LABORATORIO DIDATTICO 1

Installazione e configurazione di base dell'analizzatore di protocollo Wireshark

L'analizzatore di protocollo è uno strumento software che, oltre ad essere impiegato nella diagnostica delle problematiche delle reti e nella sicurezza, è particolarmente indicato per comprendere appieno il concetto di protocollo e l'organizzazione di una suite di protocolli.
Nelle reti di telecomunicazioni sia geografiche (WAN, *Wide Area Network*) sia locali (LAN, *Local Area Network*) è molto spesso utile avere a disposizione degli strumenti che eseguano un'analisi dei molti protocolli implementati (strati OSI 2 ÷ 7) e che siano in grado di produrre statistiche sugli errori e valutazioni sulla reale percentuale di utilizzo delle risorse di rete e delle connessioni da parte delle applicazioni. A tale scopo si può utilizzare un *analizzatore di protocollo*.

In generale un analizzatore di protocollo può svolgere le seguenti funzioni:
- *decodifica dei protocolli*: l'analizzatore cattura i dati in transito e utilizza del software opportuno (noto come *decode engine*, motore di decodifica) per analizzare i frame scambiati (strato 2 OSI) e decodificare le intestazioni dei vari protocolli da essi trasportati (strati OSI 3 ÷ 7), nonché le informazioni effettivamente scambiate dai terminali, presentando il tutto in chiaro e/o in esadecimale; è così possibile verificare il contenuti degli header dei diversi protocolli, ricercando eventuali anomalie di configurazione o di funzionamento; il traffico catturato può essere salvato su disco per poter essere esaminato successivamente;
- *valutazione della percentuale di utilizzo di una rete o di una connessione e calcolo di tutta una serie di statistiche*, come per esempio frame errati, dimensioni dei frame, ecc.; ciò consente di valutare le reali prestazioni di una rete o di una connessione, nonché di identificare quali sono le applicazioni e i protocolli che impegnano maggiormente la rete;
- *filtraggio dei protocolli*: nelle moderne reti, in cui sono utilizzati molti protocolli, la possibilità di impiegare dei filtri è importante in quanto permette di selezionare quali protocolli devono essere analizzati e visualizzati, facilitando notevolmente la focalizzazione dei problemi.

Alcuni strumenti permettono anche di *generare traffico* allo scopo, per esempio, di «stressare» una connessione per verificarne la capacità trasmissiva e la rispondenza a eventuali condizioni contrattuali particolari (banda garantita, ecc.) stabilite con l'operatore di telecomunicazioni che fornisce la connessione stessa.
Anche se le loro funzionalità sono di massima quelle sopra citate, a seconda del campo di impiego gli analizzatori di protocollo possono essere suddivisi in due categorie.
- **Analizzatori per WAN**. Sono strumenti hardware, dotati di software opportuno e interfacciabili spesso con un PC portatile per la presentazione dei risultati e la memorizzazione dei dati catturati; essi sono inseriti in una connessione, per esempio tra un router e un modem e ne consentono l'analisi a tutti i livelli;
- **Analizzatori per LAN**. Possono essere di due tipi:
  - *analizzatori hardware (HW analyzer)*, costituiti spesso da un apparato esterno contente i circuiti e il software per catturare e analizzare i dati in transito, che può essere collegato a un PC per la memorizzazione e la visualizzazione dei dati;
  - *analizzatori software (SW analyzer)*, costituiti da un pacchetto software da installare su un PC collegato in rete, purché dotato di sufficiente potenza elaborativa.

Gli analizzatori software, grazie alla potenza elaborativa dei moderni PC, hanno prestazioni molto buone ed alcuni di essi sono anche gratuiti. Esamineremo quindi solamente questo tipo di tali analizzatori.
Vi sono diversi altri tipi di analizzatori software, che vanno da programmi di cattura e decodifica (denominati *sniffer*), a programmi che sono in grado di effettuare la ricerca degli host in rete e delle porte TCP/UDP attive (come *nmap*, scaricabile gratuitamente dal sito www.insecure.org), fino a sofisticati analizzatori che consentono di avere una visione completa della rete, in termini di efficienza degli apparati e dei protocolli utilizzati.
In questo testo si presenta l'analizzatore di protocollo *Wireshark*, scaricabile gratuitamente dal sito www.wireshark.org.

Analizzatore di protocollo Wireshark

Per utilizzare un analizzatore di protocollo software è necessario installare il relativo pacchetto software su un PC dotato di una scheda di rete (*NIC, Network Interface Card*) Ethernet o Wi-Fi. L'analizzatore pone la scheda di rete in *modalità promiscua* (*promiscous mode*), modalità nella quale la scheda non effettua il normale filtraggio dei frame sulla base degli indirizzi MAC (che consiste nell'accettare solo frame aventi come destinazione il proprio indirizzo MAC o i frame di broadcast), ma passa all'analizzatore tutti i frame che riceve, indipendentemente dal loro indirizzo MAC di destinazione.

