

Inhaltsverzeichnis

Vorwort	vii
Tabellenverzeichnis	xiii
Abbildungsverzeichnis	xv
 Die Exposition	 1
1 Einleitung	3
1.1 Ziel und Beitrag der Arbeit	5
1.2 Struktur der Arbeit	6
 2 Das Transmission Control Protocol: die Grundlagen	 9
2.1 Einführung in TCP: Philosophie und Protokolldesign	11
2.1.1 TCP's fundamentale Funktionen	11
2.1.2 Struktur eines TCP-Segments	12
2.1.3 Verbindungsaufbau und -abbau	18
2.2 Zuverlässiger Datentransferdienst	21
2.2.1 Retransmission Timeout Berechnung	24
2.2.2 Round-Trip Time Measurement	27
2.2.3 Protect Against Wrapped Sequence Numbers	29
2.3 Flow Control und Window Management	30
2.3.1 Das Silly Window Syndrom	31
2.3.2 Delayed Acknowledgments	33
2.3.3 Clark und Nagle	34
2.4 Congestion Control und Loss Recovery	36
2.4.1 Slow Start und Congestion Avoidance	38
2.4.2 Fast Retransmit und Fast Recovery	40
2.4.3 Beispiel und Zustandsdiagramm	42
2.5 Zusammenfassung	44
 3 Methodik zur TCP-Performance-Analyse	 45
3.1 Evaluierungsumgebung	45
3.1.1 Realisierung und Implementierung	46
3.1.2 Netzwerktopologie und Setup	47

3.2	Tools zur Evaluierung einer TCP-Verbindung	51
3.2.1	Generierung eines TCP-Datentransfer	51
3.2.2	Erfassung einer TCP-Verbindungen	52
3.2.3	Darstellung einer TCP-Verbindungen	53
3.3	TCP Time-Sequence-Graphen	54
3.3.1	Slow Start und Congestion Avoidance	54
3.3.2	Fast Retransmit und Fast Recovery	57
3.3.3	Retransmission Timeout	58
3.4	Typische Fehlerquellen	60
4	Congestion Control und Loss Recovery in einem modernen TCP-Stack	67
4.1	Congestion Control und Loss Recovery Erweiterungen	69
4.1.1	Limited Transmit	69
4.1.2	Appropriate Byte Counting	71
4.1.3	NewReno TCP	72
4.2	SACK-basiertes Congestion Control und Loss Recovery	76
4.2.1	TCP's Selective Acknowledgment Options	76
4.2.2	SACK-basierte Loss Recovery	79
4.2.3	Forward Acknowledgement	83
4.3	Experimentelle Congestion Control und Loss Recovery Erweiterungen	85
4.3.1	Congestion Window Validation	85
4.3.2	Early Retransmit	86
4.3.3	TCP-LCD Algorithmus	87
4.3.4	Rate-Halving	90
4.4	Erkennung und Vermeidung von Spurious Retransmissions	92
4.4.1	Eifel Detection Algorithmus	93
4.4.2	Erkennung mittels Duplicate Selective Acknowledgments	94
4.4.3	Forward RTO-Recovery	96
4.4.4	Eifel Response Algorithmus	98
4.4.5	TCP-NCR Algorithmus	100
4.5	Zusammenfassung	104
	Die Komplikation, die Peripetie und das retardierende Moment	107
5	Das Problem des Packet Reordering	109
5.1	Definition und Metriken	109
5.1.1	Formale Definition des Reordering	109
5.1.2	Metriken zur Quantifizierung des Reordering	111
5.2	Reordering ist kein pathologisches Phänomen	113
5.2.1	Aktive Messungen	114
5.2.2	Passive Messungen	116
5.3	Ursachen des Reordering im Internet	117

5.4	Auswirkungen des Reordering auf TCP	120
5.4.1	Reordering auf dem Vorwärtspfad	121
5.4.2	Beispiele für Reordering auf dem Vorwärtspfad	124
5.4.3	Reordering auf dem Rückwärtspfad	127
5.4.4	Mögliche Lösungsansätze	129
5.5	Erkennung und Quantifizierung des Reordering	130
5.5.1	Erkennung durch Zählen von Duplicate Acknowledgments	130
5.5.2	Erkennung anhand von Selective Acknowledgments	132
5.5.3	Erkennung anhand von Duplicate Selective Acknowledgments	134
5.5.4	Erkennung anhand der TCP Timestamps Option	137
5.6	Proaktive und reaktive Maßnahmen gegenüber Reordering	138
5.6.1	Practices for TCP Senders in the Face of Reordering	139
5.6.2	Delayed Congestion Response TCP	140
5.6.3	Reordering-Robust TCP with DSACK	141
5.6.4	Algorithmus nach Leung und Ma	143
5.6.5	TCP for Persistent Reordering	143
5.6.6	Maßnahmen des Linux TCP-Stack	144
5.7	Zusammenfassung	147
6	TCP-aNCR - An Adaptive Non-Congestion Robustness for TCP	149
6.1	Erkennung und Quantifizierung des Reordering	150
6.1.1	Grundlegendes Konzept	151
6.1.2	Der Algorithmus	152
6.1.3	Protokollschritte im Detail	155
6.1.4	Diskussion des Algorithmus	157
6.2	Proaktive Maßnahmen gegenüber Reordering	162
6.2.1	Grundlegendes Konzept	163
6.2.2	Der Algorithmus	164
6.2.3	Protokollschritte im Detail	168
6.2.4	Diskussion des Algorithmus	170
6.3	Beispiele des TCP-aNCR Algorithmus	179
6.4	Interoperabilität und Sicherheitsaspekte	186
6.5	Zusammenfassung	189
7	Analyse, Evaluation und Bewertung	191
7.1	Implementierung des TCP-aNCR Algorithmus	191
7.2	Messszenarien und Evaluierungsumgebung	193
7.2.1	Metriken und Parameter zur Evaluation	194
7.2.2	Realisierte Messszenarien	194
7.2.3	Datenverkehrsmodell	197
7.2.4	Netzwerktopologie und Setup	198
7.3	Evaluation des TCP-aNCR Algorithmus	200
7.4	Zusammenfassung	222

Das Dénouement	227
8 Zusammenfassung	229
8.1 Bewertung der Ergebnisse	229
8.2 Ausblick auf zukünftige Arbeiten	230
A Linux TCP und Reordering	233
A.1 Erkennung und Quantifizierung des Reordering	233
A.2 Reaktive Maßnahmen gegenüber Reordering	235
A.3 Proaktive Maßnahmen gegenüber Reordering	236
B Tools and Software	239
B.1 UMIC-Mesh.net – Ein hybrides Wireless Mesh Network Testbed	239
B.2 Flowgrind – Ein Performance-Analyse-Tool für TCP	243
C RFCs zu TCP im Überblick	251
C.1 Elementare Funktionalität	252
C.2 Empfohlene Erweiterungen	254
C.3 Experimentelle Erweiterungen	259
C.4 Unterstützende Dokumente	263
Glossar	271
Akronyme	287
Literatur	291
Lebenslauf	315