, ntp, time_tcp, file, /etc/inetsim/inetsim  
# time_udp, daytime_tcp, daytime_udp, echo_tcp,  
# echo_udp, discard_tcp, discard_udp, quotd_tcp,  
# quotd_udp, chargen_tcp, chargen_udp, finger,=  
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,  
# ftps, irc, https  
  
start_service dns 11-09-19 14:55:19 (delta: 0 seconds)  
start_service http  
start_service https started (PID 18683)  
start_service smtp handle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm 1  
start_service smtps  
start_service pop3 handle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm 1  
start_service pop3s  
start_service ftp started (PID 18685)  
start_service ftps started (PID 18684)  
  
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute  
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

```
francesco@kali: ~  
File Actions Edit View Help  
GNU nano 7.2 /etc/inetsim/inetsim.conf *  
  
# The services to start  
#  
# Syntax: start_service <service name> sim/report  
#  
# Default: none  
#  
# Available service names are:  
# dns, http, smtp, pop3, tftp, ftp, ntp, time_tcp, file, /etc/inetsim/inetsim  
# time_udp, daytime_tcp, daytime_udp, echo_tcp,  
# echo_udp, discard_tcp, discard_udp, quotd_tcp,  
# quotd_udp, chargen_tcp, chargen_udp, finger,=  
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,  
