DAEMON PROCESSES

DAEMON PROCESSES

- They have a long life
- They are started when the system is bootstrapped and terminate only when system is shutdown
- They don’t have a controlling terminal

Daemon characteristics

- When we execute ps-efjc command

  we can see that certain processes have no parent process ID ,process group ID and session ID
- Such processes are called daemon processes

- Certain daemon processes
- Syslogd – logs system messages for a an operator
- Sendmail – standard mailer daemon
- Cron – execute commands ay specified date and time
- Inetd – listens to system’s network interfaces for incoming requests from various network servers

Coding rules

- Call fork and have the parent exit
- Call setsid to create new session
- Change current working directory to new directory
- Set the file mode creation mask to 0
- Unneeded file descriptors should be closed

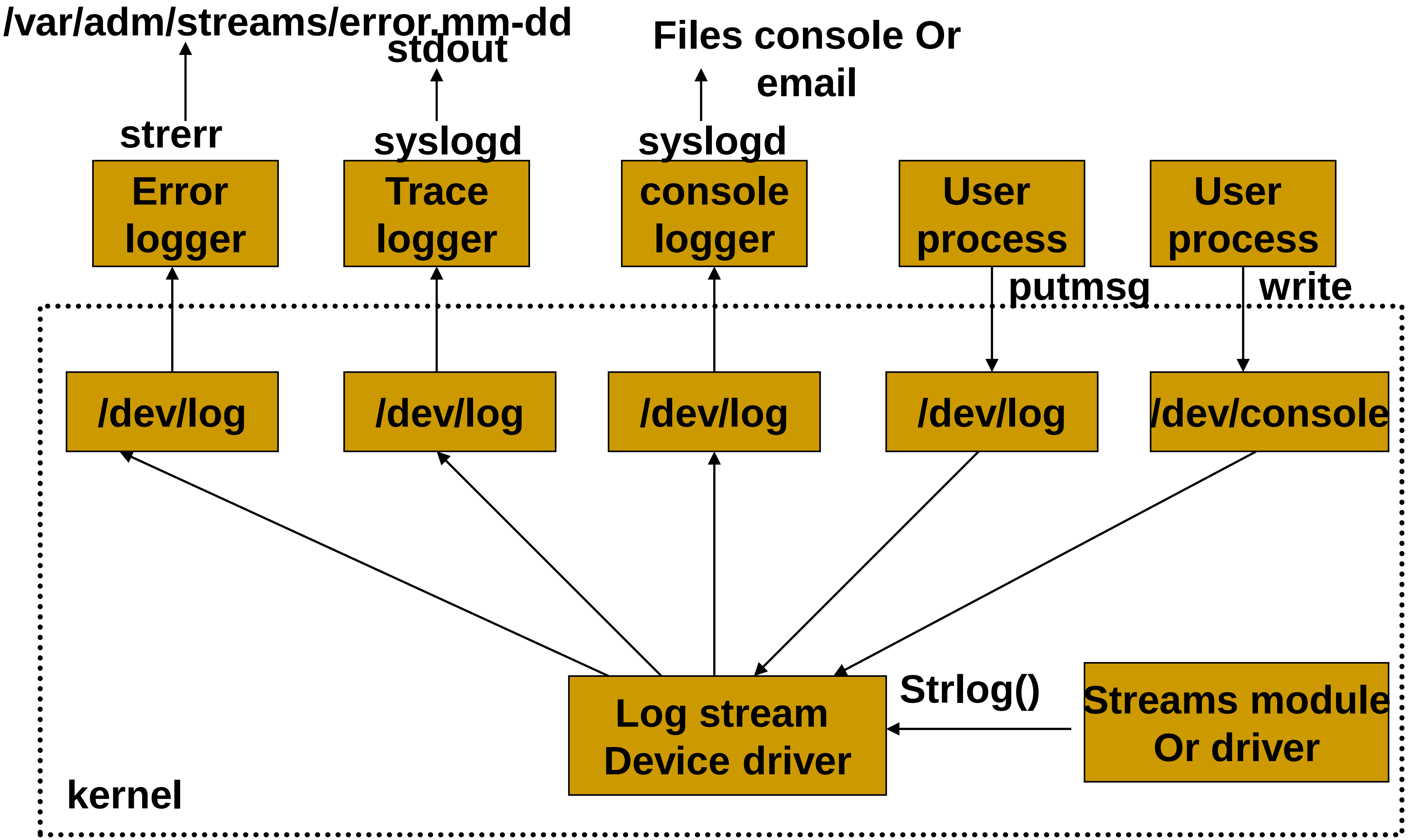
```
#include  <sys/types.h>
#include  <sys/stat.h>
#include  <fcntl.h>
#include  "ourhdr.h"

int  daemon_init(void)
{
    pid_t     pid;
if ( (pid = fork()) < 0)
        return(-1);
    else if (pid != 0)
        exit(0);     /* parent goes bye-bye */
                                 /* child continues */
   setsid();         /* become session leader */
   chdir("/");
                     /* change working directory */
   umask(0);
   return(0);
}
```

Error loging

- A daemon process cant write to the standard error , since it doesn’t have a controlling terminal

