Università di Verona, Facoltà di Scienze MM.FF.NN.

Insegnamento di Reti di Calcolatori

Strumenti e protocolli di rete all'opera (parte 1)

Davide Quaglia

Scopo di questa esercitazione è imparare l'utilizzo di tool software per analizzare il traffico di una rete osservando il comportamento dei vari protocolli.

Preambolo: Analisi della configurazione di rete

Per visualizzare le impostazioni di rete del proprio PC con Linux occorre dare il comando:

```
$ /sbin/ifconfig -a
```

In Windows invece occorre dare il comando:

```
c:\> ipconfig /all
```

Tale comando visualizza la lista delle interfacce di rete e, per ognuna, le impostazioni MAC e IP. L'interfaccia ethernet viene indicata con `eth0` e si può osservare l'indirizzo MAC (6 byte esadecimali).

1 Analizzatori di protocollo

Esistono strumenti software chiamati *Network Protocol Analyzer* o *sniffer* che consentono di analizzare tutti i pacchetti che arrivano alla/e propria/e interfaccia di rete. L'analisi consiste nell'ispezione bit per bit di ciascun pacchetto e nel calcolo di statistiche sull'insieme dei pacchetti raccolti. Tali tool permettono anche di impostare filtri per limitare la quantità di pacchetti raccolti. I toll più usati sono:

- ***tcpdump*** : è un *sniffer* a linea di comando per Linux.
- ***wireshark*** (ex ***ethereal***) : le funzionalità che offre sono molto simili a quelle di tcpdump, ma in più è dotato di un'interfaccia grafica e di più funzionalità di ordinamento e filtraggio. Disponibile per Windows, Linux, MAC OS X, Solaris.
- ***tethereal*** : è una versione testuale di wireshark per Linux.
- ***analyzer***: implementazione di sniffer in ambiente Windows.
- ***windump***: implementazione di sniffer in ambiente Windows.

Questi tool sono scaricabili da:

- **tcpdump** http://www.tcpdump.org
- **ethereal (wireshark)** http://www.wireshark.org
- **thethereal** http://www.ethereal.com
- **analyzer** http://analyzer.polito.it
- **windump** http://www.winpcap.org/windump

Questi tool di analisi si basano tutti sulla libreria C chiamata ***libpcap*** (***winpcap*** nel caso di Windows). Questa libreria è nata intorno al 1993 all'Università della California. È supportata pienamente dalla comunità opensource ed è reperibile al sito http://www.tcpdump.org (tcpdump è lo sniffer per

eccellenza che sfrutta la libpcap).
Le principali funzioni di questa libreria sono la possibilità di trovare le interfacce di rete della propria macchina, gestire potenti filtri di cattura, analizzare pacchetto per pacchetto, e ottenere statistiche sulle catture.

Il seguito dell'esercitazione si focalizzerà su Tcpdump e Wireshark; per approfondimenti sui comandi e sulle opzioni dee vari tool si faccia riferimento a:

- **tcpdump** http://www.tcpdump.org/tcpdump_man.html
- **ethereal** (**wireshark**) http://www.wireshark.org/docs/
- **thethereal** http://www.ethereal.com/docs/man-pages/tethereal.1.html

1.1 Scaricamento, installazione e utilizzo di Wireshark

Wireshark si scarica liberamente dal sito http://www.wireshark.org o probabilmente è presente già nell'installazione della propria distribuzione Linux.

ATTENZIONE: per poter utilizzare le funzionalità di cattura di un analizzatore di protocollo in ambiente Linux bisogna essere autenticati come utente *root* o aver installato il tool con *setuid* a root. In alternativa Wireshark può essere utilizzato per analizzare catture precedentemente effettuate da utente root e salvate su file nel formato standard *pcap*.

