

LAB08

Protocolo HTTP Servidor Apache

A. Protocolo HTTP

Neste exercício vamos utilizar a aplicação `telnet`, ligar-nos directamente à porta TCP 80 do servidor web, e familiarizarmo-nos com os comandos do protocolo HTTP.

1. Corre a aplicação `telnet` e pede a página de entrada do servidor web

```
#telnet 10.10.23.13 80
GET / HTTP/1.0
(linha em branco)
(linha em branco)
```

Que é que recebeste como resposta?

Linha de status: _____

Cabeçalhos: _____

Dados (html): _____

```
#telnet intranet.ualg.pt 80
GET /templates/manga/images/logo.png HTTP/1.0
(linha em branco)
(linha em branco)
```

Que é que recebeste como resposta?

Linha de status: _____

Cabeçalhos: _____

Dados: _____

2. Neste exercício vamos enviar dados para o servidor juntamente com o URL utilizando o comando `GET`

`test2.pl` é um programa em Perl que se encontra no servidor e que devolve uma página html com todas as variáveis fornecidas pelo servidor web ao programa (ver o código em apêndice). É através da variável _____ que o programa tem conhecimento dos dados que nós enviamos.

```
#telnet 10.10.23.13 80
GET /cgi-bin/test2.pl?nome=bruno&idade=23 HTTP/1.0
(linha em branco)
(linha em branco)
```

Que é que recebeste como resposta?

QUERY_STRING=_____

Numero:

Nome:

Data:

3. Neste exercício vamos enviar dados para o servidor web através do comando POST. `post.pl` é um programa em Perl que se encontra no servidor e que devolve uma página html com o corpo da mensagem HTTP (ver o código em apendice)

```
#telnet 10.10.23.13 80
POST /cgi-bin/post.pl HTTP/1.0
Content-Length: 22
(linha em branco)
ano=2003&mes=11&dia=20
(linha em branco)
```

Que é que recebeste como resposta?

B. Instalação do servidor Apache

4. Para instalar o servidor Apache basta executar o comando

```
#dpkg --purge apache
#apt-get install apache
```

5. Verifica que o serviço está a funcionar com o browser (opera...) abrindo o URL `http://localhost/`. (Não te esqueças de desligar o proxy!). Vê o que está definido no ficheiro de configuração `/etc/apache/httpd.conf`

Nota 1: é preciso definir a directiva `DocumentRoot` igual a `/var/www`

Nota 2: é preciso definir a directiva `ServerName` igual a `server__.sala160.ualg.pt`

Nota 3: é preciso que o nome canónico `server__.sala160.ualg.pt` esteja registado no servidor de DNS, que este esteja a funcionar, e que o "resolver" esteja a utilizar esse servidor¹

Funciona? _____

6. Altera a página de entrada do site para uma da tua preferência substituindo `/var/www/index.html` pela tua página html preferida. Por exemplo guarda a página de entrada do Google (`www.google.com`) utilizando o menu

(Opera) File > Save Page as ... > `/var/www/google.html`

Copia a página que acabaste de guardar para `/var/www/index.html`:

```
#cp -a /var/www/index.html /var/www/index.html.original
#mv /var/www/google.html /var/www/index.html
```

¹ Commandos que tens que executar para correr o servidor de DNS e configurar o "resolver"

```
#/etc/init.d/bind start
#xedit /etc/resolv.conf
search sala160.ualg.pt
nameserver 127.0.0.1
```

Numero:

Nome:

Data:

7. Faz refresh à página `http://localhost/`

(Opera) View > Reload

Quais são as diferenças relativamente à página original do google? _____

O que deves fazer para aparecer o logotipo do google? _____

Qual o nome do serviço e o IP do servidor que tens que “assaltar” (hijack) para re-direccionares os acessos ao google para o teu servidor? _____

C. Criação do directório web pessoal de um utilizador

8. Cria o utilizador `cantiflas` (se ainda não existir!)

```
#adduser cantiflas
#chmod a+x /home/cantiflas
```

9. Muda para o utilizador `cantiflas` e cria o directório `public_html`

```
#su - cantiflas
$mkdir public_html
$chmod a+rx public_html
```

10. Copia a tua página web preferida para dentro do directório `public_html`

```
$cp /var/www/index.html ~/public_html
```

11. Verifica com o browser que a página web pessoal do utilizador `cantiflas` está agora acessível em `http://localhost/~cantiflas`

Sucesso? _____

12. Qual a directiva em `/etc/apache/httpd.conf` que terás para mudar para que a pasta web pessoal `public_html` passe a chamar-se `web_folder` ?

