

Exercício 2

Aluno: Caio Augusto Alves Nolasco

RA: 195181

Instituto de Computação Universidade Estadual de Campinas

Campinas, 24 de Novembro de 2020.

Sumário

1	Questão 1	2
2	Questão 2	2
3	Questão 3	3
4	Questão 4	6

1 Questão 1

De maneira geral, o backlog do socket do servidor informa o número máximo de conexões pendentes suportadas por um socket TCP. De forma mais específica, o tamanho definido pelo valor do parâmetro backlog da função listen() define o valor máximo da soma dos tamanhos de duas filas de conexões no socket: uma fila para conexões que ainda esperam o término do three way handshake e outra fila para conexões que já tiveram o handshake concluído.

2 Questão 2

Para kernels do Linux 2.2 em diante, o valor de backlog especifica o tamanho máximo de file para conexões estabelescidas que esperam ser aceitas. O valor padrão de backlog é salvo em uma variavel de sistema no endereço /proc/sys/net/core/somaxconn, com seu valor padrão igual a 128.

A seguir, uma foto da manpage da chamada listen()

```
File Edit View Search Terminal Help
EBADF The argument <u>sockfd</u> is not a valid file descriptor.
                ENOTSOCK
The file descriptor <u>sockfd</u> does not refer to a socket.
                 EOPNOTSUPP
The socket is not of a type that supports the listen() operation.
 CONFORMING TO
POSIX.1-2001, POSIX.1-2008, 4.4BSD (listen() first appeared in 4.2BSD).
                 To accept connections, the following steps are performed:

    A socket is created with socket(2).

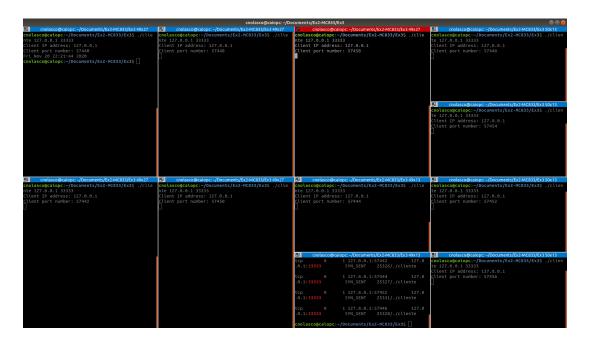
                           The socket is bound to a local address using bind(2), so that
other sockets may be connect(2)ed to it.

    A willingness to accept incoming connections and a queue limit
for incoming connections are specified with listen().

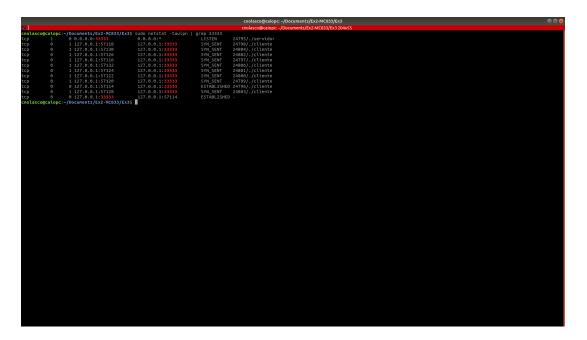
                           4. Connections are accepted with accept(2).
                 POSIX.1 does not require the inclusion of sys/types.h>, and this header file is not required on Linux. However, some historical (BSD) implementations required this header file, and portable applications are probably wise to include it.
                 The behavior of the <a href="backlog">backlog</a> argument on TCP sockets changed with Linux 2.2. Now it specifies the queue length for <a href="completely">completely</a> established sockets waiting to be accepted, instead of the number of incomplete connection requests. The <a href="mailto:maximum">maximum</a> length of the queue for incomplete sockets can be set using <a href="mailto:jproc/sys/net/ipv4/tcp-max-syn-backlog">jproc/sys/net/ipv4/tcp-max-syn-backlog</a>. When syncookies are enabled there is no logical maximum length and this setting is ignored. See <a href="mailto:tcp-max-syn-backlog">tcp-max-syn-backlog</a>. When syncookies are enabled there is no logical maximum length and this setting is ignored. See <a href="tcp-max-syn-backlog">tcp-max-syn-backlog</a> or when syncookies are enabled there is no logical maximum length and this setting is ignored. See <a href="tcp-max-syn-backlog">tcp-max-syn-backlog</a> or when syncookies are enabled there is no logical maximum length and this setting is ignored.
                 If the backlog argument is greater than the value in //proc/sys/net/core/somaxconn, then it is silently truncated to that value; the default value in this file is 128. In kernels before 2.4.25, this limit was a hard coded value, SOMAXCONN, with the value 128.
EXAMPLE
                  See bind(2).
SEE ALSO
accept(2), bind(2), connect(2), socket(2), socket(7)
 COLOPHON
Manual page listen(2) line 43/99 93% (press h for help or q to quit)
```

3 Questão 3

Os dez clientes são conectados simultaneamnte usando o programa Terminator para executar o cliente. Simultaneamente, o comando netstat pertinente para a porta do servidor também é chamado nesse momento.



Com o valor de backlog setado para 0, apenas um cliente conesegue se conectar imediatamente ao servidor.



Similarmente, com o backlog igual a 1, dois clientes se conectam imediatamente ao servidor.

```
| Comparison | Com
```

Com o backlog setado para 9, todos os cliente conseguem se conectar instantaneamente ao servidor.

4 Questão 4

O programa atualmente não faz uma chamada a função wait() e nem a função waitpid(), logo os processos filhos se tornam zumbis. Para lidar com os filhos zumbis, faz-se uma chamada para reconhecer o sinal SIGCHLD, e então uma chamada para wait().

```
void sig_chld() {{\bar{g}}}

pid t pid;
int stat;

pid = wait(&stat);
printf("child %d terminated\n", pid );
}
```