In questo primo laboratorio didattico si illustrano l'installazione e le funzionalità di base dell'analizzatore di protocollo *Wireshark*.

L'analizzatore Wireshark[1] è costituito da un pacchetto software che è scaricabile gratuitamente dal sito www.wireshark.org (è disponibile sia per le distribuzioni Linux sia per i sistemi operativi Windows) e si installa come un normale pacchetto software. Dopo aver installato l'analizzatore di protocollo su un PC connesso in rete, nell'esempio con sistema operativo *Windows 7*, si carica l'analizzatore di protocollo cliccando sull'icona corrispondente.

Wireshark è un analizzatore di protocollo che permette analisi molto dettagliate e approfondite su un gran numero di protocolli sia della suite TCP/IP sia di altre suite di protocolli. Inizialmente è quindi consigliabile operare con le impostazioni di default e modificare solo ciò che serve per effettuare le analisi di base. Per addentrarsi nei dettagli è necessario conoscere approfonditamente i singoli protocolli, il che esula dallo scopo di questo testo. In FIGURA 1 è mostrata la schermata iniziale di Wireshark

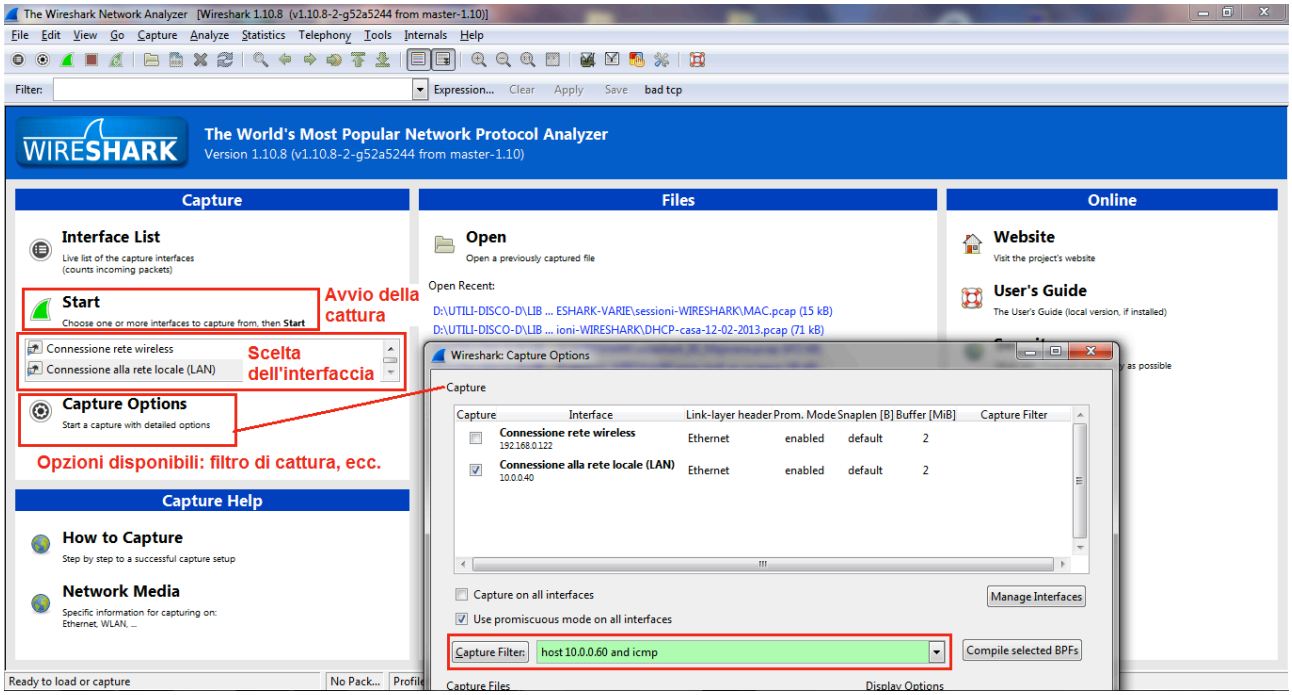

FIGURA 1 Schermata iniziale di Wireshark: possibilità di scelta dell'interfaccia, delle opzioni e di avvio della sessione di cattura.

Alcune indicazioni iniziali sono le seguenti.

