



1. Verifique a localização do ficheiro php.exe. De seguida coloque na PATH do sistema o endereço do ficheiro. Para tal aceda ao painel de controlo, escolha a opção sistema → Definições avançadas do sistema → Variáveis de ambiente. Acrescente ao path o endereço do ficheiro php.exe
2. Agora ative o cliente de telnet no seu Windows. Para tal aceda ao painel de controlo, escolha a opção → Programas e funcionalidades → Ativar ou desativar funcionalidades do Windows → Cliente Telnet. Selecione esta opção.
3. Para que serve o cliente de telnet?
4. Num ficheiro de texto escreva o seguinte programa php que implementa um servidor:

```
<?php
error_reporting(E_ALL);

/* Allow the script to hang around waiting for connections. */
set_time_limit(0);

/* Turn on implicit output flushing so we see what we're getting as it comes in. */
ob_implicit_flush();

$address = '193.137.128.171';
$port = 10000;

if (($sock = socket_create(AF_INET, SOCK_STREAM, SOL_TCP)) === false) {
    echo "socket_create() failed: reason: " . socket_strerror(socket_last_error()) . "\n";
}

if (socket_bind($sock, $address, $port) === false) {
    echo "socket_bind() failed: reason: " . socket_strerror(socket_last_error($sock)) . "\n";
}

if (socket_listen($sock, 5) === false) {
    echo "socket_listen() failed: reason: " . socket_strerror(socket_last_error($sock)) . "\n";
}

do {
    if (($msgsock = socket_accept($sock)) === false) {
        echo "socket_accept() failed: reason: " . socket_strerror(socket_last_error($sock)) . "\n";
        break;
    }

    /* Send instructions. */
    $msg = "\nWelcome to the PHP Test Server. \n" .
        "To quit, type 'quit'. To shut down the server type 'shutdown'. \n";

    socket_write($msgsock, $msg, strlen($msg));

    do {
        if (false === ($buf = socket_read($msgsock, 2048, PHP_NORMAL_READ))) {
            echo "socket_read() failed: reason: " . socket_strerror(socket_last_error($msgsock)) . "\n";
            break 2;
        }
    }
    if (!$buf = trim($buf)) {
        continue;
    }
}
```

```

    if ($buf == 'quit') {
        break;
    }
    if ($buf == 'shutdown') {
        socket_close($msgsock);
        break 2;
    }
    $talkback = "PHP: You said '$buf'.\n";
    socket_write($msgsock, $talkback, strlen($talkback));

    echo "$buf\n";

} while (true);

    socket_close($msgsock);
} while (true);

socket_close($sock);
?>

```

- 5.** Grave o ficheiro com o nome servidor.php.
- 6.** Substitua a variável address pelo endereço do seu computador. Veja como pode obter o endereço do seu computador.
- 7.** Que tipo de socket está a ser criado?
- 8.** O que faz o programa?
- 9.** Aceda a uma linha de comandos e execute a instrução php -q servidor.php
- 10.** Aceda a outra linha de comandos e faça telnet <endereço máquina local> 10000
- 11.** Envie várias mensagens e veja o que acontece.
- 12.** Faça as alterações necessárias para que o computador de um colega seu execute o programa servidor e você execute o cliente telnet enviando mensagens para o computador do seu colega.
- 13.** Altere o seu programa para que o servidor envie ao cliente a mensagem recebida, mas invertida, ou seja se o cliente enviar uma mensagem “Bom dia” o servidor responde “aid moB”
- 14.** Altere o seu programa para que funcione com sockets UDP ao invés de TCP.