D. Criação de um directório com acesso restrito

13. Cria a pasta "`privado`" do utilizador `cantiflas`

```
$mkdir ~/public_html/privado
$chmod a+x ~/public_html/privado
```

14. Cria o ficheiro `~/public_html/privado/.htaccess` com o seguinte conteúdo

```
AuthType Basic
AuthName "cantiflas"
AuthUserFile "/home/cantiflas/public_html/privado/.htpasswd"
Require valid-user
```

15. Cria o ficheiro .htpasswd

```
$cd ~/public_html/privado/
$/usr/bin/htpasswd -c -b .htpasswd cantiflas segredo
```

16. Verifica que te é pedida uma password para aceder ao URL

```
http://localhost/~cantiflas/privado
```

Qual a directiva que deves incluir no ficheiro .htaccess para garantir que o cantiflas é um utilizador válido do servidor? _____

Quais são as directivas que estão dentro do contentor <Files ~ "^\.ht" > que impedem que se possa fazer download do ficheiro .htpasswd ?

(Sugestão: a tua página web pessoal no DEEI <http://www.deei.fct.ualg.pt/~axxxx/>, pode ser protegida por password utilizando os comando que aprendeste neste guião ...)

E. Criação de dois hosts virtuais

17. Altera o servidor de DNS e introduz duas entradas adicionais em /etc/bind/db.hosts.

```
server__      IN      A          10.11.160.__
girls         IN      CNAME     server__.sala160.ualg.pt.
boys         IN      CNAME     server__.sala160.ualg.pt.
```

Substitui __ pelo número do teu servidor. Não te esqueças de fazer o restart ao serviço DNS (/etc/init.d/bind restart).

Nota: Se o servidor de DNS não existir, no mínimo é preciso que existam entradas para estes nomes em /etc/hosts !

18. Cria as pastas girls e boys em /var/www

```
#mkdir /var/www/girls
#mkdir /var/www/boys
#chmod a+rx /var/www/girls
#chmod a+rx /var/www/boys
```

19. Altera o ficheiro de configuração do Apache (/etc/apache/httpd.conf)

```
NameVirtualHost      10.11.160.__
<VirtualHost  boys.sala160.ualg.pt>
    ServerName        boys.sala160.ualg.pt
    DocumentRoot      /var/www/boys/
</VirtualHost>
```

Numero:

Nome:

Data:

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

20. Coloca dois ficheiros html **diferentes** em /var/www/girls e /var/www/boys. (por exemplo as paginas de entrada de <http://www.goodgirls.com/> e <http://www.niceguys.com/>)

21. Verifica com o browser (proxy desligado!) que os URLs

```
http://boys.sala160.ualg.pt/
http://girls.sala160.ualg.pt/
```

devolvem páginas diferentes, apesar de ambos os sites estarem alojadas no mesmo servidor web...

Sucesso?_____

22. Termina aqui este laboratório. Remove o serviço, desliga o servidor e o monitor.

```
#dpkg --purge apache
#halt -p
```

APÊNDICE

test2.pl

```
#!/usr/bin/perl

print "Content-type:text/html\n\n";
print <<EndOfHTML;
<html><head><title>Print Environment</title></head>
<body>
EndOfHTML

foreach $key (sort(keys %ENV)) {
    print "$key = $ENV{$key}<br>\n";
}

print "</body></html>";
```

post.pl

```
#!/usr/bin/perl

print "Content-type:text/html\n\n";

read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
@pairs = split(/&/, $buffer);
foreach $pair (@pairs) {
    ($name, $value) = split(/=/, $pair);
    $value =~ tr/+//;
    $value =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",
hex($1))/eg;
    $FORM{$name} = $value;
}

print "<html><head><title>Form Output</title></head><body>";
print "<h2>Results from FORM post</h2>\n";

foreach $key (keys(%FORM)) {
    print "$key = $FORM{$key}<br>";
}

print "</body></html>";
```