

Überlebensfähige Sicherheitskomponenten für Hochgeschwindigkeitsnetze

Entwurf und Realisierung am Beispiel einer
Packet Screen

als Dissertation zur Erlangung des Doktorgrades
am Fachbereich Informatik der Universität Hamburg

vorgelegt von

Carsten Benecke

aus Hamburg

Hamburg 2002

Genehmigt vom Fachbereich Informatik der Universität Hamburg

auf Antrag von Prof. Dr. B.E. Wolfinger, Universität Hamburg
 Prof. Dr. K. Brunnstein, Universität Hamburg
 Dr. H.-J. Mück, Universität Hamburg

Hamburg, den 12. Februar 2002
(Tag der Disputation)

Prof. Dr. H.-S. Stiehl
Dekan

Berichte aus dem Forschungsschwerpunkt Telekommunikation
und Rechnernetze

Band 3

Carsten Benecke

**Überlebensfähige Sicherheitskomponenten
für Hochgeschwindigkeitsnetze**

Entwurf und Realisierung am Beispiel einer Packet Screen

Shaker Verlag
Aachen 2002

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Benecke, Carsten:

Überlebensfähige Sicherheitskomponenten für Hochgeschwindigkeitsnetze :
Entwurf und Realisierung am Beispiel einer Packet Screen /

Carsten Benecke. Aachen : Shaker, 2002

(Berichte aus dem Forschungsschwerpunkt Telekommunikation
und Rechnernetze; Bd. 3)

Zugl.: Hamburg, Univ., Diss., 2002

ISBN 3-8265-9994-2

Copyright Shaker Verlag 2002

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen
oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungs-
anlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 3-8265-9994-2

ISSN 1439-3573

Shaker Verlag GmbH • Postfach 1290 • 52013 Aachen

Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9

Internet: www.shaker.de • eMail: info@shaker.de

Vorwort des Herausgebers

Die zunehmende Bedeutung sicherer Informatiksysteme — und dabei insbesondere sicherer Kommunikations- und Rechnernetze (wie dem Internet) — wird inzwischen allseits anerkannt. Gleichwohl besitzen die gegenwärtig im Einsatz befindlichen Rechner und Netze nicht selten noch gravierende Sicherheitslücken.

Noch ungünstiger stellt sich die Situation dar, wenn nicht nur sichere sondern überlebensfähige Informatiksysteme gefordert werden, d.h. Systeme, die gleichzeitig ein hohes Maß an Sicherheit (z.B. Abhörsicherheit), Leistungsfähigkeit (z.B. garantierte Dienstgüte bei Rechnernetzdiensten) und Zuverlässigkeit (z.B. Verfügbarkeit) aufweisen sollen.

Die vorliegende Arbeit behandelt die Entwicklung überlebensfähiger Informatiksysteme am Beispiel von Sicherheitskomponenten in Netzen und legt dabei besonderes Gewicht auf eine ganzheitliche Sicht der Entwurfsproblematik für überlebensfähige Systeme. Im Gegensatz zu bisherigen Arbeiten, bei denen in der Regel nur Teilaspekte Berücksichtigung finden, schenkt der Autor den Aspekten Sicherheit, Leistungsfähigkeit und Zuverlässigkeit gleichermaßen Beachtung.

Dabei erzielt Herr Benecke wertvolle Beiträge in sehr unterschiedlichen Bereichen, u.a. durch

- Entwicklung von Richtlinien für einen, auf eine ganzheitliche Sicht ausgerichteten, Systementwurf mit Erarbeitung entwurfsunterstützender (semi-)formaler Beschreibungs- und Analysetechniken (insbesondere durch Erweiterung von “Message Sequence Charts — MSCs” sowie der Sprache “Specification and Description Language — SDL”);
- prototypische Umsetzung des vorgeschlagenen neuen Entwurfskonzeptes unter exemplarischer Anwendung der erarbeiteten Beschreibungs- und Analysehilfsmittel beim Entwurf einer parallelen “Packet Screen” (als einer zentralen Sicherheitskomponente in Rechnernetzen);
- analytische Modelle zur validen Leistungs- und Zuverlässigkeitsbewertung von parallelen “Packet Screens”;
- Entwurf und Implementierung eines leichtgewichtigen und sicheren Gruppenmanagement-Protokolls.