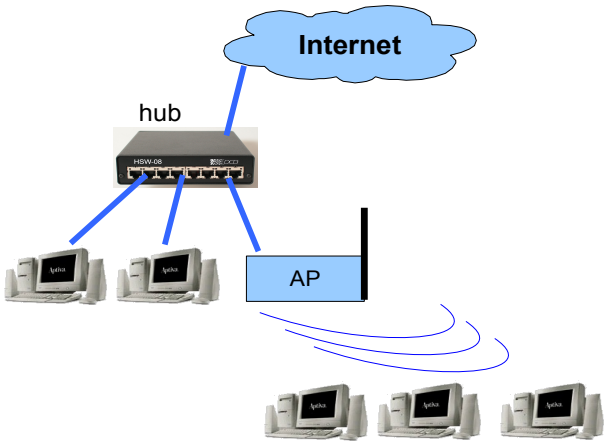
1.2 Alcuni concetti sullo *sniffing*

Sniffing in **reti non-switched**: In questo tipo di reti locali il mezzo trasmissivo è condiviso o per la presenza di un *hub* o perché la trasmissione è wireless; su un mezzo condiviso tutte le schede di rete dei computer nella rete locale ricevono tutti i pacchetti, anche quelli destinati ad altri, ma prendono in considerazione solo i propri analizzando l'indirizzo MAC di destinazione. Lo sniffing in questo caso consiste nell'impostare sull'interfaccia di rete la cosiddetta *modalità promiscua*, che disattiva il “filtro hardware” basato sul MAC permettendo al sistema l'ascolto di tutto il traffico passante sul mezzo fisico.

Sniffing in **reti ethernet switched**: In questo caso l'apparato centrale della rete, definito switch, si preoccupa, dopo un breve transitorio, di inoltrare su ciascuna porta solo il traffico destinato al dispositivo collegato a quella porta; ciascuna interfaccia di rete riceve, quindi solo i pacchetti destinati al proprio indirizzo, i pacchetti multicast e quelli broadcast. L'impostazione della modalità promiscua è quindi insufficiente per poter intercettare il traffico in una rete gestita da switch. Un metodo per poter ricevere tutto il traffico dallo switch da una porta qualunque è il **MAC flooding**. Tale tecnica consiste nell'inviare ad uno switch pacchetti appositamente costruiti per riempire la tabella dello switch di indirizzi MAC fittizi. Questo attacco costringe lo switch ad entrare in una condizione detta di *fail open* che lo fa comportare come un hub, inviando così gli stessi dati a tutti gli apparati ad esso collegati.

2 Analisi di una rete

Analizzando i pacchetti uno per uno si possono scoprire cose molto interessanti sui servizi e sulle macchine presenti in quel momento in rete. Nel seguito si farà riferimento alla seguente configurazione di rete:



2.1 Utilizzo di tcpdump

Questo applicativo è un tool di cattura, attraverso il quale è possibile monitorare il traffico in una rete. Il tool permette, tra l'altro, di limitare la cattura dei pacchetti impostando dei filtri basati, ad esempio, sull'interfaccia di ascolto, sul protocollo o sulla porta utilizzata. Sono inoltre disponibili una serie di opzioni, in particolare vi è la possibilità di limitare il numero di pacchetti catturati o quanti byte acquisire per ciascun pacchetto. Inoltre è possibile salvare su file i pacchetti catturati per leggerli con lo stesso programma in un secondo tempo.

Si riportano alcuni esempi di utilizzo; si noti che per catturare pacchetti occorre lanciare il programma dall'utente root mentre per analizzare file di catture precedenti si può essere utenti non privilegiati.

Utilizzo con le impostazioni di default e senza salvare i pacchetti catturati (vengono solo visualizzati):

```
$ /usr/sbin/tcpdump
```

Analizziamo il comando con le varie opzioni utilizzate:

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.cap -s 0
```

-i eth0 : rappresenta l'interfaccia dalla quale si intende catturare

-w eth0.cap : rappresenta il nome del file in cui verranno messi i pacchetti catturati

-s0 : la flag *-s* permette di specificare quanti byte acquisire da ogni singolo pacchetto (default 68); con *-s 0* si richiede di acquisire l'intero pacchetto

Per interrompere il monitoraggio occorre utilizzare la combinazione *CTRL + C*.

Altri esempi di utilizzo di tcpdump:

1) viene selezionata l'interfaccia *eth0* e catturato il flusso da e verso *edalab-srv01.sci.univr.it*, scrivendo sul file *eth0.cap*

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.log host edalab-srv01.sci.univr.it
```

2) come il precedente esempio ma in questo caso vengono letti solo i pacchetti IP contenenti TCP.

