

# UNIX Network Programming Volume 2

*Second Edition*

## Interprocess Communications

*by W. Richard Stevens*



Prentice Hall PTR  
Upper Saddle River, NJ 07458  
[www.phptr.com](http://www.phptr.com)



# Abbreviated Table of Contents

<b>Part 1. Introduction</b>	<b>1</b>
Chapter 1. Introduction	3
Chapter 2. Posix IPC	19
Chapter 3. System V IPC	27
<b>Part 2. Message Passing</b>	<b>41</b>
Chapter 4. Pipes and FIFOs	43
Chapter 5. Posix Message Queues	75
Chapter 6. System V Message Queues	129
<b>Part 3. Synchronization</b>	<b>157</b>
Chapter 7. Mutexes and Condition Variables	159
Chapter 8. Read-Write Locks	177
Chapter 9. Record Locking	193
Chapter 10. Posix Semaphores	219
Chapter 11. System V Semaphores	281
<b>Part 4. Shared Memory</b>	<b>301</b>
Chapter 12. Shared Memory Introduction	303
Chapter 13. Posix Shared Memory	325
Chapter 14. System V Shared Memory	343
<b>Part 5. Remote Procedure Calls</b>	<b>353</b>
Chapter 15. Doors	355
Chapter 16. Sun RPC	399
Appendix A. Performance Measurements	457
Appendix B. A Threads Primer	501
Appendix C. Miscellaneous Source Code	505
Appendix D. Solutions to Selected Exercises	515

# Table of Contents

<b>Preface</b>		<b>xiii</b>
<b>Part 1. Introduction</b>		<b>1</b>
<b>Chapter 1. Introduction</b>		
1.1	Introduction	3
1.2	Processes, Threads, and the Sharing of Information	5
1.3	Persistence of IPC Objects	6
1.4	Name Spaces	7
1.5	Effect of <code>fork</code> , <code>exec</code> , and <code>exit</code> on IPC Objects	9
1.6	Error Handling: Wrapper Functions	11
1.7	Unix Standards	13
1.8	Road Map to IPC Examples in the Text	15
1.9	Summary	16
<b>Chapter 2. Posix IPC</b>		<b>19</b>
2.1	Introduction	19
2.2	IPC Names	19
2.3	Creating and Opening IPC Channels	22
2.4	IPC Permissions	25
2.5	Summary	26

<b>Chapter 3.</b>	<b>System V IPC</b>	<b>27</b>
3.1	Introduction	27
3.2	key-t Keys and ftok Function	28
3.3	ipcqerm Structure	30
3.4	Creating and Opening IPC Channels	30
3.5	IPC Permissions	32
3.6	Identifier Reuse	34
3.7	ipcs and ipcrm Programs	36
3.8	Kernel Limits	36
3.9	Summary	38
<b>Part 2. Message Passing</b>		<b>41</b>
<b>Chapter 4.</b>	<b>Pipes and FIFOs</b>	<b>43</b>
4.1	Introduction	43
4.2	A Simple Client-Server Example	43
4.3	Pipes	44
4.4	Full-Duplex Pipes	50
4.5	popen and pclose Functions	52
4.6	FIFOs	54
4.7	Additional Properties of Pipes and FIFOs	58
4.8	One Server, Multiple Clients	60
4.9	Iterative versus Concurrent Servers	66
4.10	Streams and Messages	67
4.11	Pipe and FIFO Limits	72
4.12	Summary	73
<b>Chapter 5.</b>	<b>Posix Message Queues</b>	<b>75</b>
5.1	Introduction	75
5.2	mq_open, mq_close, and mq_unlink Functions	76
5.3	mq_getattr and mq_setattr Functions	79
5.4	mq_send and mq_receive Functions	82
5.5	Message Queue Limits	86
5.6	mq_notify Function	87
5.7	Posix Realtime Signals	98
5.8	Implementation Using Memory-Mapped I/O	106
5.9	Summary	126
<b>Chapter 6.</b>	<b>System V Message Queues</b>	<b>129</b>
6.1	Introduction	129
6.2	msgget Function	130
6.3	msgsnd Function	131
6.4	msgrcv Function	132
6.5	msgctl Function	134
6.6	Simple Programs	135
6.7	Client-Server Example	140
6.8	Multiplexing Messages	142