Ein besonderes Merkmal der Arbeit liegt dabei in ihrer starken Praxisorientierung und der konsequenten Nutzung der langjährigen Erfahrungen des Autors (z.B. basierend auf seinen vielbeachteten Projektarbeiten im “DFN-Firewall-Labor” des Fachbereichs Informatik der Universität Hamburg), um die Praxistauglichkeit von entwurfsunterstützenden Beschreibungs-, Modellierungs- und Analysetechniken anhand der konkreten Implementierung von Prototypen für Sicherheitskomponenten kritisch zu überprüfen und zu beurteilen.

Daher dürfte die vorliegende Arbeit nicht nur für Systementwickler sondern auch für Betreiber sicherer Kommunikationsnetze von beträchtlichem Interesse sein und (hoffentlich) wertvolle Anstöße für weitere FuE-Aktivitäten auf dem zur Zeit noch weiten Weg zu hochgradig überlebensfähigen Informatiksystemen liefern.

Hamburg, im Februar 2002

Bernd E. Wolfinger

Kurzfassung

Überlebensfähige Systeme zeichnen sich durch eine besonders gute Resistenz gegen verschiedenartige Störungen aus. Zu diesen Störungen gehören beispielsweise Ausfälle einzelner Komponenten aufgrund von Software- oder Hardwarefehlern oder auch Beeinträchtigungen, die durch Angriffe von Computer-Kriminellen verursacht werden.

Um ein überlebensfähiges System zu entwerfen und zu realisieren, sind daher eine Vielzahl verschiedenartiger Analyse- und Syntheseschritte erforderlich. Das Ziel der (wiederholten) Systemanalyse ist es, die unterschiedlichen Bedrohungen, Fehlermöglichkeiten und Leistungsengpässe zu identifizieren. Die Bewertung vorhandener Systeme erfordert daher in der Regel die Anwendung von Methoden aus verschiedenen Disziplinen, insbesondere aus den Bereichen des Fehler-, Leistungs-, und Sicherheitsmanagements. Bei der darauf folgenden Synthese müssen die gefundenen Defizite ausgeräumt werden. Das Hauptproblem bei der Synthese von überlebensfähigen Systemen liegt in der Notwendigkeit zur gleichzeitigen Optimierung teilweise gegenläufiger Ziele, die dazu führen können, daß beispielsweise die Erhöhung der Leistungsfähigkeit zu neuen, bisher nicht vorhandenen Sicherheitsrisiken führt.

Zur Zeit gibt es leider kaum praktische Erfahrungen und konkretes Methodenwissen im Entwurf von überlebensfähigen Systemen. In der vorliegenden Arbeit soll daher am Beispiel der Transformation einer wichtigen bekannten Firewall-Komponente untersucht werden, ob durch die sukzessive Verbesserung eines Systems mit Hilfe von Standardmethoden der Systemanalyse und -synthese schließlich ein überlebensfähiges Pendant entwickelt werden kann. In Fällen, bei denen vorhandene Analysemethoden/-werkzeuge nicht ausreichen, um die erforderlichen Untersuchungen durchzuführen, werden entsprechende Erweiterungen motiviert und angewendet. In der Arbeit werden insbesondere Erweiterungen an Meßwerkzeugen und Meßmethoden vorgestellt sowie verschiedene formale Spezifikations- und Analysemethoden miteinander kombiniert.

