TCP/IP Architecture, Design and Implementation in Linux. Practitioners

Description: The only single-source reference on the concept and implementation of TCP/IP in Linux.

As open source software becomes a trusted part of business and research systems, it's no wonder that a combination of the Transmission Control Protocol/Internet Protocol (TCP/IP) and the Linux operating system is becoming more common. TCP/IP's prevalence allows easy communication among computers using various operating systems, whether Windows, Mac OS, Linux, or Unix. And Linux, because it is open source and thus modifiable, has become a frequent choice for developers who want a customizable operating system on which to build their applications.

This book describes the design and implementation of TCP/IP in Linux, from simple client-server applications to more complex executions. Topical coverage includes:

- Basic socket concepts and implementations
- The Linux implementation of network packets
- TCP read/write
- TCP algorithms for data transmission and congestion control
- TCP timers
- IP layer and routing tables implementation
- IP forwarding and quality of service implementation
- Netfilter hooks for the stacks
- Network Soft IRQ
- How to debug a TCP/IP stack

All topics are discussed in a concise, step-by-step manner and the book is complemented with helpful illustrations to give readers a better understanding of the subject. TCP/IP Architecture, Design, and Implementation in Linux is an indispensable resource for embedded-network product developers, network security product developers, IT network architects, researchers, and graduate students.

Contents:

- Preface.
- Acknowledgments.
- 1. INTRODUCTION.
  - 1.1 Overview of TCP/IP Stack.
  - 1.2 Source Code Organization for Linux 2.4.20.
  - 1.3 TCP/IP Stack and Kernel Control Paths.
  - 1.4 Linux Kernel Until Version 2.4 Is Non-preemptible.
  - 1.5 Linux Process and Thread.
  - 1.6 Kernel Synchronization Mechanism.
1.7 Application Interfaces for TCP/IP Programming.
1.8 Shutdown.
1.9 I/O.
1.10 TCP State.
1.11 Summary.

2. PROTOCOL FUNDAMENTALS.
2.1 TCP.
2.2 TCP Options (RFC 1323).
2.3 TCP Data Flow.
2.4 Delayed Acknowledgment.
2.5 Nagle’s Algorithm (RFC 896).
2.6 TCP Sliding Window Protocol.
2.7 Maximizing TCP Throughput.
2.8 TCP Timers.
2.9 TCP Congestion Control.
2.10 TCP Performance and Reliability.
2.11 IP (Internet Protocol).
2.12 Routing.
2.13 netstat.
2.14 traceroute.
2.15 ICMP.
2.16 ping.
2.17 ARP/RARP.
2.18 Summary.

3. KERNEL IMPLEMENTATION OF SOCKETS.
3.1 Socket Layer.
3.2 VFS and Socket.
3.3 Protocol Socket Registration.
3.4 struct inet—protosw.
3.5 Socket Organization in the Kernel.
3.6 Socket.
3.7 inet—create.

3.8 Flow Diagram for Socket Call.

3.9 Summary.

4. KERNEL IMPLEMENTATION OF TCP CONNECTION SETUP.

4.1 Connection Setup.

4.2 Bind.

4.3 Listen.

4.4 Connection Request Handling by Kernel.

4.5 Accept.

4.6 Client Side Setup.

4.7 Summary.

5. sk—buff AND PROTOCOL HEADERS.

5.1 struct sk—buff.

5.2 struct skb—shared—info.

5.3 sk—buff and DMA SKB—FRAG—STRUCT.

5.4 Routines Operating on sk—buff.

5.5 sk—buff Builds Protocol Headers as It Traverses Down the Protocol Layers.

5.6 sk—buff Extracts Protocol Headers as It Traverses Up the Protocol Layers When a Packet Arrives.

5.7 Summary.

6. MOVEMENT OF sk—buff ACROSS PROTOCOL LAYERS.

6.1 Packet Traversing Down the TCP/IP Stack.


6.3 Kernel Flow for a Packet Moving Down the Stack.

6.4 Packet Traversing Up the TCP/IP Stack.

6.5 Kernel Flow for a Packet Moving Up the Stack.

6.6 Summary.

7. TCP SEND.

7.1 TCP Segmentation Unit for Sending Data.

7.2 Segmentation with Scatter Gather Technique.

7.3 Sending OOB Data.

7.4 Flow for TCP Segmentation Unit and Send Process.

7.5 Functional Level Flow for Segmentation and Send Mechanism.
7.6 Summary.

8. TCP RECEIVE.

8.1 Queuing Mechanism.

8.2 Processing of TCP Data from the Receive Queue.

8.3 TCP Urgent Byte Processing.

8.4 DATA Flow Diagram for Receiving Data over the TCP Socket.

8.5 Summary.

9. TCP MEMORY MANAGEMENT.

9.1 Transmit Side TCP Memory Management.

9.2 Receive Side TCP Memory Management.

9.3 Freeing of Memory Allocated to a Receive Buffer.

9.4 System–Wide Control Parameters Are Worth Noticing When It Comes to TCP Memory Management.

9.5 Summary.

10. TCP TIMERS.

10.2 TCP Retransmit Timer.

10.3 Zero Window Probe Timer.

10.4 Delay ACK Timer.

10.5 Keepalive Timer.

10.6 SYN–ACK Timer.

10.7 TIME—WAIT Timer.

10.7.8 ——tcp—tw—hashdance().