- Tramite il menu *File* è possibile, tra l'altro, salvare su disco un'analisi effettuata o aprire un file di analisi precedentemente salvato per analizzarlo nuovamente.
- Tramite il menu *Edit* è possibile, tra l'altro configurare l'interfaccia di rete che viene utilizzata di default e abilitare la risoluzione degli indirizzi MAC, IP e dei *port number*, cliccando su *Edit -> Preferences*. Nella finestra che compare (FIGURA 2) clicchiamo su *Capture* e verifichiamo che l'interfaccia di default sia la scheda di rete tramite cui si è connessi e che sia abilitata la cattura[2] dei pacchetti in modalità promiscua (*promiscous mode*).
  Clicchiamo quindi su *Name Resolution* e abilitiamo (o disabilitiamo a seconda delle esigenze):

[1] Wireshark è l'evoluzione dell'analizzatore di protocollo Ethereal.

[2] In ambiente Windows Wireshark fa uso del pacchetto software WinPcap per catturare tutti i frame che giungono alla scheda di rete.

- *La risoluzione degli indirizzi MAC (o indirizzi Ethernet)*; l'indirizzo MAC è un indirizzo costituito 6 coppie di cifre esadecimali (48 bit) che identifica univocamente una scheda di rete Ethernet. L'indirizzo MAC viene assegnato alla scheda dal costruttore e le prime 3 coppie di cifre esadecimali identificano proprio il costruttore; gli indirizzi MAC della scheda sorgente, che emette un frame, e di quella destinazione (a cui è diretto il frame) vengono inseriti nell'header di ciascun frame Ethernet; tramite questa funzione è cosi possibile sapere quali sono i costruttori delle schede di rete Ethernet che inviano i frame che giungono al computer su cui è installato *Wireshark*;
- *La risoluzione degli indirizzi IP nei nomi host*; il protocollo IP identifica univocamente ogni interfaccia di rete tramite un indirizzo IP, per cui affinché possa avere luogo l'instradamento dei pacchetti IP ogni interfaccia di ogni macchina collegata in rete deve avere un indirizzo IP. Per le persone, però, è più semplice identificare le macchine tramite dei nomi. Allo stesso modo i siti Internet risiedono su macchine collegate in rete ed è necessario conoscere l'indirizzo IP della macchina (e dell'interfaccia) su cui risiede il sito. Quando digitiamo sulla barra degli indirizzi di un browser il nome di un sito o quando esploriamo le risorse di rete e vediamo i nomi delle macchine collegate in rete (*hostname*) operiamo in un modo che i computer non possono utilizzare. Il servizio DNS (*Domain Name System*), fornito da un *server DNS*, ha proprio il compito di restituire l'indirizzo IP associato a un nome oppure di fornire il nome di una macchina di cui si conosce l'indirizzo IP. Se si è connessi in rete ed è correttamente configurato il servizio DNS, abilitando questa funzione siamo in grado di conoscere i nomi delle macchine o dei siti che inviano o a cui sono diretti pacchetti IP (incapsulati in frame Ethernet) da/verso la nostra macchina; bisogna però considerare che questa opzione può determinare un aumento consistente del traffico verso il server DNS, per cui va impiegata con cautela;
- *La risoluzione dei port number nei nomi dei servizi*; un *port number* identifica *un protocollo* dello strato di applicazione che fornisce *servizi* (*server*) o che richiede *servizi* (*client*); poiché i port number dei servizi lato server sono in genere predefiniti, abilitando questa funzione siamo in grado di conoscere i nomi dei servizi (server) che inviano o a cui sono diretti i segmenti TCP/UDP che entrano o escono dalla nostra macchina (incapsulati in pacchetti IP) .

• Tramite il menu *View* è possibile scegliere, tra l'altro, quali barre di menu visualizzare e cosa visualizzare durante un'analisi: la successione dei frame (*Packet list*); il contenuto completo di un singolo frame (*Packet details*), con gli header di tutti i protocolli incapsulati; tutti i byte (compreso il contenuto informativo vero e proprio) che formano un frame e la loro codifica in esadecimale e in ASCII (*Packet Byte*) .

