**调整内核printk打印级别–减少启动时的打印信息**

有时调试内核模块，打印信息太多了，可以通过修改/proc/sys/kernel/printk文件内容来控制。默认设置是

# cat /proc/sys/kernel/printk

7 4 1 7

该文件有四个数字值，它们根据日志记录消息的重要性，定义将其发送到何处。关于不同日志级别的更多信息，请查阅syslog(2)联机帮助。上面显示的4个数据分别对应：

控制台日志级别：优先级高于该值的消息将被打印至控制台

默认的消息日志级别：将用该优先级来打印没有优先级的消息

最低的控制台日志级别：控制台日志级别可被设置的最小值(最高优先级)

默认的控制台日志级别：控制台日志级别的缺省值

数值越小，优先级越高

其实这四个值是在kernel/printk.c 中被(默认值也在这里定义)定义的，如下：

int console\_printk[4] = {

DEFAULT\_CONSOLE\_LOGLEVEL, /\* console\_loglevel \*/

DEFAULT\_MESSAGE\_LOGLEVEL, /\* default\_message\_loglevel \*/

MINIMUM\_CONSOLE\_LOGLEVEL, /\* minimum\_console\_loglevel \*/

DEFAULT\_CONSOLE\_LOGLEVEL, /\* default\_console\_loglevel \*/

};

内核通过printk() 输出的信息具有日志级别，日志级别是通过在printk() 输出的字符串前加一个带尖括号的整数来控制的，如printk("<6>Hello, world!\n");。内核中共提供了八种不同的日志级别，在 linux/kernel.h 中有相应的宏对应。

#define KERN\_EMERG "<0>" /\* systemis unusable \*/

#define KERN\_ALERT "<1>" /\* actionmust be taken immediately \*/

#define KERN\_CRIT "<2>" /\*critical conditions \*/

#define KERN\_ERR "<3>" /\* errorconditions \*/

#define KERN\_WARNING "<4>" /\* warning conditions \*/

#define KERN\_NOTICE "<5>" /\* normalbut significant \*/

#define KERN\_INFO "<6>" /\*informational \*/

#define KERN\_DEBUG "<7>" /\*debug-level messages \*/

所以printk() 可以这样用：printk(KERN\_INFO"Hello, world!\n");。

未指定日志级别的printk() 采用的默认级别是DEFAULT\_MESSAGE\_LOGLEVEL，这个宏在kernel/printk.c 中被定义为整数4，即对应KERN\_WARNING。

如果要想在内核启动过程中打印少的信息，就可以根据自己的需要在kernel/printk.c中修改以上数值，重新编译即可！

/\* printk's without a loglevel use this.. \*/

#define DEFAULT\_MESSAGE\_LOGLEVEL 4 /\* KERN\_WARNING \*/

了解了上面的这些知识后，我们就应该知道如何手动控制printk打印了。例如，我想屏蔽掉所有的内核printk打印，那么我只需要把第一个数值调到最小值1或者0。

# echo 1 4 1 7 > /proc/sys/kernel/printk

或者

# echo 0 4 0 7 > /proc/sys/kernel/printk