10.8 Summary.

11. TCP CORE PROCESSING.

11.1 TCP Incoming Segment Processing.

11.2 Fast Path Processing.

11.3 Slow Path Processing.

11.4 Processing of Incoming ACK.

11.5 Processing of SACK blocks.

11.6 Reordering Length.

11.7 Processing TCP Urgent Pointer.

11.8 Processing Data Segments in Slow Path.
11.9 Overview of Core TCP Processing.
11.10 Summary.

12. TCP STATE PROCESSING.
12.1 Overview of State Processing.
12.2 TCP States.
12.3 Processing of Duplicate/Partial ACKs in Recovery State.
12.4 Processing of Duplicate/Partial ACKs in Loss State.
12.5 Default Processing of TCP States.
12.6 Processing of TCP Non-open States when ACKed Beyond tp high—seq.
12.7 Summary.

13. NETLINK SOCKETS.
13.1 Introduction to Netlink Sockets.
13.2 Netlink Socket Registration and Initialization at Boot Time.
13.3 How Is the Kernel Netlink Socket Created?
13.4 How Is the User Netlink Socket Created?
13.5 Netlink Data Structures.
13.6 Other Important Data Structures.
13.7 Netlink Packet Format.
13.8 Netlink Socket Example tc Command for Adding a qdisc.
13.9 Flow Diagram for tc Command in Kernel Space.
13.10 Summary.

14. IP ROUTING.
14.1 Routing.
14.2 Policy-Based Routing.
14.3 Multipathing.
14.4 Record Route Options (RFC 791) and Processing by Linux Stack.
14.5 Source Routing.
14.6 Linux Kernel Implementation of Routing Table and Caches.
14.7 Routing Cache Implementation Overview.
14.8 Managing Routing Cache.
14.9 Implementation Overview of Forwarding Information Base (FIB).
14.10 Adding New Entry in Routing Table Using ip Command (RT Netlink Interface).
14.11 What Happens When the ip Command Is Run with a Rule Option for Adding an Entry in the Routing Table?


14.13 Summary.

15. IP QUALITY OF SERVICE IN LINUX (IP QoS).

15.1 Introduction.

15.2 Basic Components of Linux Traffic Control.

15.3 Linux Implementation of pfi fo—fast qdisc.

15.4 Queueing Discipline Data Structure.

15.5 tc User Program and Kernel Implementation Details.

15.6 The tc Commands for Creating Class Hierarchy for CBQ.

15.7 Filters.

15.8 u32 Filter Implementation.

15.9 Route Filter Implementation.

15.10 Enqueue.

15.11 Overview of Linux Implementation of CBQ.

15.12 cbq—dequeue().

15.13 Summary.

16. IP FILTER AND FIREWALL.

16.1 Netfilter Hook Framework.

16.2 Netfilter Hooks on IP Stack.

16.3 Overview of Netfilter Hooks on Linux TCP–IP Stack.

16.4 Registration of Netfilter Hooks.

16.5 Processing of Netfilter Hooks.

16.6 Compatibility Framework.

16.7 Ip Chains.

16.8 How Is the Packet Filtered with Ipchains.

16.9 Iptables.

16.10 Iptables Filter Rules and Target Organization.

16.11 Organization of Filter Rules and Target for Iptables.

16.12 Filtering Packets with Iptables.

16.13 Summary.
19.14 Icrash Output for Setting Up Route Filter Using tc Command.

19.15 Netlink Data Structure.

19.16 Summary.

20. NEXT EDITION.

Bibliography.

Index.

Ordering: Order Online - http://www.researchandmarkets.com/reports/2174810/

Order by Fax - using the form below

Order by Post - print the order form below and send to

Research and Markets,
Guinness Centre,
Taylors Lane,
Dublin 8,
Ireland.
Fax Order Form
To place an order via fax simply print this form, fill in the information below and fax the completed form to 646-607-1907 (from USA) or +353-1-481-1716 (from Rest of World). If you have any questions please visit http://www.researchandmarkets.com/contact/

Order Information
Please verify that the product information is correct.

Product Name: TCP/IP Architecture, Design and Implementation in Linux. Practitioners
Web Address: http://www.researchandmarkets.com/reports/2174810/
Office Code: SCD2LHDD

Product Format
Please select the product format and quantity you require:

<table>
<thead>
<tr>
<th>Product Format</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Copy (Hard Back):</td>
<td>USD 116 + USD 29 Shipping/Handling</td>
</tr>
</tbody>
</table>

* Shipping/Handling is only charged once per order.

Contact Information
Please enter all the information below in BLOCK CAPITALS

Title:  
Mr  [ ]  Mrs  [ ]  Dr  [ ]  Miss  [ ]  Ms  [ ]  Prof  [ ]  
First Name:  
Email Address: *  
Job Title:  
Organisation:  
Address:  
City:  
Postal / Zip Code:  
Country:  
Phone Number:  
Fax Number:  

* Please refrain from using free email accounts when ordering (e.g. Yahoo, Hotmail, AOL)
Payment Information

Please indicate the payment method you would like to use by selecting the appropriate box.

☐ Pay by credit card: You will receive an email with a link to a secure webpage to enter your credit card details.

☐ Pay by check: Please post the check, accompanied by this form, to:
Research and Markets,
Guinness Center,
Taylors Lane,
Dublin 8,
Ireland.

☐ Pay by wire transfer: Please transfer funds to:
Account number 833 130 83
Sort code 98-53-30
Swift code ULSBIE2D
IBAN number IE78ULSB98533083313083
Bank Address Ulster Bank,
27-35 Main Street,
Blackrock,
Co. Dublin,
Ireland.

If you have a Marketing Code please enter it below:

Marketing Code: __________________________

Please note that by ordering from Research and Markets you are agreeing to our Terms and Conditions at http://www.researchandmarkets.com/info/terms.asp

Please fax this form to:
(646) 607-1907 or (646) 964-6609 - From USA
+353-1-481-1716 or +353-1-653-1571 - From Rest of World