Svr5 Streams log Driver



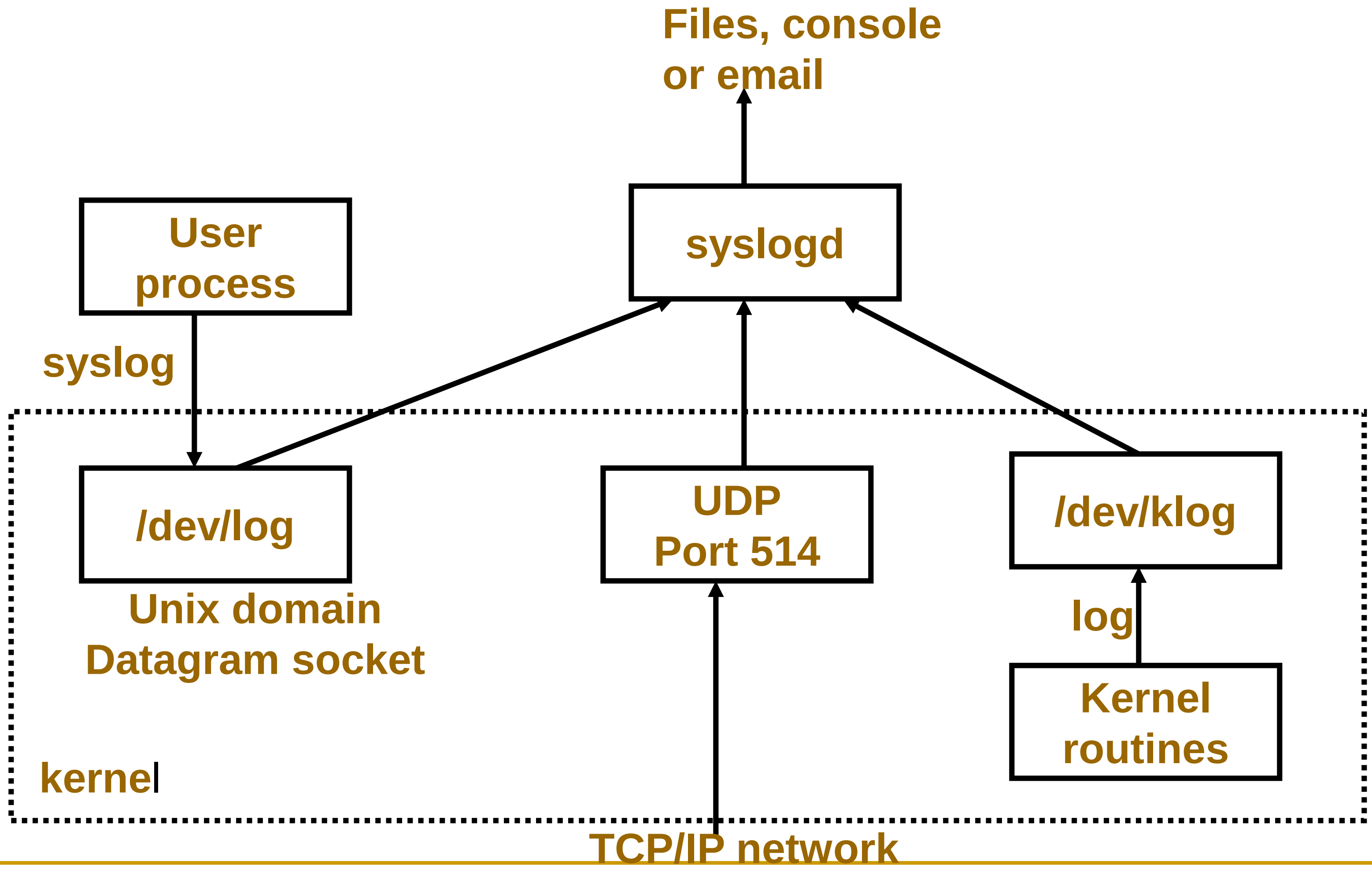
- Generating log messages

1. Routines within the kernel can call strlog to generate log messages.
2. A user process can putmsg to /dev/log. This message can be sent to any of the three loggers.
3. A user process can write to /dev/console. This message is sent only to the console logger.

- **Reading log messages**

1. **The normal error logger is strerr(1M). It appends these messages to a file in the directory /var/adm/streams.**
2. **The normal trace logger is strace(1M).**
3. **The standard console logger is syslogd, a BSD-derived program.**

4.3+BSD syslog Facility



- **Three ways of generating messages**

1. **Kernel routines can call the log function**
2. **Most user processes call the syslog function to generate log messages**
3. **A user process on this host, or on some other host that is connected to this by host by a TCP**

- Syslog

```
#include <syslog.h>
void openlog (char *ident, int option ,
                                             int facility);
void syslog (int priority, char *format,…);
void closelog (void);
```

- Calling openlog is optional
- Calling closelog is also optional

Option	Description
LOG_CONS	the message is written to the console
LOG_NDELAY	Open the UNIX domain datagram socket immediately
LOG_PERROR	Write the message to standard error in addition to sending it to syslogd
LOG_PID	Log the process ID with each message

Facility	Description
LOG_AUTH	Authorization programs
LOG_CRON	Cron and at
LOG_DAEMON	System daemons
LOG_KERN	Message generated by kernel
LOG_LOCAL0	Reserved for local use
LOG_LPR	Line printer system

Facility	Description
LOG_MAIL	**The mail system**
LOG_NEWS	**The usenet network news system**
LOG_SYSLOG	**The syslogd daemon itself**
LOG_USER	**Messages from other user processes**
LOG_UUCP	**The UUCP system**

Level	Description
LOG_EMERG	Emergency
LOG_ALERT	Condition that must be fixed immediately
LOG_CRIT	Critical condition
LOG_ERR	Error condition
LOG_WARNING	Warning condition
LOG_NOTICE	Normal but significant condition
LOG_INFO	Informational message
LOG_DEBUG	Debug message

Client-Server Model

- A common use of daemon process is as a server process
- Server is a process that waits for a client to contact it , requesting some type of service

Questions

- Discuss daemon characteristics and coding rules (10)
- What is a daemon? Give its basic coding rules. (10)