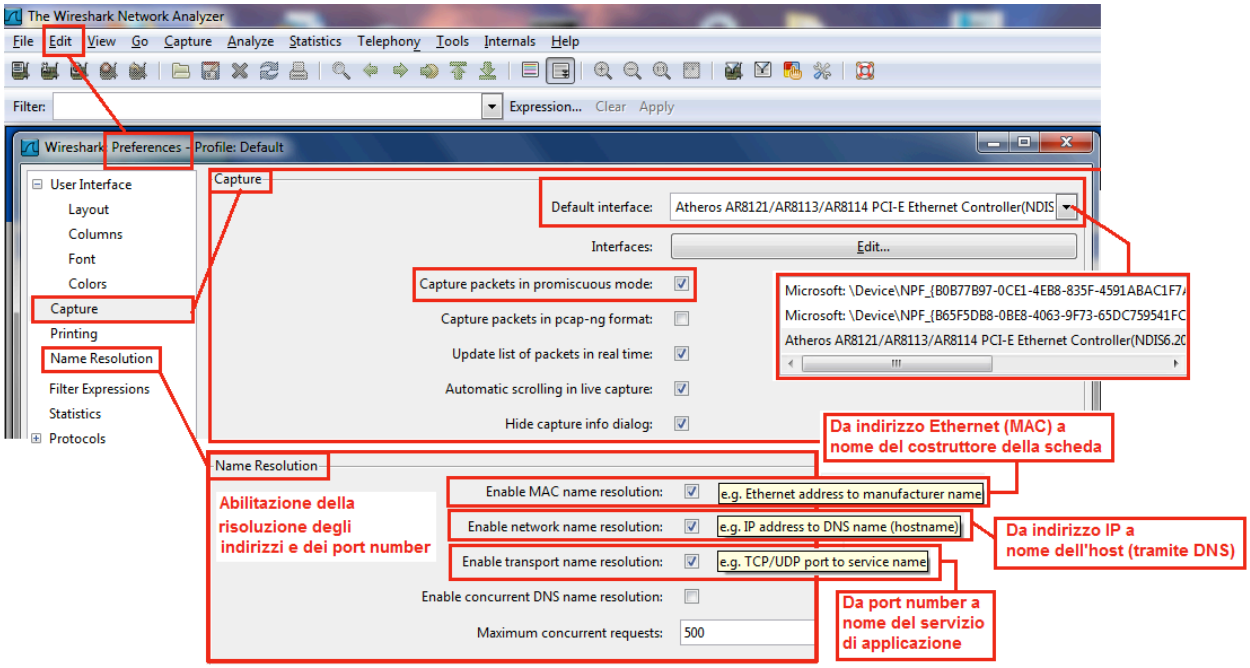

FIGURA 2 Scelta dell'interfaccia di default e abilitazione della risoluzione degli indirizzi.

Cattura ed analisi del traffico che viene ricevuto e trasmesso dalla scheda di rete

Cliccando su *(Capture) Start* è possibile far partire la cattura di tutti i frame che sono inviati o che giungono alla scheda di rete del computer su cui è installato Wireshark. Cliccando su (*Capture) Stop* la cattura si ferma ed è possibile passare alla fase di analisi. A questo punto sull'analizzatore compaiono, riportati in tre sottofinestre (dette *pane*), le seguenti informazioni:

- *prima sottofinestra*; riporta la successione dei frame catturati visualizzando (è possibile modificare le colonne da visualizzare tramite il menu View Displayed Columns):
  - il numero del frame (No.)
  - un'informazione relativa al tempo della cattura, valutata rispetto all'avvio della cattura stessa;
  - l'indirizzo IP o l'indirizzo MAC sorgente
  - l'indirizzo IP o l'indirizzo MAC destinazione
  - il protocollo di livello più elevato trasportato dal frame;
  - delle informazioni aggiuntive sulla funzione svolta dal frame, estrapolate da Wireshark sulla base del contenuto dei campi del protocollo di livello più elevato;
- *seconda sottofinestra*; riporta la struttura e il contenuto completo di un singolo frame selezionato nella prima finestra, nonché cliccando su [+], il contenuto di ciascun header dei protocolli trasportati dal frame stesso;
- *terza sottofinestra*; riporta in formato esadecimale (o binario) e ASCII l'intero contenuto del frame selezionato, comprese le informazioni trasportate; selezionando nella seconda finestra un protocollo o un campo di un protocollo compaiono evidenziati in blu nella terza finestra i valori in esadecimale e in ASCII dei bit che lo compongono.

Per esempio in FIGURA 3 si evidenzia nella prima sottofinestra la successione dei frame catturati, nella 2a sottofinestra la composizione del frame N. 581, selezionato sulla prima finestra, mentre la terza finestra mostra il contenuto in esadecimale ed in ASCII della riga selezionata sulla 2a finestra. L'analisi della seconda finestra mostra anche l'incapsulamento dei protocolli in quanto, partendo dal basso, si rileva che la PDU del protocollo HTTP con la quale si richiede una pagina web (www.google.it) viene incapsulata in un segmento TCP avente come porta di destinazione (*Dst Port*) la porta 80, che identifica il lato server del servizio (protocollo) HTTP; a sua volta il segmento TCP viene incapsulato in un pacchetto IP in cui sono contenuti l'indirizzo IP del mittente e del destinatario (avendo abilitato la risoluzione degli indirizzi Wireshark mostra l'hostname di destinazione); infine il pacchetto IP è incapsulato in un frame Ethernet nel cui header sono presenti gli indirizzi MAC della scheda di rete sorgente e di quella destinazione (è quella del router tramite cui si accede a Internet dalla LAN); anche qui Wireshark mostra i nomi dei costruttori delle schede.

