

Web server

Corso di **Applicazioni Telematiche**

A.A. 2005-06 – Lezione n.4

Prof. Roberto Canonico

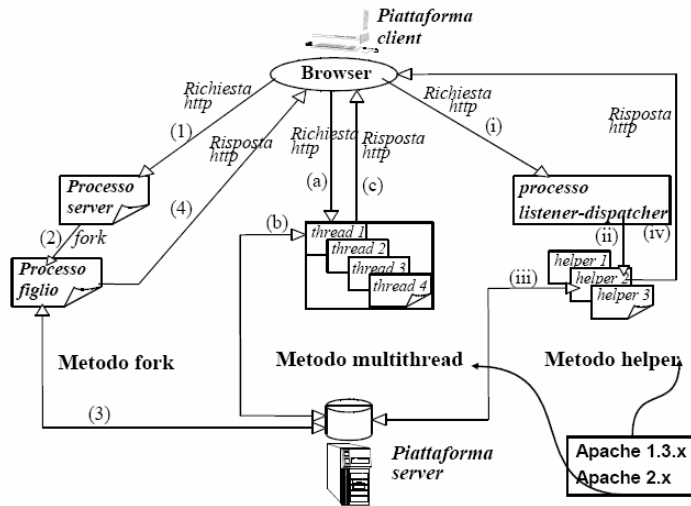
Università degli Studi di Napoli Federico II

Facoltà di Ingegneria

Web Server

- Per realizzare un sistema “web server” si ricorre tipicamente ad un software apposito che implementa:
 - la logica del protocollo HTTP lato server
 - la traduzione tra URL e file memorizzati in un file system locale
 - funzioni di logging
 - eventuali estensioni che consentono di eseguire codice in seguito alla richiesta di una URL
- Esistono varie implementazioni software di web server: Apache httpd, IIS di Microsoft, Zeus, ...
- Esistono anche implementazioni minimali di web server interamente hardware per sistemi embedded (interfacce web per la configurazione di apparecchi di vario tipo)

Web Server: gestione della concorrenza



Gestione della concorrenza (2)

- Server monoprocesso
- Servizio parallelo mediante fork
 - Si genera un nuovo processo ad ogni richiesta
- Servizio parallelo mediante processi helper
 - Usato da Apache 1.3
 - Si crea un pool di processi helper a cui vengono smistate le richieste
- Servizio parallelo con thread
 - Usato da Apache 2.0
 - I thread sono più "leggeri" dei processi

Pro e contro del multithreading

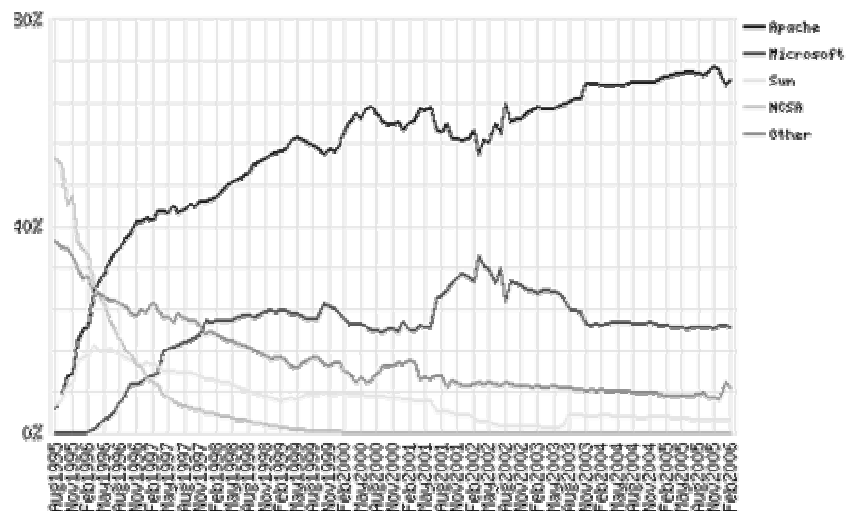
- Pro: Creazione di thread ha costo minore rispetto a processi
 - non c'è bisogno di copiare lo spazio di indirizzamento del processo padre nel figlio
- Pro: Context switch ha costo minore rispetto ai processi
 - Spazio di indirizzamento è condiviso tra i thread, non bisogna cambiare lo spazio di indirizzamento
- Contro: un malfunzionamento di un thread si può propagare ad altri thread perché condividono la stessa memoria

