

第十三章 守护进程

1. 守护进程特征

Wuhan University

- 守护进程（**daemon**）是生存期长的一种进程，常常在系统引导装入时启动，在系统关闭时才终止

```
ps -axj
```

2. 编程规则

Wuhan University

- 调用umask，将文件模式创建屏蔽字设置为一个已知值（通常为0）
- 调用fork，然后使父进程exit
 - 如果shell命令启动，父进程终止会让shell认为命令已经执行完毕
 - 子进程继承了父进程的进程组ID，获得了一个新的进程ID，保证了子进程不是一个进程组的组长进程，是进行setsid调用的先决条件
- 调用setsid创建一个新会话，使调用进程
 - 成为新会话的首进程；成为新进程组的组长进程；没有控制终端。

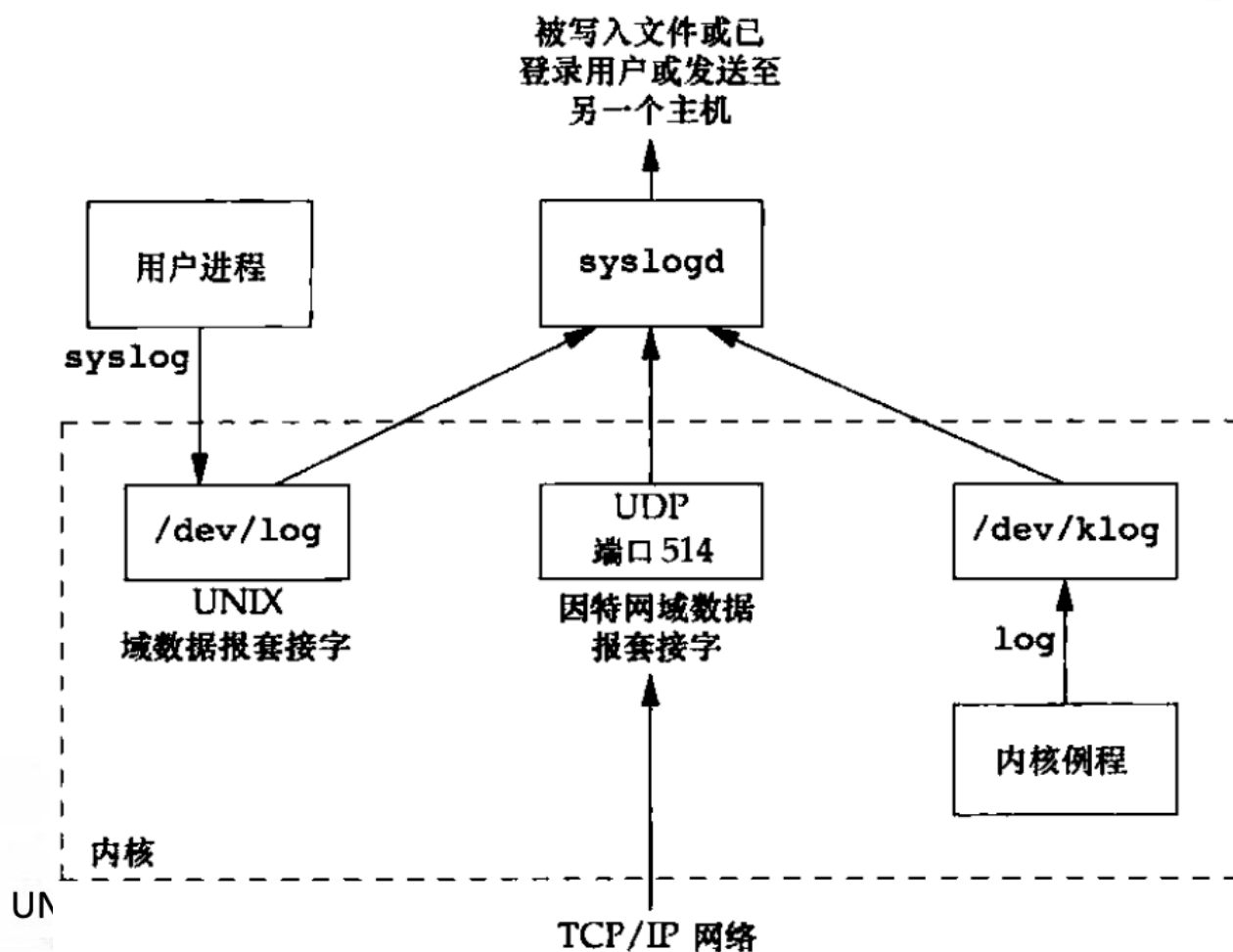
2. 编程规则

Wuhan University

- 将当前工作目录改为根目录或某个指定位置。
- 关闭不再需要的文件描述符。
- 某些守护进程打开/dev/null使其具有文件描述符0、1、2。保证守护进程不与终端设备相关联。

3. 出错记录

- BSD的syslog设施



3. 出错记录

- BSD的syslog设施

- 内核例程调用log函数，任何用户进程都可以通过打开并读取/dev/klog设备来读取这些消息
- 调用syslog(3)函数来产生日志消息，消息被发送至UNIX域数据报套接字/dev/log。
- 本机还是TCP/IP网络上的日志消息都通过UDP端口514进行发送

```
#include <syslog.h>
```

```
void openlog(const char *ident, int option, int facility);
```

```
void syslog(int priority, const char *format, ...);
```

```
void closelog(void);
```

```
int setlogmask(int maskpri);
```

