# 《深入理解Linux环境编程》目录

1. 前言
   1. 为什么要写这本书
   2. 本书特色
2. 基础知识
3. 一个Linux程序的诞生记

（讲述在Linux环境下，程序是如何生成的，简单介绍编译，链接等过程）

1. 程序的构成

（简单介绍Linux程序的二进制文件，包括各个段及其作用）

1. 程序是如何“跑”的
2. 加载过程
3. 运行过程
4. 退出过程
5. 什么是库函数
6. 什么是系统调用
7. 一些基础概念
8. 线程安全
9. 原子的
10. 可重入
11. 阻塞与非阻塞
12. 同步与异步
13. 工具介绍
14. 文件I/O
15. Linux中的文件
16. 文件、文件描述符、文件表
17. 内核中的实现

（内核中对文件、文件描述符及文件表的实现）

1. 打开文件
2. open介绍
3. 内核如何定位文件
4. 到底“打开”了什么
5. 内核如何选择文件描述符
6. 常见错误解析
7. 创建文件
8. creat的使用
9. creat与open
10. 常见错误解析
11. 关闭文件
12. close介绍
13. “关闭”了什么
14. 遗忘close会有什么问题
15. 如何查找文件资源泄漏
16. 关闭非法文件描述符是否导致错误
17. 常见错误解析
18. 文件寻找
19. lseek介绍
20. 小心lseek的返回值
21. 可以使用lseek确定文件大小吗
22. 读取文件
23. read介绍
24. 文件表首次登场
25. 部分读取
26. 常见错误解析
27. 写入文件
28. write介绍
29. 部分写入
30. write是原子的吗
31. 多个写者的问题
32. 文件表的二次登场
33. 文件的原子操作
34. 原子追加
35. 原子读写
36. 原子创建文件
37. 文件描述符的复制
38. dup与dup2的介绍
39. 什么被复制了
40. 为什么要复制
41. 文件数据同步
42. fsync、fdatasync、与sync介绍
43. 同步的必要性
44. 同步的机制
45. 同步何时完成
46. 文件与目录
47. 获取文件基本信息
48. stat、fstat、和lstat介绍
49. 文件基本信息解析
50. 内核如何保存文件基本信息
51. 文件权限
52. access介绍
53. 权限检查机制
54. umask介绍
55. umask机制
56. 为什么要用umask
57. chown、fchown与lchown介绍
58. chown、fchown与lchown机制
59. 权限位解析
60. 更多的标志位
61. SUID介绍与实现
62. SGID介绍与实现
63. Sticky 位介绍与实现
64. 文件的截断
65. truncate与ftruncate介绍
66. 截断的内核实现
67. 为什么需要截断