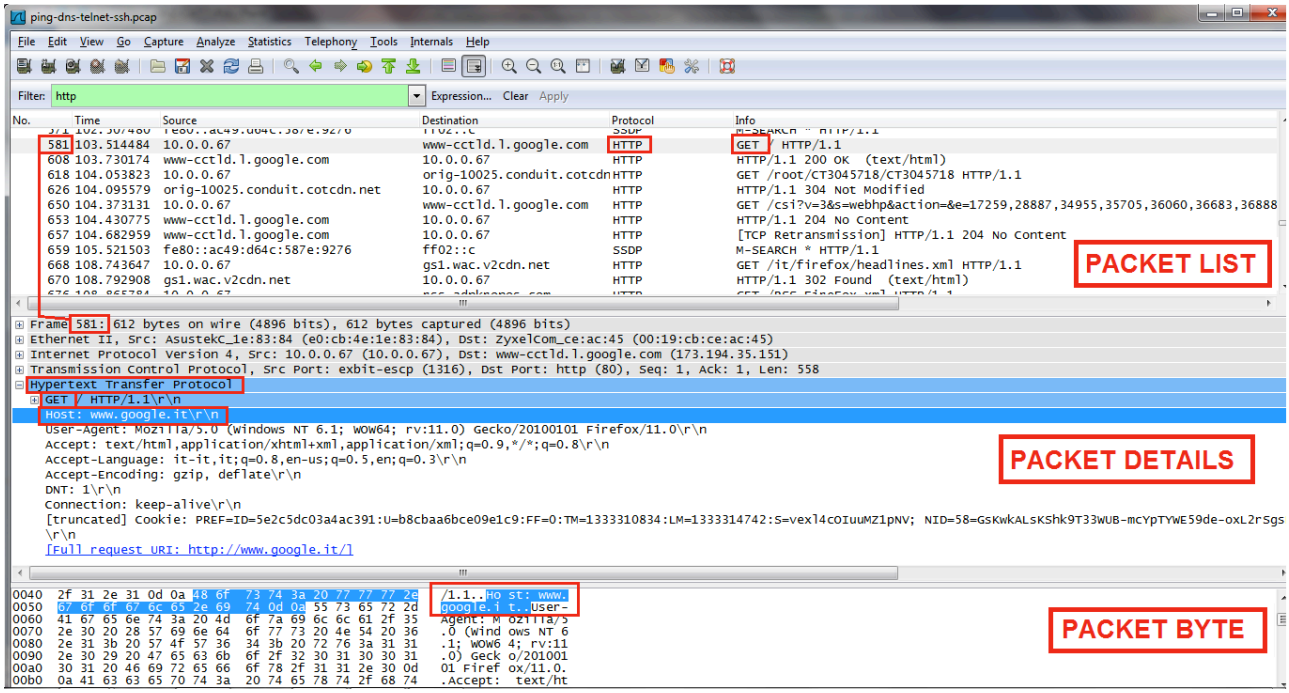

FIGURA 3 Esempio di traffico catturato.

Filtri di visualizzazione e di cattura
Normalmente quando si lancia la cattura del traffico vi sono diversi protocolli che non rivestono alcun interesse ai fini di una certa analisi, come per esempio lo Spanning-Tree Protocol, il Netbios, l'ARP, ecc., ma che al contrario possono rendere l'analisi più difficoltosa. Per ovviare a questo inconveniente è possibile effettuare due tipi di filtraggi: *a) filtraggio sulla visualizzazione*; b) *filtraggio sulla cattura*.

a) Filtraggio sulla visualizzazione
Si ottiene inserendo un *Display Filter* che permette di *restringere la visualizzazione* ai soli protocolli o campi dell'header dei protocolli e/o agli host di interesse. Il filtro può essere inserito dal menu *Analyze, Display Filter* (o cliccando su *Filter* nella barra dei menu). Si apre una finestra (FIGURA 3) in cui è possibile inserire una stringa (in *minuscolo*) detta *Filter string*, che definisce le caratteristiche del filtro, ed assegnare un nome al filtro (*Filter name*). Cliccando su *New* il filtro viene salvato, mentre lo si applica cliccando su *Apply* o OK. Nella creazione della *Filter string* è utile cliccare su *Expression* per avere l'elenco dei parametri (o *primitive*), indicati come *Field name* (nomi dei protocolli con i relativi campi), che è possibile inserire, nonché quella degli operatori che possono essere espressi in una notazione simile a quella del linguaggio C oppure con un'abbreviazione inglese (TABELLA 1).

