Progettazione di Servizi Web e Reti di Calcolatori

Politecnico di Torino – Prof. Antonio Lioy

AA 2019-2020, esercitazione di laboratorio n.1

Prompt dei comandi

Per utilizzare gli applicativi **netstat, wget, ps** e **tasklist** (spiegati nel seguito), è necessario aprire una finestra di **prompt dei comandi** (ad esempio, dal menù di avvio di Windows, digitare “cmd” e premere “invio”).

NOTA**:** Tra diverse versioni di Windows, ed a maggior ragione tra diversi sistemi operativi, si possono riscontrare differenze nei significati delle varie opzioni o persino mancanze.

Pertanto, verranno forniti i comandi specifici e le opzioni per i seguenti sistemi operativi:

- Windows 10
- Linux
- macOS (dalla versione 10.12)

Guida all’installazione degli applicativi

Wireshark

Per l’installazione di wireshark occorre:

- scaricare l’installer dell’applicativo al link: https://www.wireshark.org/#download, selezionando il sistema operativo utilizzato (es. Windows Installer 64-bit)
- avviare l’installer e seguire la procedura guidata

N.B. per il sistema operativo Linux Ubuntu è possibile installare wireshark direttamente da terminale utilizzando il comando:

- **sudo apt install wireshark**

wget

Nel caso dei sistemi macOS e Linux il comando è già a disposizione ed utilizzabile attraverso il terminale.

Nel caso del sistema operativo Windows 10 è necessario effettuare le seguenti operazioni:

- scaricare l'applicazione "wget.exe" dal link: https://eternallybored.org/misc/wget/1.20.3/64/wget.exe
- spostare l'applicazione nella cartella C:\windows\System32
- aprire una nuova finestra col prompt dei comandi e testare il comando con **wget -h**

netstat, wget, ps, tasklist

Questi applicativi sono già installati sui sistemi operativi sopra indicati.

Identificativo di processo

Nei moderni sistemi operativi ogni processo viene identificato univocamente da un numero intero non negativo, chiamato “**process identifier**” (abbreviato in **PID**).

Il meccanismo che assegna questo numero allo specifico processo può variare a seconda del sistema operativo. Spesso i PID sono assegnati in sequenza, secondo l'ordine temporale di creazione dei processi.

Nell'ambiente operativo Windows è possibile visualizzare la lista dei PID per i processi in esecuzione in vari modi, ad esempio usando il comando **tasklist** all'interno di una finestra di **prompt dei comandi.**

Nel caso invece di ambiente macOS o Linux è possibile ottenere lo stesso risultato attraverso il comando **ps** (process status) con diverse opzioni. Viene di frequente utilizzato **ps aux**, dove:

- **a** mostra anche i processi degli altri utenti, e non del solo utente che ha lanciato il comando;
- **u** usa un formato con informazioni utili per l'analisi dell'utilizzo di risorse (memoria e CPU) dei processi;
- **x** mostra anche i processi che non hanno un terminale controllante.

Introduzione all'uso di "netstat"

L'applicazione **netstat** permette di visualizzare le principali informazioni sulle connessioni TCP (indirizzi IP, porte utilizzate, stati del protocollo) e sul protocollo UDP (indirizzi IP, porte utilizzate) attive (anche quelle in fase di terminazione) per il nodo su cui viene eseguito il comando.

Comando:

`netstat` [*opzioni …*]

Di seguito sono riportate le opzioni più utili ai fini dell'esercitazione, indicando per quale sistema operativo l'opzione funzioni (ok nella colonna corrispondente), rispetto ai sistemi operativi macOS, Linux **e** Windows**:**

Opzione	Descrizione	macOS	Linux	Win
-a	visualizza tutte le connessioni TCP attive e le porte TCP ed UDP in ascolto (per default netstat visualizza solo le connessioni TCP attive)		ok	ok
-b	visualizza il nome file del processo collegato alla connessione TCP o alla porta in ascolto			ok
-n	mostra gli indirizzi ip numerici (anziché simbolici) degli host e delle porte (ed è quindi più veloce perché non deve fare query inverse sul DNS)		ok	ok
-o	visualizza l'ID del processo (PID) collegato alla connessione TCP o alla porta in ascolto			ok
-p	mostra il nome del programma e il relativo PID che ha instaurato una connessione		ok	

-p <protocollo>	mostra tutte le connessioni attive relative al protocollo specificato, Il <*protocollo*> può assumere i seguenti valori: *tcp, udp, tcpv6, or udpv6*	ok		ok
-s	visualizza le statistiche per protocollo. Per default le statistiche sono visualizzate per i protocolli TCP, UDP, ICMP e IP. Se il protocollo IPv6 è installato è possibile visualizzare anche le statistiche delle versioni v6 di TCP e UDP trasportati da IPv6 ed ICMPv6		ok	ok
-v	aggiunge informazioni all'output (es. numero PID)	ok		
-t	mostra solo le connessioni TCP		ok	
-u	mostra solo le connessioni UDP		ok	
intervallo	visualizza periodicamente le informazioni richieste, con *intervallo* (in secondi) pari a quello specificato			ok

Introduzione all'uso di "wget"

L'applicativo **wget**[1] permette di scaricare file usando principalmente i protocolli HTTP(S) e FTP.