# ftps, irc, https  
  
start_service dns 11-09-19 14:55:19 (delta: 0 seconds)  
start_service http  
start_service https started (PID 18683)  
start_service smtp handle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm 1  
start_service smtps  
start_service pop3 handle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm 1  
start_service pop3s  
start_service ftp started (PID 18685)  
start_service ftps started (PID 18684)  
  
[ Cancelled ]  
  
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute  
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

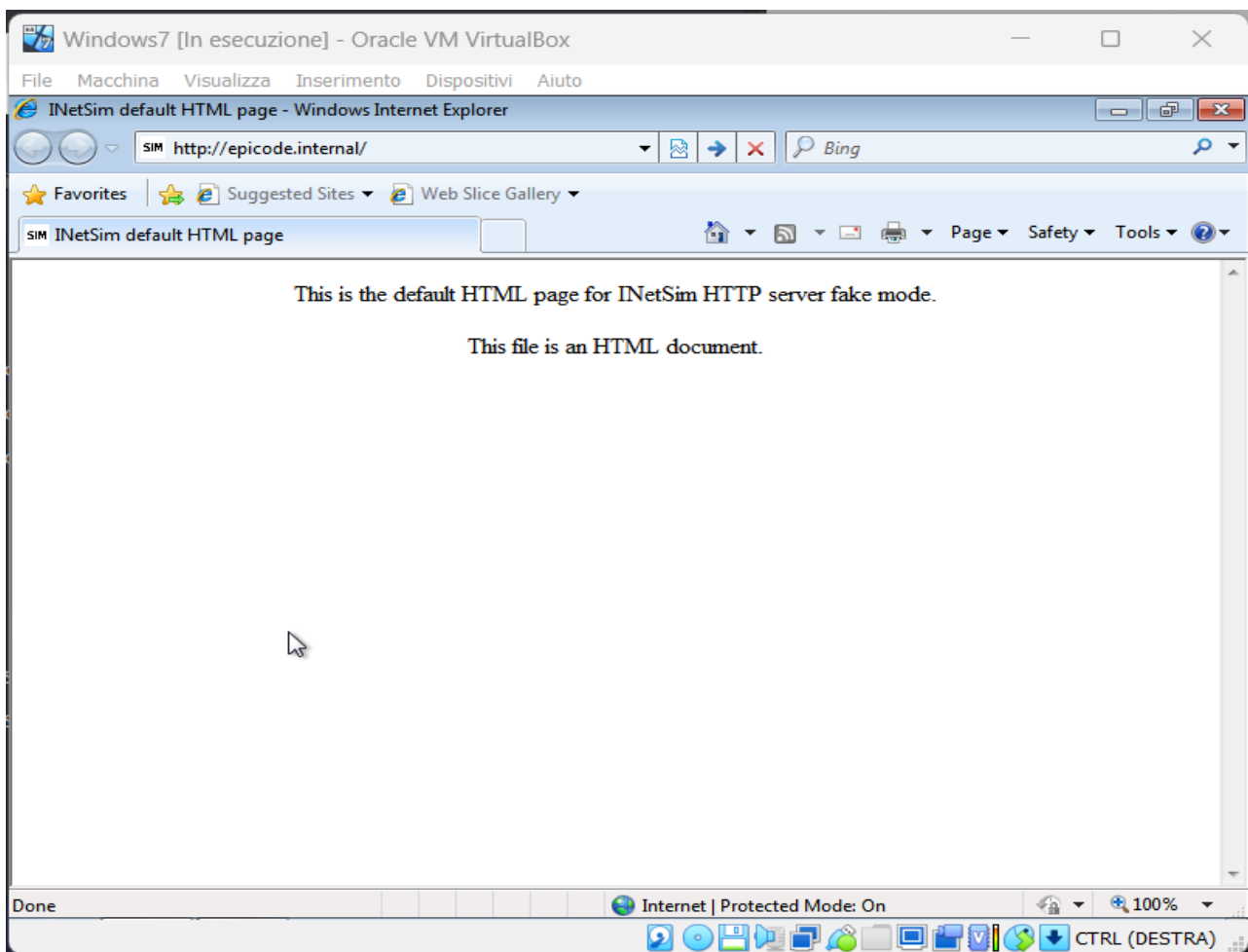
Queste sono dunque le caratteristiche delle due macchine:

Kali → IPv4 address **192.168.32.100** – Mac Address **08:00:27:b1:88:e2**

Windows7 → IPv4 Address **192.168.32.101** – Mac Address **08:00:27:a5:26:95**

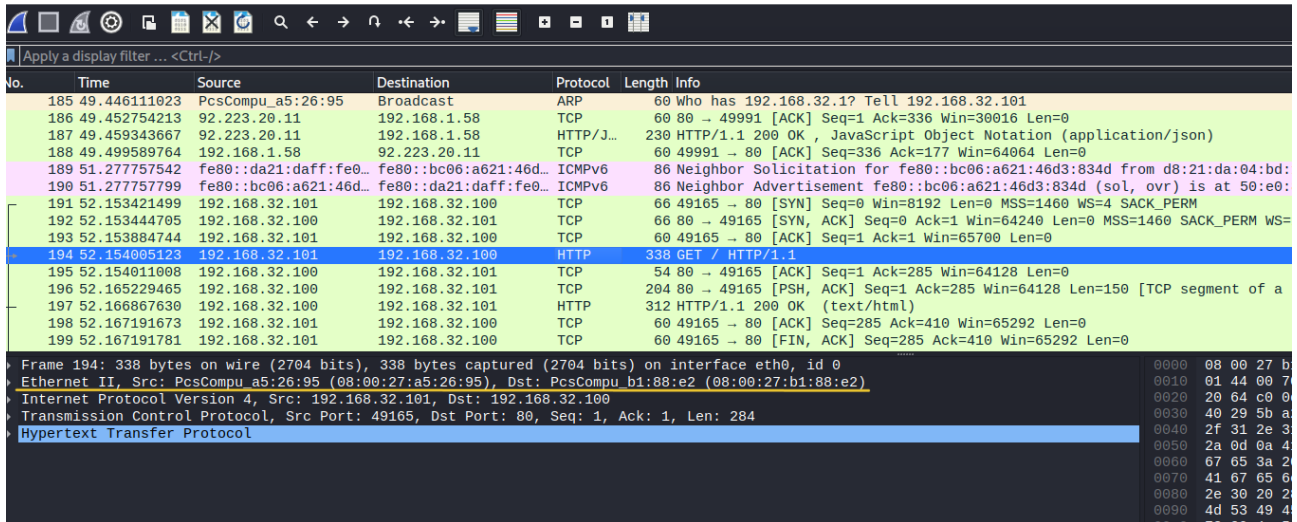
DNS server → IPv4 Address **192.168.32.100** - Stato: **ATTIVO**

Lo svolgimento dell'esercitazione consisteva nel presentare una dal client al server su protocollo https del dominio 192.168.32.100 nominato "epicode.internal".



Tramite il programma Wireshark era richiesto lo sniffing di dei Mac Address (sottolineati in giallo) di entrambe le macchine durante la trasmissione dei pacchetti.