Apache HTTP Server

- Apache HTTP Server è il web server HTTP della Apache Software Foundation
- Sviluppato e aggiornato per sistemi operativi come UNIX e Windows
- Open-source
- Apache è dal 1996 il web server più diffuso in Internet
- <http://httpd.apache.org>
- La versione più recente (2.x) ha una struttura sw che combina l'esistenza di più processi con il multithreading



Market Share for Top Servers



news.netcraft.com - February 2006

Modularità di Apache

- Apache è un server modulare: le sue funzionalità sono rese disponibili tramite dei moduli che possono essere aggiunti o disattivati nel server sfruttando il modulo *mod_so*
- Le funzionalità di base sono incluse nel core
- Nel file di configurazione (`httpd.conf`) si possono scegliere i moduli da aggiungere al server
- I moduli possono essere :
 - Compilati staticamente nel web server
 - Aggiunti dinamicamente nel file `httpd.conf` tramite il supporto DSO (Dynamic Shared Object)

File di configurazione

- Apache viene configurato tramite delle direttive riportate nel file di testo `httpd.conf`
- La configurazione è aggiornata solo al riavvio del server
- La gestione dei tipi MIME è gestita nel file `mime.types`
- Le direttive si applicano all'intero server, a meno di specificare lo scope, se consentito, in cui sono valide (es. usando `<Directory>` per definire la directory in cui applicarla)
- Il file di configurazione principale è `httpd.conf`, ma se ne possono creare di nuovi, per ogni sito o ogni directory. Si usano i file `.htaccess`, le cui direttive vengono applicate solo al sotto albero in cui il file `.htaccess` è presente.

Uso delle direttive nei file di configurazione

- Le direttive nei file di configurazione possono essere applicate all'intero server, a directory, a file, host o URL
- Possono essere valutate solo all'avvio del server (es. `<IfModule>`) oppure ad ogni richiesta ricevuta
- Possono riguardare la struttura del filesystem (es. `<Directory>`) oppure la struttura del sito (Es. `location`)

```
<Directory /home/pippo/public_cgi/>  
    Options ExecCGI  
</Directory>
```

Autenticazione

- Richiede innanzitutto la creazione di un file delle password utente (`htpasswd -c C:\www\passwd`)

- Utilizzando il modulo `mod_auth_dbm` le utenze possono anche essere gestite tramite un database

- Esempio di configurazione degli accessi:

```
AuthType Basic
AuthName "Restricted Area"
AuthUserFile C:/www/passwd
require valid-user
```

- L'abilitazione all'accesso può essere anche assegnato in base all'host-name o all'host-address della macchina che richiede il documento

```
allow from 143.218.1.3
deny from 143.218.1.165
```

Autenticazione (2)

- **AuthType Basic**

- Seleziona il metodo di autenticazione utilizzato
- Basic è implementato dal modulo `mod_auth`
- Digest è implementato dal modulo `mod_auth_digest`

- **AuthName "Restricted Area"**

- identifica un nome per l'area a cui accedere per consentire il riutilizzo delle password al client

- **AuthUserFile C:/www/passwd**

- il file contenente le password

- **Require valid-user**

- specifica il nome dell'utente abilitato all'accesso. Si possono anche creare dei gruppi per consentire l'accesso a più utenti

Autenticazione (3)

```
C:\www\htdocs\private>c:\wamp\Apache\Apache2\bin\htpasswd
                        -c C:\www\passwd paperino
Automatically using MD5 format.
New password: *****
Re-type new password: *****
Adding password for user paperino
```