Neben der Diskussion der Defizite des vorhandenen Systems und der notwendigen Verbesserungen wird auch der "Tradeoff" zwischen verschiedenen Entwurfsalternativen diskutiert bzw. mit Hilfe von analytischen Modellberechnungen auf Basis von empirischen Meßergebnissen verdeutlicht. Die abschließende Analyse des Prototypen einer "überlebensfähigen Packet Screen" zeigt, daß Leistungsfähigkeit, Verfügbarkeit und Sicherheit durchaus gleichzeitig erreichbar sind und die gewählte Methode der "iterativen Adaption" erfolgreich für die Transformation eines Systems in ein überlebensfähiges Pendant eingesetzt werden kann.

Abschließend wird anhand eines Architekturbeispiels für ein überlebensfähiges Perimeternetz der Transfer von Ergebnissen auf andere wichtige Netzkomponenten vorgestellt. Es wird somit gezeigt, daß die am Beispiel gewonnenen Ergebnisse vielfach anwendbar sind, insbesondere auf andere Sicherheitskomponenten in Hochgeschwindigkeitsnetzen.

Abstract

Survivable systems are known to be resistant to different kinds of problems. Among these are failures due to software or hardware faults, but also attacks caused by computer criminals.

The design and implementation of survivable systems therefore requires a variety of different steps to support system analysis and synthesis. It is the goal of a repeatedly applied analysis of the system to identify all kinds of threats, errors, and performance bottlenecks. Thus, the evaluation of a system usually requires the combination of failure-, performance- and security management. During the synthesis of a survivable system in its design phase all recognized deficiencies have to be removed. The main problem of building survivable systems is the optimization of more than one competitive goals. For example, increasing the performance of the system may result into additional security risks.

At present, we are not only lacking experience in the design of survivable systems, but we are also in need of general methods which will ultimately transform a system into a survivable counterpart. This thesis demonstrates the migration of an important firewall building block, i.e. a packet screen, to a survivable system as a case study. The chosen method will adapt the packet screen step by step to the analyzed requirements. Doing this, standard methods will be applied whenever possible. These methods will be combined and extended whenever needed. It will be shown that also the extension of tools for the performance analysis is necessary. Moreover, it will be stated that a general security analysis requires the combination of several formal methods to increase the insight into the analyzed systems.

The tradeoff between design alternatives will be discussed as well and exemplified by means of analytical models and empirical load measurements. A final analysis will demonstrate that the chosen method results in a survivable packet screen: Performance, reliability and security can indeed be combined in one system.

Finally, an example of a survivable perimeter network will show that the results can be transferred to many other security building blocks for high speed networks. Thus, we conclude that our new method of "step by step adaption" as well as the chosen formal techniques are applicable to many different subsystems of communication networks.

Danksagung

An dieser Stelle möchte ich mich für die vielfältige Unterstützung bedanken, die mir die Anfertigung dieser Dissertationsschrift erst ermöglichte.

Mein ganz besonderer, herzlicher Dank gilt Herrn Dr. Hans-Joachim Mück, der mir die langjährige Forschung in den zwei Drittmittelprojekten “Firewall-Labor für Hochgeschwindigkeitsnetze” (DFN-FWL) und “Sicherheit in ATM-Netzen” am Rechenzentrum des Fachbereichs Informatik der Universität Hamburg ermöglichte.

Für die überaus gute, fruchtbare Zusammenarbeit möchte ich den folgenden ehemaligen Kollegen aus dem Projekt DFN-FWL und den Partnerprojekten (DFN-CERT und DFN-PCA) danken, die mir immer mit Rat und Tat zu Seite standen: Dr. Uwe Ellermann (DFN-FWL), Stefan Kelm (DFN-PCA), Britta Liedtke (DFN-PCA), Wolfgang Ley (DFN-CERT), Les Schäfer (DFN-CERT) und Dr. Klaus-Peter Kossakowski (DFN-CERT), sowie Dr. Marcus Pattloch (DFN-Verein).

Für die fleißige Arbeit im Firewall-Labor danke ich Olaf Gellert, Stephan Holst, Axel Großklaus, Gregor Goldbach und Michael Krooß. Für die gute Kooperation danke ich meinen Kollegen vom Rechenzentrum, insbesondere dem Operating-Team, Reinhard Zierke und Andreas Lukas.