（以实际的例子，讲解截断的必要性）

1. 文件链接
2. link与unlink介绍
3. link是软链接还是硬链接
4. 巧用unlink
5. remove与unlink
6. symlink与readlink介绍
7. 常见错误解析
8. 标准I/O库
9. stdin、stdout与stderr是文件指针吗？
10. buffer引发的趣题
11. fopen与open标志位对比
12. fdopen后的文件描述符需要关闭吗？
13. 同时读写的痛苦
14. ferror返回的错误标志
15. clearerr清除错误标志后，文件指针可用吗？
16. ugetc压回多个字符的后果
17. 小心fgetc与putc
18. 注意fread与fwrite的返回值
19. 使用安全的I/O函数
20. 创建临时文件不同方法的对比与分析
21. 进程环境
22. main是入口吗？
23. “活雷锋”exit
24. atexit介绍
25. 使用atexit
26. atexit实现机制
27. atexit的局限
28. 小心处理环境变量
29. 使用动态库
30. 生成动态库
31. 使用动态库
32. 自动加载
33. 手工加载
34. 平顺升级动态库
35. 使用动态内存
36. 尴尬的realloc
37. 如何防止内存越界
38. 如何查找内存泄露
39. “长跳转”longjmp
40. setjmp与longjmp的使用
41. “长跳转”实现机制
42. “长跳转”的陷阱
43. 进程控制
44. 进程ID解析
45. fork解析
46. fork使用介绍
47. fork的实现机制
48. fork与锁
49. vfork与fork的区别
50. fork错误分析
51. 如何调试子进程
52. 系统调用exit
53. “双胞胎”exit？
54. exit实现机制
55. 为什么要“wait”子进程
56. exec解析
57. exec使用介绍
58. exec实现机制
59. 为什么要有exec
60. system解析
61. system使用介绍
62. system实现机制
63. 又爱又恨的system
64. 会“死机”的system
65. 常见错误解析
66. 理解信号机制
67. 什么是信号
68. signal解析
69. signal使用介绍
70. SIGDFL和SIGIGN
71. Linux的信号是可靠的吗
72. SIGCLD与SIGCHLD
73. Linux如何发送信号
74. Linux如何接收信号
75. Linux如何阻塞信号
76. sigaction的实现
77. siglongjmp与longjump
78. sigsuspend用途
79. 危险的system
80. sleep的精度
81. 巧用alarm
82. 用文件读取信号
83. 理解Linux线程
84. 线程与进程
85. pthread库线程模型
86. pthread库接口介绍
87. phtread\_create解析
88. pthread\_exit解析
89. pthread\_join解析
90. pthread\_cancel解析
91. pthread\_cleanup\_push与pthread\_cleanup\_pop
92. pthread\_detach解析
93. pthread的mutex
94. mutex介绍
95. mutex实现分析
96. 如何避免死锁
97. pthread的读写锁
98. 读写锁介绍
99. 读写锁实现分析
100. 读写锁性能分析
101. pthread的条件变量
102. 条件变量介绍
103. 条件变量实现分析
104. 条件变量的惊群效应
105. 线程控制
106. 线程属性
107. 线程属性接口介绍
108. 线程属性解析
109. mutex属性
110. mutex属性接口介绍
111. mutex属性解析
112. 读写锁属性
113. 读写锁属性接口介绍
114. 读写锁属性解析
115. 条件变量属性
116. 条件变量属性接口介绍
117. 条件变量属性解析
118. 线程专属数据
119. 线程专属数据接口介绍
120. 线程专属数据解析
121. 与TLS对比
122. 线程取消选项
123. 取消接口介绍
124. 取消接口实现分析
125. 取消点解析
126. 巧用pthread\_once
127. 线程与信号
128. 多线程，谁来接收信号？
129. 使用pthread\_sigmask代替sigprocmask
130. sigwait实现解析
131. pthread\_kill实现解析
132. 多线程不要fork
133. pread与pwrite
134. 高级I/O操作
135. 非阻塞I/O解析
136. 记录锁解析
137. 使用记录锁
138. 记录锁实现解析
139. 记录锁的缺陷
140. 劝告锁与强制锁
141. I/O多路复用
142. select使用
143. select实现解析
144. poll使用
145. poll实现解析
146. select和poll不能restart吗？
147. epoll使用
148. epoll实现解析
149. Linux的异步I/O
150. 什么是异步I/O
151. glibc的AIO
152. Linux内核的AIO
153. readv与writev解析
154. readv与writev使用介绍
155. readv与writev实现解析
156. 内存映射I/O
157. mmap使用介绍
158. mmap实现解析
159. 常见错误解析
160. 进程间通信
161. 管道
162. 管道的使用
163. 管道的实现解析
164. 常见错误解析
165. popen与pclose
166. popen的使用
167. popen实现解析
168. 命名管道
169. fifo的使用
170. fifo实现解析
171. XSI IPC的标示符和key
172. 消息队列
173. 消息队列使用
174. 消息队列实现解析
175. 常见错误解析
176. 信号量
177. 信号量使用
178. 信号量实现解析
179. 常见错误解析
180. 共享内存
181. 共享内存使用
182. 共享内存实现解析
183. 常见错误解析
184. Unix域socket
185. socketpair使用
186. socketpair实现解析
187. 命名unix域socket实现
188. 网络通信
189. socket文件描述符
190. 什么是socket
191. socket与文件描述符
192. 网络地址
193. 小心地址转换ntohl
194. 地址查找
195. 绑定IP地址
196. bind使用
197. bind实现解析
198. 客户端连接过程
199. connect使用
200. 非阻塞connect
201. connect实现解析
202. UDP可以connect吗？
203. 服务器端连接过程
204. listen使用
205. listen实现解析
206. accpet使用
207. accept实现解析
208. TCP三次握手实现解析
209. 发送数据
210. 从用户态到socket流程
211. 数据包结构
212. 数据包的发送流程
213. 接收数据
214. 网卡如何接收数据包
215. 数据包接收流程
216. 如何选择接收socket
217. 原始套接字
218. 什么是原始套接字
219. 原始套接字用途
220. 原始套接字与协议栈
221. 大师的错误?
222. 总结
223. 为什么要学习内核代码
224. 怎样学习内核代码