TABELLA 1 Principali operatori utilizzabili nelle Filter string dei Display filter.

Operatore	Notazione C	Notazione inglese	Esempio
Uguale	==	eq	ip.addr eq 10.0.0.1 ip.addr == 10.0.0.1
Diverso	!=	ne	telnet.data ne "password" telnet.data != "password"
And	&&	and	ip.addr eq 10.0.0.1 and tcp.port eq 80 ip.addr == 10.0.0.1 && tcp.port == 80
Or	||	or	tcp.port eq 80 or udp.port eq 53 tcp.port == 80 || udp.port == 53
Not	!	not	tcp.port eq 80 and not (ip.addr eq 10.0.0.1) tcp.port == 80 && ! (ip.addr == 10.0.0.1)

In alternativa (FIGURA 5) la *Filter string* può essere digitata direttamente nella barra *Filter*, cliccando poi su *Apply*, nonché salvata sulla barra stessa cliccando su *Save*. Il filtro può essere rimosso cliccando su *Clear*. Per rimuovere il filtro salvato dalla barra si agisce su *Edit -> Preferences -> Filter Expressions -> Remove*.

b) **Filtraggio sulla cattura**
Si ottiene inserendo un *Capture Filter* sull'interfaccia selezionata, che permette di *non catturare* il traffico generato da determinati protocolli e/o host. Il filtro può essere creato, in modo analogo ai *Display Filter*, tramite il menu *Capture -> Capture Filters* ed inserito nelle *Capture Options* dell'interfaccia usata, cliccando su *Capture ->Options* e sul nome dell'interfaccia che si impiega (FIGURA 6). Si apre una finestra in cui cliccando su *Capture Filters* compare l'elenco dei filtri che è possibile inserire. Il filtro è definito da una stringa (*Filter string*) contenente una o più *primitive* con dei connettivi avente il seguente formato:
[not] *primitive* [and|or [not] *primitive* ...].
Esempi di primitive sono:

- **[src|dst] host <ind. IP>**, indica un'interfaccia di rete (host) identificata da un indirizzo IP; può essere *preceduto* da **src** o **dest** per indicare un host sorgente o destinazione, per esempio "*host 10.0.0.200*" abilita la cattura dei soli pacchetti diretti o provenienti dall'host 10.0.0.200 (si vedano i filtri predefiniti per altri esempi);
- **[tcp|udp] [src|dst] port (<** *port number (o protocollo di applicazione associato))*>; permette di catturare il traffico solo dalla porta TCP o UDP specificata, come per esempio "*port (80)"* (o "*port http")* per catturare il traffico http; "*udp port (domain)*" o "*udp port (53)*" per catturare il traffico DNS, ecc.
- **ether [src|dst] host <indirizzo Ethernet (MAC)>** indica un'interfaccia di rete (host) identificata da, detto anche *indirizzo Ethernet* detto anche un indirizzo MAC o indirizzo fisico; *host* può essere

preceduto da *src* o *dst* per indicare un host sorgente o destinazione, per esempio "*ether src host e0:cb:4e:1e:83:84*" abilita la cattura dei soli frame provenienti dalla scheda di rete (host) avente indirizzo Ethernet (MAC) *e0:cb:4e:1e:83:84* (si vedano i filtri predefiniti per altri esempi);

- **[src|dst] net <net> [mask <subnet mask>]**; indica una rete IP identificata da un indirizzo IP di rete ed eventualmente da una subnet mask espressa in notazione decimale puntata (si veda il capitolo 4 per i dettagli); per esempio "net 10.0.0.0 mask 255.255.255.0" permette di catturare solo il traffico da/verso la rete IP caratterizzata dall'indirizzo IP 10.0.0.0 con subnet mask 255.255.255.0.
- **ether|ip broadcast|multicast**; permette di catturare solo il traffico di tipo broadcast (diretto a tutti) o multicast (diretto a un gruppo specifico di host) a livello Ethernet oppure a livello IP.