Für die vielen Hinweise und Anregungen zur schriftlichen Fassung dieser Arbeit danke ich Dr. Uwe Ellermann, Dr. Klaus-Peter Kossakowski, Wolfgang Ley, Stephan Kelm, Dr. Hans-Joachim Mück, Heino Peters und Dr. Michaela Welk.

Für ihre Betreuung und die Bereitschaft ein Gutachten zu dieser Arbeit anzufertigen, danke ich Prof. Dr. Bernd E. Wolfinger und Prof. Dr. Klaus Brunnstein sowie Dr. Hans-Joachim Mück. Besonderen Dank schulde ich Prof. Dr. Bernd E. Wolfinger für seine jahrelange Betreuung meiner akademischen Arbeiten (Studien-, Diplomarbeit und Dissertationsschrift) und die hiermit verbundenen, vielen hilfreichen Hinweise und Anregungen, insbesondere im Bereich der Modellbildung.

Abschließend danke ich meiner Familie für ihre Geduld mit mir und meiner Arbeit.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ziele der Arbeit	2
1.2	Weitere Vorgehensweise	4
2	Überlebensfähige Systeme: Notwendigkeit, Entwurfsempfehlungen und Mechanismen	7
2.1	Risiken durch die gängige Entwurfspraxis	8
2.1.1	Fallbeispiel 1: Priorisierung des Leistungsmanagements	8
2.1.2	Fallbeispiel 2: Ignorieren des Sicherheitsmanagements	9
2.2	Überlebensfähige Systeme	10
2.2.1	Definition: Überlebensfähigkeit	11
2.2.2	Anforderungen an überlebensfähige Systeme	12
2.2.3	Primäre Eigenschaften von überlebensfähigen Systemen	15
2.2.4	Entwurfsempfehlungen für überlebensfähige Systeme	16
2.3	Überlebensfähigkeit am Beispiel der Packet Screen	18
2.3.1	Beispiel: Überlebensfähige Firewall-Komponenten	19
2.3.1.1	Firewalls für Zugriffskontrolle und Audit	20
2.3.1.2	Bedeutung der Firewalls für die Netzsicherheit	22
2.3.1.3	Bedeutung der Packet Screen für Firewall-Architekturen	23
2.3.2	Notwendige Reduzierung der Komplexität	27
2.4	Zusammenfassung	28
3	Engpässe durch Packet Screens in Hochgeschwindigkeitsnetzen	31
3.1	Leistungsbewertung von Rechen- und Kommunikationssystemen	32

3.2	Szenarien, Last-, Konfigurations- und Leistungskenngrößen für die Bewertung von Packet Screens	33
3.2.1	Szenarien für die Leistungsbewertung von Packet Screens	35
3.2.1.1	Allgemeiner Versuchsaufbau	35
3.2.1.2	Versuchsaufbau für Fast Ethernet-Messungen	38
3.2.2	Last-, Konfigurations- und Leistungskenngrößen für die Bewertung von Packet Screens	39
3.2.2.1	Lastkenngrößen	42
3.2.2.2	Konfigurationskenngrößen	44
3.2.2.3	Leistungskenngrößen	44
3.3	Anforderungen an Meßwerkzeuge	45
3.3.1	Statistische Absicherung	46
3.3.2	Trennung von Konfigurations- und Meß-Dienstzugangspunkt	47
3.3.3	Synchronisation mehrerer Meßinstanzen	47
3.4	Wahl geeigneter Parameter und Konfigurierung der Meßsysteme	48
3.4.1	Verwendete Hardware für die Meßdurchführung	48
3.4.2	Größe der “Service Data Unit” auf der Transportebene	49
3.4.3	Puffergröße des “Service Access Points”	50
3.4.4	Transportprotokoll	50
3.4.5	Meßdauer, Wiederholungen und Vertrauensintervalle	51
3.4.6	Anzahl der Filterregeln	52
3.5	Theoretischer Durchsatz und Referenzmessungen	53
3.6	Packet Screen als Engpaß in Hochgeschwindigkeitsnetzen	55
3.6.1	Verarbeitung von Filterregeln bei Packet Screens	55
3.6.2	Leistungsbewertung einer Fast Ethernet-basierten Packet Screen	59
3.7	Zusammenfassung	60
4	Parallele Packet Screen für Hochgeschwindigkeitsnetze	63
4.1	Leistungsanforderungen an eine Packet Screen für Hochgeschwindigkeitsnetze	64
4.2	Leistungssteigerung durch parallele Protokollverarbeitung	66
4.2.1	Gängige Klassifikationen im Bereich der Parallelverarbeitung	66