File .htpasswd

```
AuthType Basic
AuthName "Restricted Files"
# (Following line optional)
AuthBasicProvider file
AuthUserFile C:\www\passwd
Require user paperino
```

Siti virtuali con Apache: httpd.conf

```
<VirtualHost *:80>
  DocumentRoot C:/www/htdocs/pippo
  ServerName pippo
  ErrorLog logs/pippo-error_log
  CustomLog logs/pippo-access_log common
</VirtualHost>
<VirtualHost *:80>
  DocumentRoot C:/www/htdocs/pluto
  ServerName pluto
  ErrorLog logs/pluto-error_log
  CustomLog logs/pluto-access_log common
</VirtualHost>
```

I due siti a confronto



Utilizzo di script CGI con Apache

- Il CGI (Common Gateway Interface) fornisce al server un meccanismo per interagire con programmi esterni nella creazione dei contenuti attraverso l'uso di programmi (script) CGI
- Ci sono due metodi per configurare l'uso di CGI in Apache
 - Definire una directory che contiene gli script, i quali devono avere i diritti di esecuzione
 - `ScriptAlias /cgi-bin /usr/local/apache/cgi-bin`
 - Dichiarare il permesso di eseguire gli script all'interno di determinate directory
 - `<Directory /usr/local/apache/htdocs/somedir>`
`Options +ExecCGI`
`</Directory>`
- Inoltre è possibile specificare l'estensione dei file da eseguire via CGI: ***AddHandler cgi-script cgi pl***

I file di log di Apache

- ***/logs/access.log***
 - registra gli accessi avvenuti con successo
- ***/logs/error.log***
 - registra (e descrive brevemente) gli errori
- Nel file di configurazione è possibile settare i nomi dei file di log:
 - `ErrorLog logs/error.log`
 - `CustomLog logs/access.log`
- Appositi tool sono in grado di produrre statistiche e grafici dall'analisi dei log (es. AWStats)
 - Analisi di Data Mining sui log (Web Mining)

Server access log file

Formato CLF

IP Address	Userid	Time	Method/URL/ Protocol	Status	Size
------------	--------	------	----------------------	--------	------

IP Address → indirizzo Internet della macchina da cui proviene la richiesta, può corrispondere all'indirizzo di un server proxy.

UserId → presente quando la richiesta richiede l'autenticazione per accedere a determinati file. Se non disponibile compare il segno "-".

Time → tempo al quale il Web server ha ricevuto la richiesta

Method/URL/Protocol → modalità di accesso al Web server

Status → codice ritornato dal server in risposta all'azione richiesta

Size → numero di bytes trasmessi

Server access log file

Formato ECLF

IP Address	Userid	Time	Method/URL/ Protocol	Status	Size	Referrer	Agent
------------	--------	------	----------------------	--------	------	----------	-------

IP Address → indirizzo Internet della macchina da cui proviene la richiesta, può corrispondere all'indirizzo di un server proxy

UserId → presente quando la richiesta richiede l'autenticazione per accedere a determinati file.

Time → momento in cui il Web server ha ricevuto la richiesta

Method/URL/Protocol → modalità di accesso al Web server

Status → codice ritornato dal server in risposta all'azione richiesta

Size → numero di bytes trasmessi

Referrer → URL del documento da cui è stata emessa la richiesta della pagina

Agent → sistema operativo e browser utilizzati dal client

Finalità del Web Mining

- Capire il comportamento degli utenti
 - Analizzare il comportamento di navigazione degli utenti per capirne le preferenze e quindi massimizzare l'attività commerciale
 - Capire chi compra e che cosa compra
 - Scoprire la % di clienti acquisiti/persi
- Determinare l'efficacia di un sito Web
 - Capire i pattern di navigazione per
 - migliorare la struttura del sito
 - personalizzare la navigazione
- Misurare il successo dell'attività di marketing
 - Capire l'impatto sull'attività commerciale delle campagne di marketing
 - Classificare gli utenti per personalizzare l'offerta
 - Misurare l'efficacia di specifici banner pubblicitari.

Domande?

