

TAREA 1: VALOR MÁXIMO DEL BACKLOG PARA UN SOCKET DE FLUJO

1. EN JAVA

La longitud máxima de la cola para las solicitudes de conexión entrantes se establece en el parámetro backlog.

Si el backlog es omitido, negativo o cero, se utiliza un valor predeterminado elegido por el sistema: e.g. 50. El backlog especificado debe ser ajustado por el sistema subyacente. Si el valor del backlog es excesivo para la plataforma, este es ajustado silenciosamente a un valor permitido. No existe ningún medio en Java o en la API de Berkeley Sockets para descubrir el valor eficaz del backlog.

Si llega una solicitud de conexión cuando la cola está llena, se rechaza la conexión.

2. EN C/C++

El backlog tiene un efecto en la velocidad máxima a la que un servidor puede aceptar nuevas conexiones TCP en un socket.

Muchos sistemas truncan silenciosamente el parámetro backlog de la llamada al sistema listen() a 5. La versión 1.2.13 del kernel de Linux lo hace en /usr/src/linux/net/inet/af_inet.c.

El parámetro de backlog se trunca silenciosamente a **SOMAXCONN** en /usr/src/linux/net/ipv4/af_inet.c. **SOMAXCONN se define como 128** en /usr/src/linux/socket.h para los núcleos 2.x.

Si el backlog se establece en SOMAXCONN, el proveedor de servicios subyacente responsable de los sockets establecerá el backlog a un valor razonable máximo. Si se establece en **SOMAXCONN_HINT** el backlog será ajustado para estar dentro del rango **(200, 65535)**. **SOMAXCONN_HINT** se puede utilizar para establecer el backlog a un valor más grande que el posible con **SOMAXCONN**.

SOMAXCONN_HINT solo es compatible con el proveedor de servicios TCP/IP de Microsoft. No existe una disposición estándar para obtener el valor de backlog real.

BIBLIOGRAFIA

1. <https://docs.oracle.com/javase/8/docs/api/java/net/ServerSocket.html>
2. E. Pitt, *Fundamental Networking in Java*. USA. Ed. Springer. 2006. pág. 28
3. <https://docs.microsoft.com/en-us/windows/desktop/api/winsock2/nf-winsock2-listen>
4. <https://www.linuxjournal.com/files/linuxjournal.com/linuxjournal/articles/023/2333/2333s2.html>