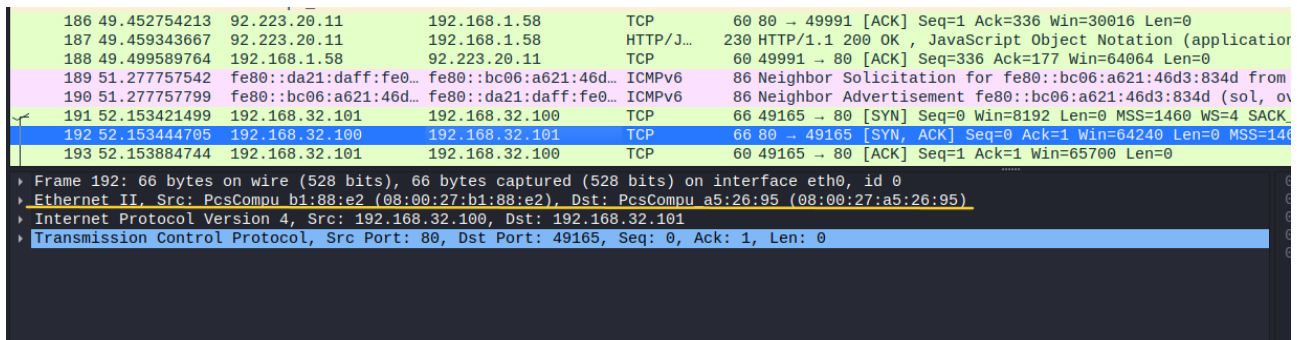
## HTTP



No.	Time	Source	Destination	Protocol	Length	Info
185	49.446111023	PcsCompu_a5:26:95	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
186	49.452754213	92.223.20.11	192.168.1.58	TCP	60	80 → 49991 [ACK] Seq=1 Ack=336 Win=30016 Len=0
187	49.459343667	92.223.20.11	192.168.1.58	HTTP/J...	230	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
188	49.499589764	192.168.1.58	92.223.20.11	TCP	60	49991 → 80 [ACK] Seq=336 Ack=177 Win=64064 Len=0
189	51.277757542	fe80::da21:daff:fe0...	fe80::bc06:a621:46d...	ICMPv6	86	Neighbor Solicitation for fe80::bc06:a621:46d3:834d from d8:21:da:04:bd:...
190	51.277757799	fe80::bc06:a621:46d...	fe80::da21:daff:fe0...	ICMPv6	86	Neighbor Advertisement fe80::bc06:a621:46d3:834d (sol, ovr) is at 50:e0:...
191	52.153421499	192.168.32.101	192.168.32.100	TCP	66	49165 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
192	52.153444705	192.168.32.100	192.168.32.101	TCP	66	80 → 49165 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=
193	52.153884744	192.168.32.101	192.168.32.100	TCP	60	49165 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
194	52.154005123	192.168.32.101	192.168.32.100	HTTP	338	GET / HTTP/1.1
195	52.154011008	192.168.32.100	192.168.32.101	TCP	54	80 → 49165 [ACK] Seq=1 Ack=285 Win=64128 Len=0
196	52.165229465	192.168.32.100	192.168.32.101	TCP	204	80 → 49165 [PSH, ACK] Seq=1 Ack=285 Win=64128 Len=150 [TCP segment of a
197	52.166867630	192.168.32.100	192.168.32.101	HTTP	312	HTTP/1.1 200 OK (text/html)
198	52.167191673	192.168.32.101	192.168.32.100	TCP	60	49165 → 80 [ACK] Seq=285 Ack=410 Win=65292 Len=0
199	52.167191781	192.168.32.101	192.168.32.100	TCP	60	49165 → 80 [FIN, ACK] Seq=285 Ack=410 Win=65292 Len=0

Frame 194: 338 bytes on wire (2704 bits), 338 bytes captured (2704 bits) on interface eth0, id 0  
Ethernet II, Src: PcsCompu a5:26:95 (08:00:27:a5:26:95), Dst: PcsCompu b1:88:e2 (08:00:27:b1:88:e2)  
Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100  
Transmission Control Protocol, Src Port: 49165, Dst Port: 80, Seq: 1, Ack: 1, Len: 284  
Hypertext Transfer Protocol

## HTTPS



No.	Time	Source	Destination	Protocol	Length	Info
186	49.452754213	92.223.20.11	192.168.1.58	TCP	60	80 → 49991 [ACK] Seq=1 Ack=336 Win=30016 Len=0
187	49.459343667	92.223.20.11	192.168.1.58	HTTP/J...	230	HTTP/1.1 200 OK , JavaScript Object Notation (applicationion
188	49.499589764	192.168.1.58	92.223.20.11	TCP	60	49991 → 80 [ACK] Seq=336 Ack=177 Win=64064 Len=0
189	51.277757542	fe80::da21:daff:fe0...	fe80::bc06:a621:46d...	ICMPv6	86	Neighbor Solicitation for fe80::bc06:a621:46d3:834d from
190	51.277757799	fe80::bc06:a621:46d...	fe80::da21:daff:fe0...	ICMPv6	86	Neighbor Advertisement fe80::bc06:a621:46d3:834d (sol, ov
191	52.153421499	192.168.32.101	192.168.32.100	TCP	66	49165 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK
192	52.153444705	192.168.32.100	192.168.32.101	TCP	66	80 → 49165 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=14
193	52.153884744	192.168.32.101	192.168.32.100	TCP	60	49165 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0

Frame 192: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface eth0, id 0  
Ethernet II, Src: PcsCompu b1:88:e2 (08:00:27:b1:88:e2), Dst: PcsCompu a5:26:95 (08:00:27:a5:26:95)  
Internet Protocol Version 4, Src: 192.168.32.100, Dst: 192.168.32.101  
Transmission Control Protocol, Src Port: 80, Dst Port: 49165, Seq: 0, Ack: 1, Len: 0

Veniva inoltre richiesto di evidenziare la differenza dei pacchetti soggetti a protocollo HTTP e HTTPS (i pacchetti trasmessi in http hanno il testo in "chiaro mentre quelli protocollati in https vengono criptati e sono quindi illeggibili).