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.cap ip proto tcp
```

3) come il precedente esempio ma in questo caso vengono letti solo i pacchetti IP contenenti TCP da e verso la porta 80 (http).

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.cap -s 0 tcp port 80
```

4) come il precedente ma in questo caso vengono letti solo i pacchetti IP contenenti TCP da e verso la porta 80 (http) dell'host *edalab-srv01.sci.univr.it*.

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.cap \
  ip proto tcp host edalab-srv01.sci.univr.it \
  and port 80
```

4) come il precedente esempio ma in questo caso vengono letti solo i pacchetti IP contenenti TCP *verso* la porta 80 dell'host *edalab-srv01.sci.univr.it (dst* sta per *destination* mentre *src* è *source)*

```
$ /usr/sbin/tcpdump -i eth0 -w eth0.cap \
  ip proto tcp \
  dst host edalab-srv01.sci.univr.it \
  and port 80
```

Le capacità di analisi e filtraggio vanno ben oltre questi esempi introduttivi, si consiglia di far riferimento alle pagine del manuale per verificarne le potenzialità.

5) come il precedente esempio ma in questo caso vengono visualizzati i pacchetti precedentemente salvati nel file eth0.cap; anche in questo caso si può applicare un filtro che permette di visualizzare solo i pacchetti di interesse

```
$ /usr/sbin/tcpdump -r eth0.cap host edalab-srv01.sci.univr.it
```

Si ricorda che non serve essere root quando Tcpdump legge un file precedentemente catturato.

Si può analizzare il contenuto di una precedente cattura e redirigere l'output su un file di testo da aprire con il proprio editor preferito.

```
$ /usr/sbin/tcpdump -e -vvv -r eth0.cap > out.txt
$ kedit out.txt
```

L'output mostra una descrizione di ogni pacchetto catturato. Ogni descrizione contiene (si veda come esempio il primo pacchetto):

- l'istante di cattura (es. 09:22:23.490424)
- header di livello 2
  - MAC sorgente (es. 00:11:43:a7:19:fa)
  - MAC destinazione (es. bradcast cioè ff:ff:ff:ff:ff:ff)
  - Protocol Type (es. IPv4 cioè 0x0800)
- header di livello 3
  - campi vari (es. tos=0x00, ttl=64, id=60927, offset=0, flags [none], proto: UDP (17), length=223)
  - sorgente IP (es. 157.27.242.154)

    - destinazione IP (es. 157.27.242.255)
- header di livello 4
    - porte sorgente e destinazione accorpate per comodità ai corrispondenti IP (es. netbios-dgm)
    - verifica della checksum
- header di livello applicazione (es. Netbios, protocollo di condivisione di dischi in rete).

Un modo per diminuire il volume di pacchetti visualizzati è impostare un filtro di lettura su un certo indirizzo MAC (sorgente o destinazione):

```
$ /usr/sbin/tcpdump -e -vvv -r eth0.cap ether host 00:11:43:3E:98:DB > mac.txt
```

Se invece si vuole isolare solo i pacchetti IP con un certo indirizzo IP (sorgente o destinazione) si usa:

```
$ /usr/sbin/tcpdump -e -vvv -r eth0.cap ip host 157.27.252.10 > host.txt
```

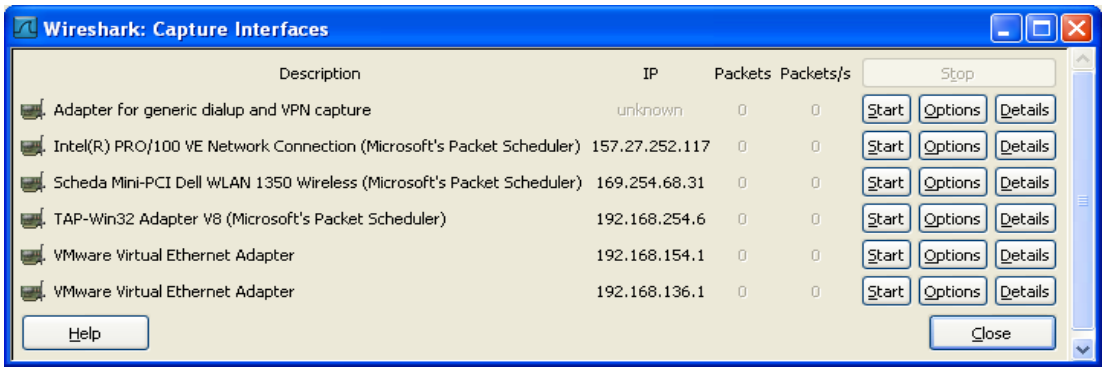
con il comando