Esempi di *Filter string*

- Filtri di visualizzazione:
  "http"; visualizza tutto il traffico generato dal protocollo *http* (FIGURA 3)
  "*telnet*"; visualizza tutto il traffico generato dal protocollo *telnet* (FIGURA 4A)
  "*telnet and ip.src eq (oppure = =) 10.0.0.67*"; visualizza solo il traffico *telnet* generato dall'*host* (src = sorgente) avente indirizzo IP 10.0.0.67 (FIGURA 4B)
  "*ssh and ip.dst eq 10.0.0.250*"; visualizza solo il traffico del protocollo *ssh* destinato all'host (dst = destinazione) avente indirizzo IP 10.0.0.250 (FIGURA 5)
  "*ssh and not (ip.addr eq 10.0.0.1)*"; visualizza il traffico del protocollo *ssh* tranne quello da/verso l'host avente indirizzo IP 10.0.0.1.
- Filtro di cattura:
  "*src host 10.0.0.200 and tcp dst port(80)*" o " *src host 10.0.0.200 and tcp dst port(http)"*; cattura solo il traffico generato dall'host (scr = sorgente) 10.0.0.200 e diretto (dst = destinazione) alla porta (80) TCP, associata al lato server del protocollo *http* (FIGURA 6).

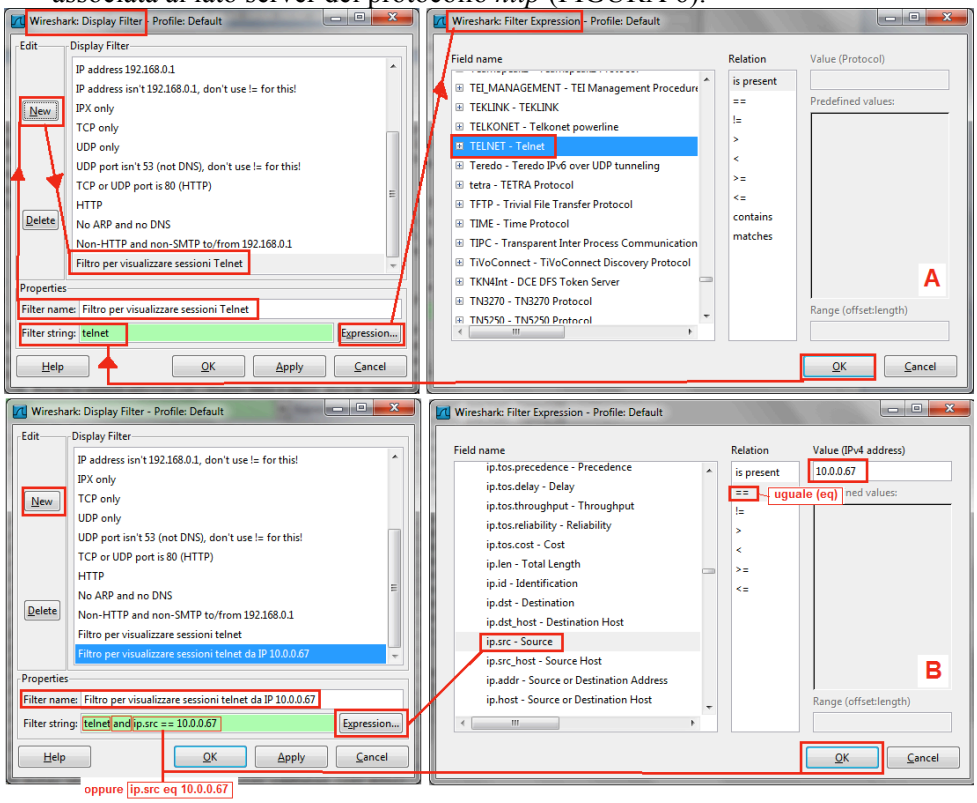

FIGURA 4 Esempi di Display Filter (Filtri sulla visualizzazione): A) tutto il traffico telnet; B) solo il traffico telnet generato dall'host con indirizzo IP 10.0.0.67