Comando per eseguire il download di una risorsa:

```
wget [ opzioni ] url
```

Dove URL è l'indirizzo della risorsa da scaricare, ad esempio:

```
wget http://www.miosito.com/foto1.jpg
```

Per le esercitazioni di laboratorio non sarà necessario specificare alcuna opzione.

Introduzione all'analizzatore di pacchetti "wireshark"

Per alcuni esercizi è necessario usare uno "sniffer" (analizzatore di pacchetti di rete). In laboratorio è installato **wireshark**[2], tra i più potenti e flessibili in ambito open-source. Per l'uso dell'applicativo fare riferimento alla relativa presentazione sul sito web del corso. Si ricorda che **wireshark** supporta due tipologie di filtri, quelli in fase di cattura[3] e quelli per la visualizzazione[4]. Al fine di comprenderne il funzionamento si consiglia di provarli entrambi.

[1] versione per Windows alla pagina https://eternallybored.org/misc/wget/
[2] http://www.wireshark.org/
[3] https://wiki.wireshark.org/CaptureFilters
[4] https://wiki.wireshark.org/DisplayFilters

Esercizio 1.1

Eseguire il comando **netstat** per visualizzare le porte TCP aperte in ascolto e le connessioni attive.

Rispondere alle seguenti domande:

1. Esistono servizi in ascolto su porte statiche?
2. In caso affermativo, guardare gli elementi nella colonna “Foreign Address”. Ci sono righe per cui sono specificati dei “Foreign Address” generici (ovvero “`*:*`”)? Se si, ipotizzare quale sia il motivo.
3. Verificare se esistono connessioni TCP attive. Se si, indicare:
    a. la quintupla identificativa della connessione
    b. lo stato del protocollo TCP in cui si trova

Esercizio 1.2

Usando il comando **netstat**, si vogliono osservare i diversi stati del protocollo TCP.

A tal fine:

1. Eseguire il comando **netstat** per visualizzare le connessioni attive.
2. Utilizzare il browser per raggiungere l’indirizzo “www.libero.it”.
3. Rilanciare il comando **netstat**.

Rispondere quindi alle seguenti domande:

1. Come varia il numero di connessioni TCP rispetto all’esercizio precedente? Spiegarne il motivo.
2. Qual è l’ID del processo a cui fanno riferimento le nuove connessioni? A quale processo è associato? A quale applicazione è associato il processo (per identificarla si suggerisce di utilizzare il Task Manager di Windows)?
3. Qual è lo stato del protocollo TCP per ogni connessione?

Chiudere il browser.

Esercizio 1.3

In modo simile all’esercizio precedente utilizzare il browser per raggiungere l’indirizzo *www.libero.it*, rilanciare periodicamente il comando **netstat** (ad esempio ogni 10 secondi) e rispondere alle seguenti domande:

1. Quali connessioni TCP sono attive e quale stato assume il protocollo? (osservare le quintuple e l’ID del processo per identificare le connessioni)
2. Esistono connessioni nello stato TIME_WAIT?
3. Esistono connessioni nello stato LAST_ACK?
4. Per quanto tempo la connessione TCP rimarrà in questo stato?
5. Considerando gli stati TCP del client, vengono elencati stati diversi da ESTABLISHED e TIME_WAIT? Perché?

Chiudere il browser. Dopo alcuni minuti (ad esempio 4) rilanciare il comando **netstat**.

Come varia il numero di connessioni TCP attive rispetto a prima? Perché?

Esercizio 1.4

Attivare **wireshark** sull'interfaccia di rete Ethernet (es. `eth0`, `Local Area Network`).

Senza aprire altre applicazioni, catturare i pacchetti per qualche istante (ad esempio 1 minuto), quindi interrompere la cattura. Memorizzare i pacchetti catturati sul file `'prova.pcapng'`.

Qual è la dimensione di tale file?

Aprire in wireshark il file `'prova.pcapng'` ed eseguire i seguenti punti:

1. stimare approssimativamente il numero di pacchetti per secondo catturati nella precedente acquisizione;
2. identificare se vi sono pacchetti TCP presenti;
3. utilizzando la funzionalità "Conversations" è possibile individuare quali host e quali livelli dello stack ISO/OSI sono coinvolti nelle diverse comunicazioni?

Esercizio 1.5

Si vogliono analizzare i protocolli coinvolti durante il trasferimento di una risorsa web.

A tal fine si segua la seguente procedura e si risponda alle domande.

Aprire l'applicazione wireshark, predisporla alla cattura dei pacchetti (specificando l'interfaccia di rete) ed avviare la cattura dei pacchetti.

Usando l'applicativo wget, scaricare il file al seguente indirizzo:

`http://security.polito.it/~lioy/01nbe/lab2/test1k.dat`

1. Quali protocolli della pila TCP/IP sono coinvolti nel trasferimento di questo file?
2. Riuscite ad identificare
    a. le fasi di handshake del protocollo TCP?
    b. le fasi di chiusura del protocollo TCP?

Per ciascun pacchetto appartenente a queste fasi, riportate i valori dei vari flag TCP ed annotate il tempo trascorso in millisecondi tra i pacchetti appartenenti a ciascuna fase.