```
$ /usr/sbin/tcpdump -X -r cattura.cap ip host 157.27.252.10 > myhost.txt
```

è possibile esaminare anche il contenuto dei pacchetti.

2.2 Utilizzo di Wireshark

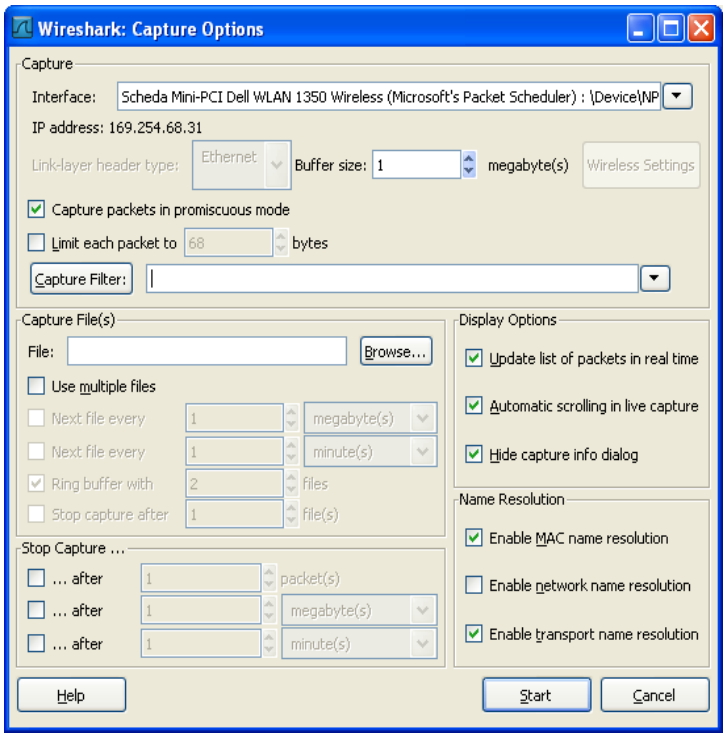
Per iniziare una cattura occorre scegliere *Interfaces* dal menu *Capture*. Apparirà la seguente finestra di dialogo in cui sono elencate tutte le interfacce di rete della propria macchina. Per ogni interfaccia di rete è possibile impostare le opzioni di cattura con il bottone *Options* e avviare la cattura con il bottone *Start*.



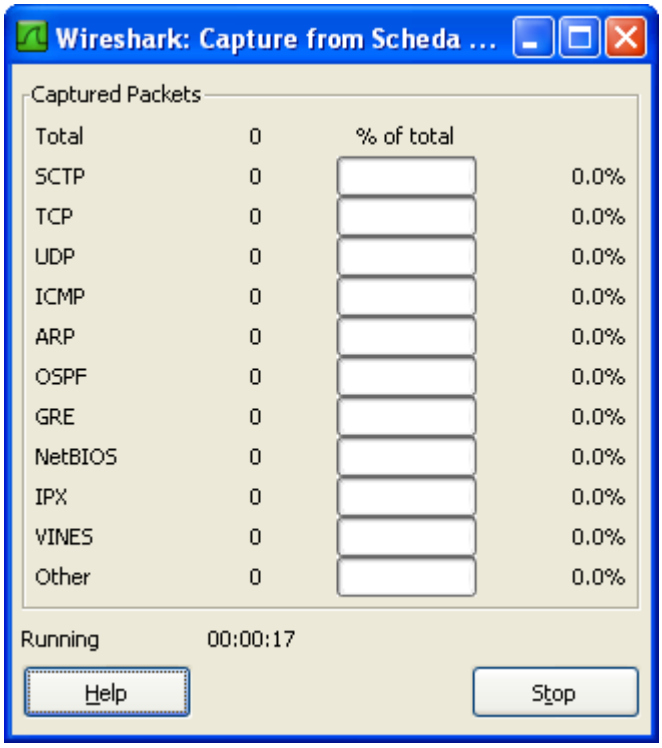
Il bottone *Options* fa apparire la finestra di dialogo delle opzioni:

- *Interface*. E' possibile cambiare l'interfaccia di cattura agendo su questa lista.
- *Link Layer Header Type*. Tale controllo indica al tool quale protocollo datalink assumere per interpretare i bit catturati; siccome la scelta dell'interfaccia di rete ha già definito questo aspetto, il controllo appare disattivato.
- *Buffer*. Permette di impostare la dimensione dell'area di memoria che il sistema operativo utilizza per parcheggiare I pacchetti prima di scriverli su file. Se durante la cattura il tool avverte di pacchetti persi occorre aumentare questa dimensione.
- *Promiscuous mode*. Abilita/disabilita la cattura in modalita promiscua.
  ATTENZIONE. Certe schede di rete non supportano tale modalita e in tal caso tale opzione non ha effetto.
- *Limit packet size*. Cattura solo N byte di ciascun pacchetto per evitare di creare un file troppo grosso. Spesso infatti la maggior parte degli header contenuti (fino al livello application) non supera i 70 byte circa.

- *Capture Filter*. E' possibile richiamare uno dei possibili filtri di cattura pre-definiti oppure crearne uno nuovo con la sintassi di Tcpdump.
- *Capture File*. Nome del file in cui verranno salvati i pacchetti. E' consuetudine dare a questo file estensione .cap per ricordarsi del formato in esso contenuto.
- *Stop capture*. E' possibile far terminare la cattura dopo N pacchetti o byte o sec/minuti/ore/giorni.
- *Display Options*. Permette di decidere se visualizzare la lista dei pacchetti catturati durante l'acquisizione; la visualizzazione puo' essere difficile se la velocita di cattura è alta.
- *Name resolution*. Permette di visualizzare gli indirizzi contenuti negli header in maniera piu' esplicativa. Per gli indirizzi MAC i primi tre byte (OUI) possono essere sostituiti dal nome dell'organizzazione. Per gli indirizzi IP si puo' ottenere il nome Internet corrispondente. Per gli indirizzi trasporto si puo' ottenere il nome del protocollo di livello applicazione (ad es. 80 → HTTP).
