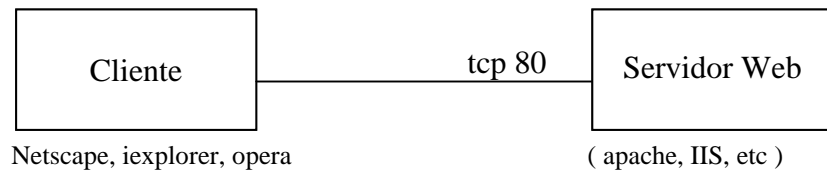


Hyper Text Transfer Protocol (HTTP)

Protocolo HTTP

- Funciona na filosofia cliente-servidor sobre o protocolo TCP na porta 80



- Tal como o protocolo SMTP, é um protocolo de texto (ASCII).

Sessão HTTP

- Cliente emite comando HTTP com endereço do documento e versão HTTP como argumentos
- Cliente envia informação adicional, propriedades do browser, etc sobre a forma de **cabeçalhos**
- Cliente envia **linha em branco para terminar o cabeçalho**
- Cliente envia dados (opcional)

Sessão HTTP (2)

Exemplo 1:

```
$telnet 66.199.249.146 80
GET /index.php HTTP/1.0
(linha em branco)
(linha em branco)
```

Exemplo 2:

```
$telnet 72.14.219.191 80
GET /cgi-bin/birthday.pl?month=august&date=24 HTTP/1.1
Host: deei.blogspot.com
User-Agent: Mozilla 2.1
Accept: image/gif, image/jpeg, */*
Cookie: sex=male
(linha em branco)
(linha em branco)
```

Sessão HTTP (3)

Exemplo 3:

```
$telnet 72.14.219.191 80
POST /cgi-bin/birthday.pl HTTP/1.1
Host: deei.blogspot.com
User-Agent: Mozilla 2.1
Accept: image/gif, image/jpeg, */*
Cookie: sex=male
Content-length: 52
(linha em branco)
```

```
month=august&date=24&first_name=John&last_name=Smith
```

Sessão HTTP (4)

- Servidor responde com linha de status
- Servidor envia um cabeçalho
- Servidor envia uma linha em branco
- Servidor envia dados (texto html, binário codificado em base 64)

Exemplo 3:

```
HTTP/1.1 200 OK
Date: Fri, 20 Sep 2003 08:17:57 GMT
Server: Apache/1.3
Content-type: text/html
Content-length: 2482
Set-Cookie: sex=male
(linha em branco)
dados (html, binário em base 64)
```

Sessão HTTP - mensagem multipart

```
HTTP/1.1 200 OK
Date: Fri, 20 Sep 2003 08:17:57 GMT
Server: Apache/1.3
Content-type: multipart/mixed; boundary="fronteira"
MIME-version: 1.0
(linha em branco)
--fronteira
Content-type: text/plain
(linha em branco)
aqui vai a foto das ferias!
--fronteira
Content-type: image/tiff
Content-transfer-encoding: base64
(linha em branco)
gajwO4+n2Fy4FV3V7zD9awd7uG8/TITP/vIocxXnnf/5mjgQjcupBUL1b3uyLwAV
tBLOP4nVLdIAhSzlZnyLAF8na0n7g6OSeej7aqI13NIXCfxDsPsY6NQjSvV77j4h
--fronteira
```

Configuração do servidor Web APACHE

A configuração do servidor web APACHE é realizada num único ficheiro:
`/etc/apache/httpd.conf`

Este ficheiro de texto (ASCII) está organizado em directivas globais e contentores com directivas locais

- **directivas globais** definem variáveis de ambiente, variáveis de funcionamento do servidor, módulos, formato dos logs, hosts virtuais, etc.
- **Contentores** restringem as directivas a um determinado espaço

```
< Directory directório >  
< /Directory >
```

```
< Files ficheiro >  
< /Files >
```


Directivas globais

ServerName www.sala160.deei.fct.ualg.pt

DocumentRoot /var/www/

ServerType standalone

Port 80

Alias /icons/ /var/www/icons

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin

UserDir public_html

AccessFileName .htaccess

DirectoryIndex index.html index.htm index.php default.html

Directivas globais para gestão de processos

MinSpareServers 5

MaxSpareServers 20

StartServers 8

MaxClients 150

MaxRequestPerChild 1000

User www-data

Group www-data

Directivas de performance

KeepAlive

KeepAlive Timeout 5

MaxKeepAlive Requests 1000

Timeout 300

Directivas de controle de acesso

```
< Directory /var/www >
```

```
Options opções
```

```
AllowOverride keyword
```

```
Order allow, deny
```

```
Allow from ...
```

```
Deny from ...
```

```
< /Directory >
```

Directivas de controle de acesso (2)

opções

All , ExecCGI , FollowSymLinks ,

Includes (permite ServerSideIncludes SSI)

IncludesNOEXEC (não permite #exec #include comandos)

Indexes

MultiViews (permite várias linguagens)

None

SymLinksIfOwnerMatch

Directivas de controle de acesso (3)

AllowOverride

(controla se o ficheiro .htaccess no directório em questão deve ser interpretado)

None, All, AuthConfig, FileInfo, Indexes, Limit

Order, Allow, Deny

Order - define a ordem pela qual as regras de controlo são avaliadas. As regras em 2º lugar são excepções às regras definidas em 1º lugar

Allow from domínios, nomes, IPs

Deny from domínios, nomes, IPs

Directivas de autenticação

```
<Directory ... >  
AuthType Basic  
AuthUserFile .htpassword  
require      valid-user  
</Directory>
```

Basic – o ficheiro definido em AuthUserFile pode ter a password em texto

AuthUserFile – define o ficheiro que contem a password. Utilizar o programa htpassword para adicionar a password ao ficheiro.

Directivas de configuração de módulos

- O apache consiste de um número grande (~75) de módulos dinâmicos (carregados quando o serviço arranca)
- Os módulos são identificados com a directiva LoadModule
- Os módulos de linguagens são interpretadores de código que devolvem HTML

Exemplo:

```
LoadModule    mod_php4    /usr/lib/apache/modules/mod_php4.so  
LoadModule    python_module /usr/lib/apache/1.3/mod_python.so
```


Directivas de configuração de Virtual Hosts (1)

- A versão 1.1 do protocolo HTTP permite virtual hosts: o mesmo web server serve vários domínios
- **NOTA: na configuração do servidor de DNS têm que estar os aliases correspondentes!**
- Exemplo:

```
server02  IN  A    10.11.160.2
```

```
www.boys  IN  CNAME  server02.sala160.deei.fct.ualg.pt.
```

```
www.girls IN  CNAME  server02.sala160.deei.fct.ualg.pt.
```

Directivas de configuração de Virtual Hosts (2)

- Os “hosts” virtuais são criados com a directiva `<VirtualHost>`:
- Exemplo:

```
<VirtualHost www.boys.sala160.deei.fct.ualg.pt>  
    ServerName www.boys.sala160.deei.fct.ualg.pt  
    DocumentRoot /var/www/boys  
</VirtualHost>
```

```
<VirtualHost www.girls.sala160.deei.fct.ualg.pt>  
    ServerName www.girls.sala160.deei.fct.ualg.pt  
    DocumentRoot /var/www/girls  
</VirtualHost>
```