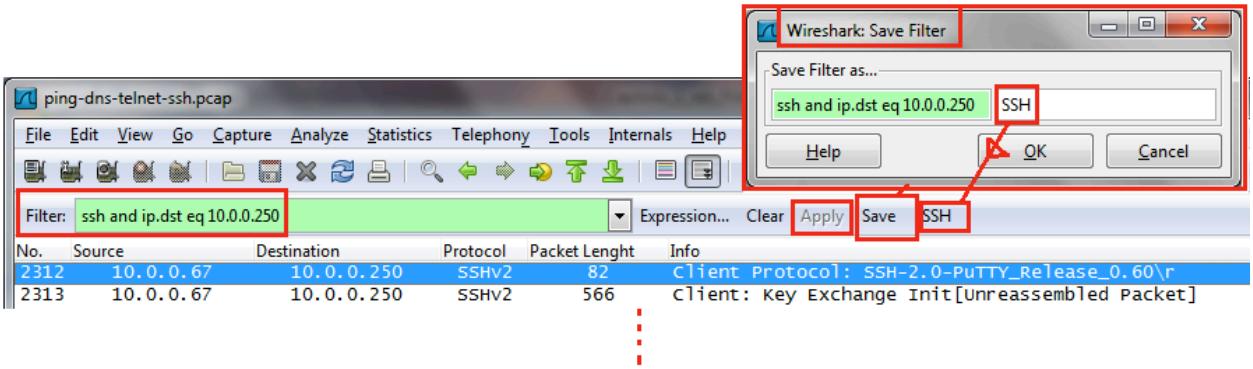

FIGURA 5 Inserimento di un filtro dalla barra Filter e suo salvataggio

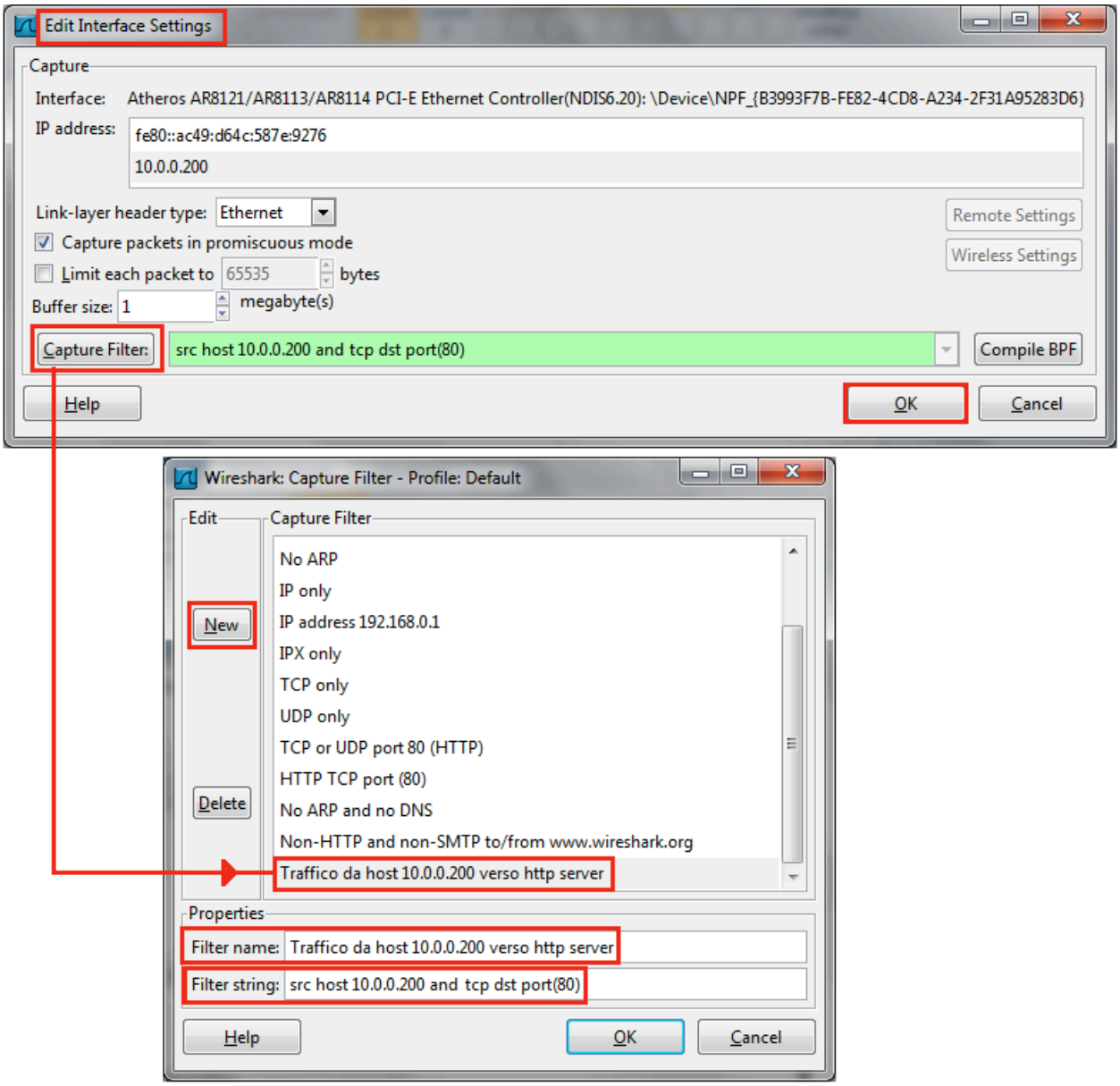

FIGURA 6 Definizione di un filtro che cattura solo il traffico generato dall'host 10.0.0.200 e diretto a un server http in ascolto sulla porta 80 TCP e suo inserimento nella configurazione dell'interfaccia.