  ATTENZIONE. La risoluzione degli indirizzi IP richiede al tool di comunicare con il server DNS e questo potrebbe "sporcare" la cattura con i pacchetti di comunicazione col DNS.

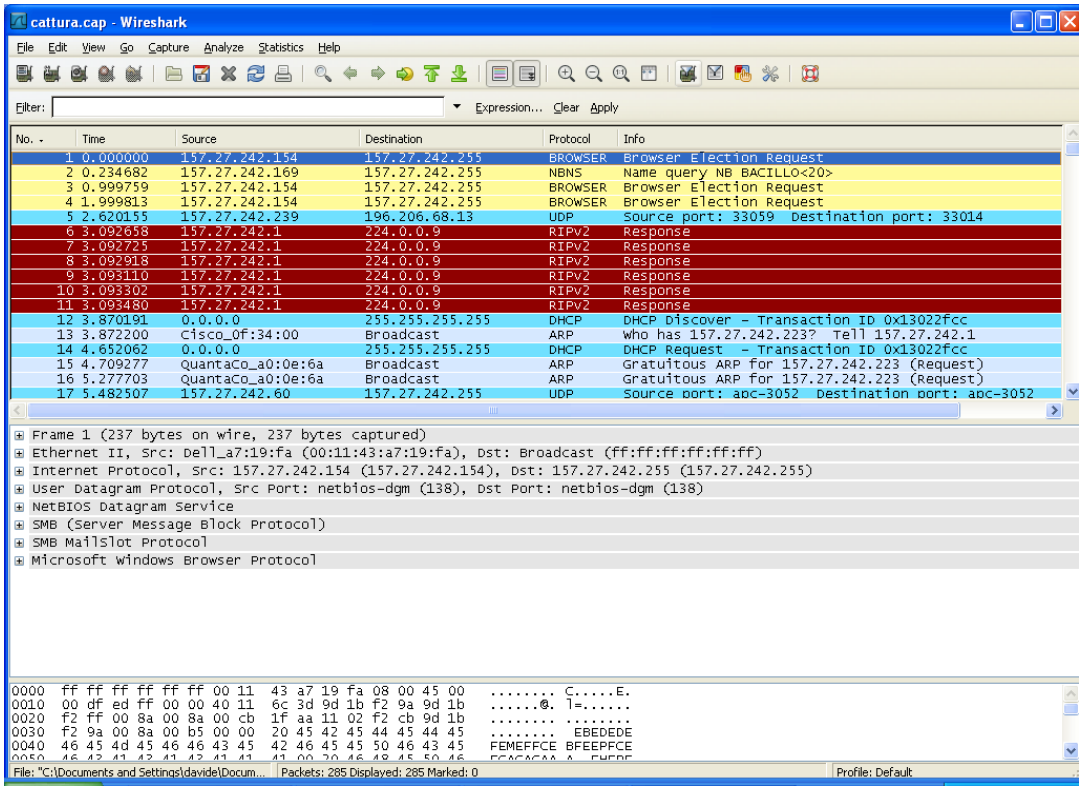


Se togliamo il segno di spunta a *Hide capture info dialog* e avviamo una cattura col bottone *Start*, apparirà la seguente finestra.



E' possibile aprire un file di pacchetti catturati precedentemente con l'opzione *Open* del menu *File*.

Sia nel caso di cattura dal vivo sia nel caso di apertura di file, Wireshark permette di visualizzare i pacchetti mediante tre pannelli come mostrato di seguito.



Il pannello più in alto contiene in forma tabulare la lista dei pacchetti catturati. Le colonne rappresentano alcuni campi degli headerdei pacchetti. E' possibile aggiungere/togliere colonne o cambiare il loro ordine mediante la voce *Preferences* del menu *Edit*. I pacchetti sono elencati in ordine

di cattura ma è possibile riordinarli secondo un diverso campo della tabella cliccando sull'intestazione della relativa colonna. E' possibile colorare tutti i pacchetti relativi ad una conversazione mediante la voce *Colorize Conversation* del menu *View*. E' possibile assegnare dei colori specifici a determinati pacchetti in base ai valori dei loro header; si utilizza l'opzione *Colouring Rules* del menu *View*.

Il pannello intermedio visualizza la struttura del pacchetto selezionato nel pannello in alto. Il pacchetto è mostrato in tutti i suoi strati: dal livello datalink al livello applicazione. Wireshark utilizza un meccanismo di interpretazione dei campi degli header che permette di trasformare le sequenze di bit in informazioni piu' facilmente leggibili dall'utente.

Il pannello in basso mostra il pacchetto come sequenza grezza di bit esattamente come è arrivato sull'interfaccia di rete.

Il menu *Analyze* consente di applicare dei filtri di visualizzazione per limitare il numero di pacchetti visualizzati. Le regole di creazione di tali filtri sono le stesse dei filtri di cattura.

Il menu *Statistics* consente di estrarre una serie di statistiche dall'insieme dei pacchetti catturati. La voce *Summary* permette di visualizzare un rapporto sommario sulla cattura mentre la la voce *IO Graph* visualizza un grafico con l'andamento di varie grandezze (ad es. Bitrate o packet rate) relative a tutti i pacchetti catturati o ad un sottoinsieme precedentemente definito mediante i colori. Per le altre voci si rimanda all'help o al manuale del tool.