Per ragioni pratiche si vedranno due screenshot per ogni pacchetto in modo da poter apprezzare le differenza con più facilità.

## HTTP

The image displays two screenshots of a Wireshark packet capture. The top screenshot shows a list of network packets, with packet 194 selected. This packet is an HTTP GET request from 192.168.32.101 to 192.168.32.100. The bottom screenshot provides a detailed view of the selected packet's structure. It shows the Ethernet II header, Internet Protocol Version 4 header, and the Hypertext Transfer Protocol (HTTP) details. The HTTP details pane is expanded, showing the request line 'GET / HTTP/1.1' and various headers including 'User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729)'. A yellow arrow points to the 'Host: 192.168.32.100' header.

Frame 194: 338 bytes on wire (2704 bits), 338 bytes captured (2704 bits) on interface eth0, id 0

Ethernet II, Src: PcsCompu\_a5:26:95 (08:00:27:a5:26:95), Dst: PcsCompu\_b1:88:e2 (08:00:27:b1:88:e2)

Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100

Transmission Control Protocol, Src Port: 49165, Dst Port: 80, Seq: 1, Ack: 1, Len: 284

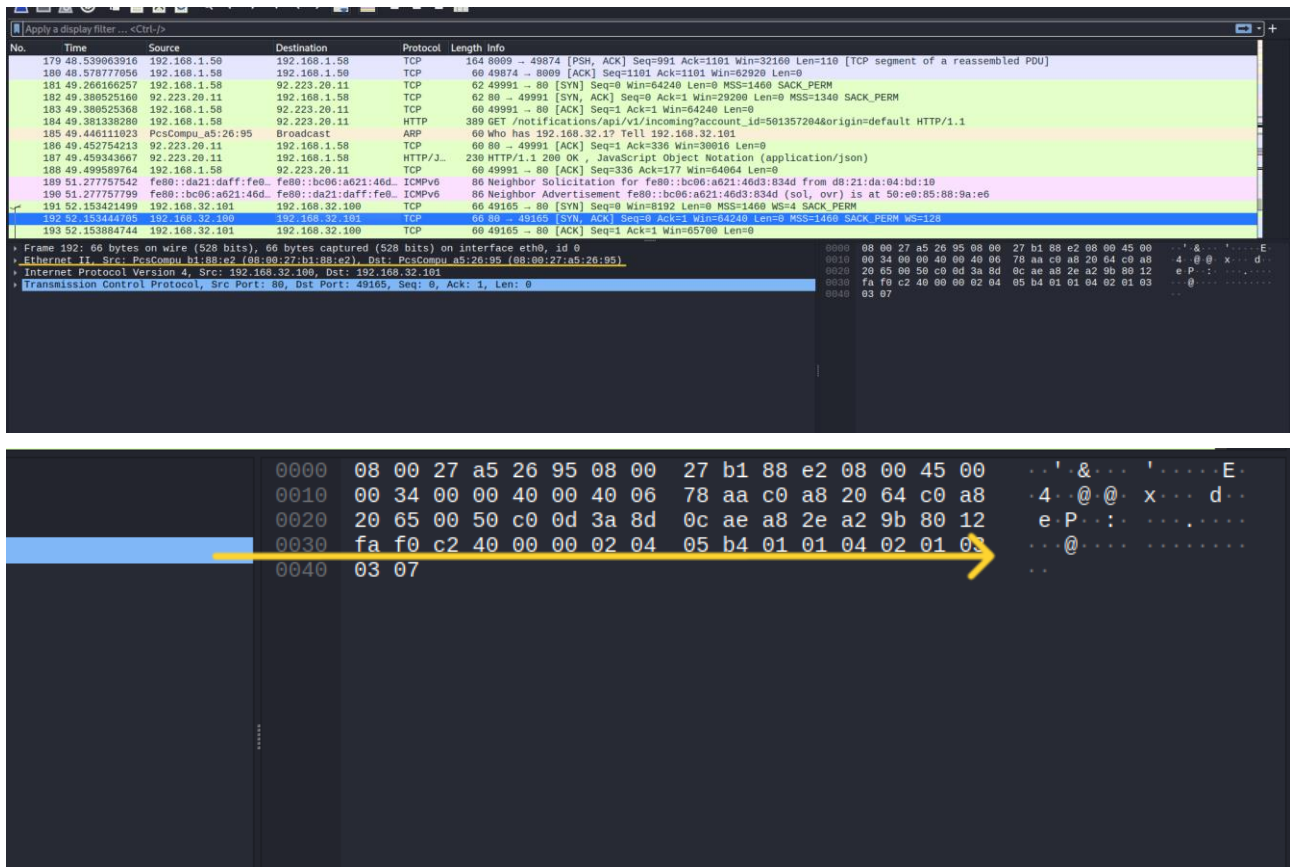
Hypertext Transfer Protocol

0000 08 00 27 b1 88 e2 08 00 27 a5 26 95 08 00 45 00 : .....&...E  
0010 01 44 00 70 40 00 00 00 37 2a c0 a8 20 65 c0 a8 : D p0...7...e  
0020 20 64 c0 a8 00 50 a8 2e a2 0b 3a 8d 0c af 50 1b : d P...: P  
0030 40 29 5b a2 00 00 47 45 54 20 2f 20 48 54 54 50 : 0)[-GE T / HTTP  
0040 2f 31 2e 31 0d 0a 41 63 63 65 70 74 3a 20 2a 2f : /1.1 Ac cept: \*/  
0050 2a 0d 0a 41 63 63 65 70 74 2d 4c 61 6e 67 75 61 : \*..Accep t-Langua  
0060 67 65 3a 20 65 6e 2d 55 53 0d 0a 55 73 65 72 2d : ge: en-U S..User-  