6.9	Message Queues with <code>select</code> and <code>poll</code>	151
6.10	Message Queue Limits	152
6.11	Summary	155

## **Part 3. Synchronization** **157**

---

<b>Chapter 7.</b>	<b>Mutexes and Condition Variables</b>	<b>159</b>
7.1	Introduction	159
7.2	Mutexes: Locking and Unlocking	159
7.3	Producer–Consumer Problem	161
7.4	Locking versus Waiting	165
7.5	Condition Variables: Waiting and Signaling	167
7.6	Condition Variables: Timed Waits and Broadcasts	171
7.7	Mutexes and Condition Variable Attributes	172
7.8	Summary	174
<b>Chapter 8.</b>	<b>Read–Write Locks</b>	<b>177</b>
8.1	Introduction	177
8.2	Obtaining and Releasing Read–Write Locks	178
8.3	Read–Write Lock Attributes	179
8.4	Implementation Using Mutexes and Condition Variables	179
8.5	Thread Cancellation	187
8.6	Summary	192
<b>Chapter 9.</b>	<b>Record Locking</b>	<b>193</b>
9.1	Introduction	193
9.2	Record Locking versus File Locking	197
9.3	Posix <code>fcntl</code> Record Locking	199
9.4	Advisory Locking	203
9.5	Mandatory Locking	204
9.6	Priorities of Readers and Writers	207
9.7	Starting Only One Copy of a Daemon	213
9.8	Lock Files	214
9.9	NFS Locking	216
9.10	Summary	216
<b>Chapter 10.</b>	<b>Posix Semaphores</b>	<b>219</b>
10.1	Introduction	219
10.2	<code>sem_open</code> , <code>sem_close</code> , and <code>sem_unlink</code> Functions	225
10.3	<code>sem_wait</code> and <code>sem_trywait</code> Functions	226
10.4	<code>sem_post</code> and <code>sem_getvalue</code> Functions	227
10.5	Simple Programs	228
10.6	Producer–Consumer Problem	233
10.7	File Locking	238
10.8	<code>sem_init</code> and <code>sem_destroy</code> Functions	238
10.9	Multiple Producers, One Consumer	242
10.10	Multiple Producers, Multiple Consumers	245

---

10.11	Multiple Buffers	249	
10.12	Sharing Semaphores between Processes	256	
10.13	Semaphore Limits	257	
10.14	Implementation Using FIFOs	257	
10.15	Implementation Using Memory-Mapped I/O	262	
10.16	Implementation Using System V Semaphores	271	
10.17	Summary	278	
<b>Chapter 11.</b>	<b>System V Semaphores</b>		<b>281</b>
11.1	Introduction	281	
11.2	<code>semget</code> Function	282	
11.3	<code>semop</code> Function	285	
11.4	<code>semctl</code> Function	287	
11.5	Simple Programs	289	
11.6	File Locking	294	
11.7	Semaphore Limits	296	
11.8	Summary	300	
<b>Part 4.</b>	<b>Shared Memory</b>		<b>301</b>
<b>Chapter 12.</b>	<b>Shared Memory Introduction</b>		<b>303</b>
12.1	Introduction	303	
12.2	<code>mmap</code> , <code>munmap</code> , and <code>msync</code> Functions	307	
12.3	Increment Counter in a Memory-Mapped File	311	
12.4	4.4BSD Anonymous Memory Mapping	315	
12.5	SVR4 <code>/dev/zero</code> Memory Mapping	316	
12.6	Referencing Memory-Mapped Objects	317	
12.7	Summary	322	
<b>Chapter 13.</b>	<b>Posix Shared Memory</b>		<b>325</b>
13.1	Introduction	325	
13.2	<code>shm_open</code> and <code>shm_unlink</code> Functions	326	
13.3	<code>ftruncate</code> and <code>fstat</code> Functions	327	
13.4	Simple Programs	328	
13.5	Incrementing a Shared Counter	333	
13.6	Sending Messages to a Server	336	
13.7	Summary	342	
<b>Chapter 14.</b>	<b>System V Shared Memory</b>		<b>343</b>
14.1	Introduction	343	
14.2	<code>shmget</code> Function	343	
14.3	<code>shmat</code> Function	344	
14.4	<code>shmdt</code> Function	345	
14.5	<code>shmctl</code> Function	345	
14.6	Simple Programs	346	
14.7	Shared Memory Limits	349	
14.8	Summary	351	

<b>Part 5. Remote Procedure Calls</b>	<b>353</b>
<b>Chapter 15. Doors</b>	<b>355</b>
15.1 Introduction	355
15.2 <code>door-call</code> Function	361
15.3 <code>door-create</code> Function	363
15.4 <code>door-return</code> Function	364
15.5 <code>door-cred</code> Function	365
15.6 <code>door-info</code> Function	365
15.7 Examples	366
15.8 Descriptor Passing	379
15.9 <code>door-server-create</code> Function	384
15.10 <code>door_bind</code> , <code>door-unbind</code> , and <code>door-revoke</code> Functions	390
15.11 Premature Termination of Client or Server	390
15.12 Summary	397
<b>Chapter 16. Sun RPC</b>	<b>399</b>
16.1 Introduction	399
16.2 Multithreading	407
16.3 Server Binding	411
16.4 Authentication	414
16.5 Timeout and Retransmission	417
16.6 Call Semantics	422
16.7 Premature Termination of Client or Server	424
16.8 XDR: External Data Representation	426
16.9 RPC Packet Formats	444
16.10 Summary	449
<b>Epilogue</b>	<b>453</b>
<b>Appendix A. Performance Measurements</b>	<b>457</b>
A.1 Introduction	457
A.2 Results	458
A.3 Message Passing Bandwidth Programs	467
A.4 Message Passing Latency Programs	480
A.5 Thread Synchronization Programs	486
A.6 Process Synchronization Programs	497
<b>Appendix B. A Threads Primer</b>	<b>501</b>
B.1 Introduction	501
B.2 Basic Thread Functions: Creation and Termination	502
<b>Appendix C. Miscellaneous Source Code</b>	<b>505</b>
C.1 <code>unpipc.h</code> Header	505
C.2 <code>config.h</code> Header	509
C.3 Standard Error Functions	510

<b>Appendix D. Solutions to Selected Exercises</b>	<b>515</b>
<b>Bibliography</b>	<b>535</b>
<b>Index</b>	<b>539</b>