

1	Grundbegriffe der Datenkommunikation	1
1.1	Überblick	1
1.2	ISO/OSI-Referenzmodell	2
1.3	TCP/IP-Referenzmodell	6
1.4	Beispiele bekannter verteilter Anwendungen	10
1.4.1	World Wide Web	10
1.4.2	Electronic Mail	13
1.4.3	WhatsApp	16
1.4.4	Skype	19
1.5	Nachrichtenaufbau und Steuerinformation	20
	Literatur	23
2	Grundkonzepte der Transportschicht	25
2.1	Grundlegende Aspekte	25
2.1.1	Typische Protokollmechanismen der Transportschicht	25
2.1.2	Verbindungsorientierte und verbindungslose Transportdienste	27
2.1.3	Zwischenspeicherung von Nachrichten	28
2.2	Verbindungsmanagement und Adressierung	30
2.2.1	Verbindungsaufbau	30
2.2.2	Verbindungsabbau	33
2.3	Zuverlässiger Datentransfer	36
2.3.1	Quittierungsverfahren	36
2.3.2	Übertragungswiederholung	38
2.4	Flusskontrolle	40
2.5	Staukontrolle	43
2.6	Segmentierung	44
2.7	Multiplexierung und Demultiplexierung	44
	Literatur	45

3	TCP-Konzepte und -Protokollmechanismen	47
3.1	Übersicht über grundlegende Konzepte und Funktionen	48
3.1.1	Grundlegende Aufgaben von TCP	48
3.1.2	Nachrichtenlänge	49
3.1.3	Adressierung	52
3.1.4	TCP-Steuerinformation	54
3.2	Ende-zu-Ende-Verbindungsmanagement	57
3.2.1	TCP-Verbindungsaufbau	57
3.2.2	TCP-Verbindungsabbau	59
3.2.3	Zustände beim Verbindungsauf- und -abbau	61
3.2.4	Zustandsautomat des aktiven TCP-Partners	63
3.2.5	Zustandsautomat des passiven TCP-Partners	64
3.2.6	Zusammenspiel der Automaten im Detail	66
3.3	Datenübertragungsphase	69
3.3.1	Normaler Ablauf	69
3.3.2	Timerüberwachung	72
3.3.3	Implizites Not-Acknowledge (NAK)	73
3.3.4	Flusskontrolle	74
3.3.5	Nagle- und Clark-Algorithmus	75
3.4	TCP-Protokolloptionen	79
3.4.1	Maximum Segment Size Option	79
3.4.2	TCP Window Scale Option	80
3.4.3	SACK-Permitted Option und SACK-Option	82
3.4.4	Timestamps Option	82
3.5	Protect Against Wrapped Sequences	83
3.5.1	Problemstellung	83
3.5.2	Maßnahmen zum Schutz vor Sequenznummernkollisionen	85
3.6	Staukontrolle	85
3.6.1	Slow-Start und Congestion Avoidance	86
3.6.2	TCP Reno und Fast-Recovery-Algorithmus	89
3.6.3	TCP NewReno	90
3.6.4	TCP BIC	92
3.6.5	TCP CUBIC	93
3.6.6	Explicit Congestion Notification	95
3.6.7	Weitere Ansätze der TCP-Staukontrolle	96
3.7	Timer-Management	98
3.7.1	Grundlegende Überlegung	98
3.7.2	Retransmission Timer	99
3.7.3	Keepalive Timer	101
3.7.4	Time-Wait Timer	103

3.7.5	Close-Wait Timer	103
3.7.6	Persistence Timer	104
3.8	TCP-Sicherheit	104
	Literatur	106
4	UDP-Konzepte und -Protokollmechanismen	107
4.1	Übersicht über grundlegende Konzepte und Funktionen	107
4.1.1	Grundlegende Aufgaben von UDP	107
4.1.2	Adressierung	108
4.1.3	UDP-Steuerinformation	109
4.1.4	Multiplexing und Demultiplexing	110
4.2	Datenübertragungsphase	111
4.2.1	Datagrammorientierte Kommunikation	111
4.2.2	Prüfsummenberechnung	112
4.3	UDP-Sicherheit	115
5	TLS – Transport Layer Security Protocol	117
5.1	Überblick über TLS	117
5.2	TLS-Protokollaufbau	119
5.3	TLS-Handshake	120
5.4	Verschlüsselte Nachrichtenübertragung	130
5.5	Abbau der TLS-Verbindung	133
5.6	Zusammenfassung und Ausblick	133
	Literatur	134
6	QUIC – UDP-Based Multiplexed and Secure Transport	135
6.1	Gründe für ein weiteres Transportprotokoll	136
6.2	QUIC-Nachrichtenaufbau	138
6.2.1	QUIC-Nachrichten in UDP-Datagrammen	138
6.2.2	QUIC-Paket-Header	139
6.3	QUIC-Verbindungen, Streams und Multiplexing	144
6.4	Verbindungsmanagement	146
6.4.1	Verbindungsaufbau	146
6.4.2	Verbindungsabbau	148
6.4.3	Verbindungsmigration	150
6.5	Datenübertragungsphase	150
6.5.1	Erzeugen und Schließen von Streams	151
6.5.2	Nachrichtenübertragung, Bestätigung und Sendungswiederholung	152
6.5.3	Flusskontrolle	158
6.5.4	Staukontrolle	159
6.6	Zusammenfassung und Ausblick	160
	Literatur	160

7	Programmierung von TCP- und UDP-Anwendungen	161
7.1	Überblick	162
7.2	Grundkonzepte der Socket-Programmierung	163
7.2.1	Einführung und Programmiermodell	163
7.2.2	TCP-Sockets	164
7.2.3	Datagramm-Sockets	166
7.3	Socket-Programmierung in C	168
7.3.1	Die wichtigsten Socket-Funktionen im Detail	168
7.3.2	Nutzung von Socket-Optionen	173
7.3.3	Nutzung von TCP-Sockets in C	175
7.3.4	Nutzung von UDP-Sockets in C	178
7.4	Socket-Programmierung in Java	181
7.4.1	Überblick über Java-Klassen	181
7.4.2	Streams für TCP-Verbindungen	185
7.4.3	Verbindungsorientierte Kommunikation über TCP	187
7.4.4	Gruppenkommunikation über UDP	188
7.5	Einfache Java-Beispiele	189
7.5.1	Ein Java-Beispielprogramm für TCP-Sockets	189
7.5.2	Ein Java-Beispielprogramm für Datagramm-Sockets	191
7.5.3	Ein Java-Beispiel für Multicast-Sockets	195
7.5.4	Zusammenspiel von Socket API und Protokollimplementierung	199
7.6	Ein Mini-Framework für Java-Sockets	201
7.6.1	Überblick über vordefinierte Schnittstellen und Objektklassen	202
7.6.2	Basisklassen zur TCP-Kommunikation	204
7.6.3	Single-threaded Echo-Server als Beispiel	208
7.6.4	Multi-threaded Echo-Server als Beispiel	212
7.7	Weiterführende Programmierkonzepte und -mechanismen	215
	Literatur	217
8	Schlussbemerkung	219
	Literatur	220
	Anhang: TCP/IP-Konfiguration in Betriebssystemen	221
	Stichwortverzeichnis	229