0070 41 67 65 6e 7a 3a 20 4d 6f 7a 69 6c 6c 61 2f 34 : Agent: M ozilla/4  
0080 2e 30 20 28 63 6f 6d 70 61 74 69 62 6c 65 3b 20 : .0 (comp atible;  
0090 4d 53 49 45 20 38 2e 30 3b 20 57 69 6e 64 6f 77 : MSIE 8.0 ; Window  
00a0 73 20 4e 54 20 36 2e 31 3b 20 57 4f 57 36 34 3b : s NT 6.1 ; WOW64;  
00b0 20 54 72 69 64 65 6e 74 2f 34 2e 30 3b 20 53 4c : Trident /4.0; SL  
00c0 43 43 32 3b 20 2e 4e 45 54 20 43 4c 52 20 32 2e : CC2; .NE T CLR 2.  
00d0 50 2e 33 30 37 32 37 30 20 2e 4e 45 54 20 43 4c : 0.50727; .NET CL  
00e0 52 20 33 2e 35 2e 33 30 37 32 39 3b 20 2e 4e 45 : R 3.5.30 729; .NE  
00f0 54 20 43 4c 52 20 33 2e 30 2e 33 30 37 32 39 29 : T CLR 3. 0.30729)  
0100 0d 0a 41 63 63 65 70 74 2d 45 6e 63 6f 64 69 6e : ..Accept -Encodin  
0110 67 3a 20 67 7a 69 70 2c 20 64 65 66 6c 61 74 65 : g: gzip, deflate  
0120 0d 0a 48 6f 73 74 3a 20 31 39 32 2e 31 36 38 2e : ..Host: 192.168.  
0130 33 32 2e 31 30 30 0d 0a 43 6f 6e 6e 65 63 74 69 : 32.100 . Connecti  
0140 6f 6e 3a 20 4b 65 65 70 2d 41 6c 69 76 65 0d 0a : on: Keep -Alive..  
0150 0d 0a

Packets: 223 · Displayed: 223 (100.0%) · Dropped: 0 (0.0%) Profile: Default



## HTTPS



Le differenze apprezzabili nelle immagini sopra sono dovute alla differenza tra i due protocolli utilizzati.

Il protocollo Hyper Text Transer Protocol, presente al settimo layer della tabella ISO/OSI, non prevede alcun tipo di cifratura del P.D.U. (che a questo livello prende il nome di Data).

Quando invece parliamo di HTTPS aggiungiamo Secure alla sigla del protocollo perché viene aggiunto un elemento di crittografia al Data in oggetto; questa aggiunta avviene, per esempio, tramite il protocollo SSL (Secure Socket Layer) che di trova al sesto livello della tabella ISO/OSI.