3. 出错记录

option	XSI	说明
LOG_CONS	•	若日志消息不能通过 UNIX 域数据报送至 syslogd, 则将该消息写至控制台
LOG_NDELAY	•	立即打开至 syslogd 守护进程的 UNIX 域数据报套接字, 不要等到第一条消息已经被记录时再打开。通常, 在记录第一条消息之前, 不打开该套接字
LOG_NOWAIT	•	不要等待在将消息记入日志过程中可能已创建的子进程。因为在 syslog 调用 wait 时, 应用程序可能已获得了子进程的状态, 这种处理阻止了与捕捉 SIGCHLD 信号的应用程序之间产生的冲突
LOG_ODELAY	•	在第一条消息被记录之前延迟打开至 syslogd 守护进程的连接
LOG_PERROR	•	除将日志消息发送给 syslogd 以外, 还将它写至标准出错 (在 Solaris 上不可用)
LOG_PID	•	记录每条消息都要包含进程 ID。此选项可供对每个不同的请求都 fork 一个子进程的守护进程使用 (与从不调用 fork 的守护进程相比较, 如 syslogd)

<i>facility</i>	XSI	说明
LOG_AUDIT		审计设施
LOG_AUTH		授权程序: login、su、getty 等
LOG_AUTHPRIV		与 LOG_AUTH 相同, 但写日志文件时具有权限限制
LOG_CONSOLE		消息写入 /dev/console
LOG_CRON		cron 和 at
LOG_DAEMON		系统守护进程: inetd、routed 等
LOG_FTP		FTP 守护进程 (ftpd)
LOG_KERN		内核产生的消息
LOG_LOCAL0	•	保留由本地使用
LOG_LOCAL1	•	保留由本地使用
LOG_LOCAL2	•	保留由本地使用
LOG_LOCAL3	•	保留由本地使用
LOG_LOCAL4	•	保留由本地使用
LOG_LOCAL5	•	保留由本地使用
LOG_LOCAL6	•	保留由本地使用
LOG_LOCAL7	•	保留由本地使用
LOG_LPR		行式打印机系统: lpd、lpc 等
LOG_MAIL		邮件系统
LOG_NEWS		Usenet 网络新闻系统
LOG_NTP		网络时间协议系统
LOG_SECURITY		安全子系统
LOG_SYSLOG		syslogd 守护进程本身
LOG_USER	•	来自其他用户进程的消息 (默认)
LOG_UUCP		UUCP 系统

3. 出错记录

Wuhan University

<i>level</i>	说明
LOG_EMERG	紧急（系统不可使用）（最高优先级）
LOG_ALERT	必须立即修复的情况
LOG_CRIT	严重情况（如硬件设备出错）
LOG_ERR	出错情况
LOG_WARNING	警告情况
LOG_NOTICE	正常但重要的情况
LOG_INFO	信息性消息
LOG_DEBUG	调试消息（最低优先级）

3. 出错记录

Wuhan University

```
openlog("lpd", LOG_PID, LOG_LPR);  
syslog(LOG_ERR, "open error for %s: %m", filename);  
  
syslog(LOG_ERR | LOG_LPR, "open error for %s: %m", filename);
```

4. 单实例守护进程

Wuhan University

- 为了正常运作，某些守护进程会实现在任一时刻只运行该守护进程的一个副本。
- 使用文件或记录锁来保证只运行一个守护进程的副本


```
#define LOCKFILE "/var/run/daemon.pid"
#define LOCKMODE (S_IRUSR|S_IWUSR|S_IRGRP|S_IROTH)
```

```
extern int lockfile(int);
```

```
int
```

```
already_running(void)
```

```
{
```

```
    int    fd;
```

```
    char    buf[16];
```

```
    fd = open(LOCKFILE, O_RDWR|O_CREAT, LOCKMODE);
```

```
    if (fd < 0) {
```

```
        syslog(LOG_ERR, "can't open %s: %s", LOCKFILE, strerror(errno));
```

```
        exit(1);
```

```
    }
```

```
    if (lockfile(fd) < 0) {
```

```
        if (errno == EACCES || errno == EAGAIN) {
```

```
            close(fd);
```

```
            return(1);
```

```
        }
```

```
        syslog(LOG_ERR, "can't lock %s: %s", LOCKFILE, strerror(errno));
```

```
        exit(1);
```

```
    }
```

```
    ftruncate(fd, 0);
```

```
    sprintf(buf, "%ld", (long)getpid());
```

```
    write(fd, buf, strlen(buf)+1);
```

```
    return(0);
```

```
}
```

ersity

5. 守护进程惯例

Wuhan University

- 守护进程使用锁文件，该文件通常存储在`/var/run`目录中。锁文件的名称通常是`name.pid`。
- 若守护进程支持配置选项，则配置文件通常在`/etc`目录中。配置文件的名字通常是`name.conf`
- 守护进程可用命令行启动，但通常由系统初始化脚本（`/etc/rc*`或`/etc/init.d/*`）启动。如果守护进程终止时，应当自动重启它，可以在`/etc/inittab`中为该守护进程包括`respawn`记录项。
- 守护进程在启动时会读取配置文件，启动以后要使修改的配置文件生效，可以重启守护进程或向守护进程发送`SIGHUP`信号。

6. 客户进程-服务器进程模型

Wuhan University

- 守护进程通常用作服务器进程。
- 服务器进程调用`fork`然后`exec`另一个程序来响应客户进行的请求。服务器进程通常管理着多个文件描述符：通信端点、配置文件、日志文件等。为了安全，需要将不需要的文件描述符关闭。