4.2.1.1	Klassifikation bezüglich Rechnerarchitektur	67
4.2.1.2	Klassifikation bezüglich Zuordnung	67
4.2.2	Klassifikation bezüglich Protokollverarbeitung	69
4.3	Entwurf einer parallelen Packet Screen	74
4.3.1	Statische und dynamische Parallelität	76
4.3.2	Parallele Packet Screen auf Basis der Paketparallelität	78
4.3.3	Lastverteilung bei der Paketparallelität	79
4.3.4	Dezentrale Zuordnung der Pakete zu Prozessoren	80
4.3.5	Dezentrale Selektion von Paketen	81
4.3.6	Selektion innerhalb des IP-Filter Moduls	85
4.3.7	Selektion innerhalb des IP-Kernel Moduls	86
4.3.8	Selektion innerhalb der Geräte-Treiber	86
4.4	Realisierung von Hochleistungs-Packet Screens	87
4.4.1	Parallele Packet Screen für "shared" Ethernet	87
4.4.2	Verbesserte parallele Packet Screen für "switched" Ethernet	89
4.5	Leistungsbewertung der parallelen Packet Screen	90
4.5.1	Parallele Packet Screen in "shared" Fast Ethernet Netzen	92
4.5.1.1	Messung ohne konfigurierte Filterregeln	92
4.5.1.2	Messung mit 50 konfigurierten Filterregeln	94
4.5.1.3	Bidirektionale Messung mit zwei Lastgeneratoren	94
4.5.2	Parallele Packet Screen in "switched" Fast Ethernet Netzen	98
4.5.3	Zusammenfassung: Meßergebnisse Parallele Packet Screen	99
4.6	Analytische Bewertung der Parallelen Packet Screen	100
4.6.1	Modell einer parallelen Packet Screen	100
4.6.2	Analytische Berechnung des Paketdurchsatzes	103
4.6.3	Bestimmung der Werte für eine realistische Modellparametrisierung	105
4.6.4	Modellbasierte Analyse des Gewinns durch Parallelverarbeitung	108
4.6.5	Auswirkungen des Einsatzes schnellerer Hardware	109
4.6.6	Auswirkungen des Einsatzes aufwendiger Filteroperationen	110
4.6.7	Auswirkungen der Verbindungsparallelität	112
4.7	Zusammenfassung	113

5	Variationen des Selektionsalgorithmus als Basis für eine flexible Parallelverarbeitung	115
5.1	Unterschiedliche Leistungsfähigkeit der Prozessoren	115
5.2	Dynamisches Filtern und Verbindungsparallelität	117
5.2.1	Dynamisches Filtern für "Remote Procedure Call"-Zugriffskontrolle .	118
5.2.2	Dynamisches Filtern für "Intrusion Detection System"-Rückkoppelung	119
5.2.3	Anforderung: Unterstützung dynamischer Filter	121
5.2.4	Algorithmus zur Unterstützung dynamischer Filter	121
5.2.5	Weitere Anwendungsmöglichkeiten der Verbindungsparallelität . . .	124
5.2.6	Mehrstufige Selektion	124
5.3	Zusammenfassung	125
6	Ausfallsicherheit der parallelen Packet Screen	127
6.1	Zuverlässigkeit der Packet Screen und Auswirkungen durch den Ausfall von Prozessoren	128
6.1.1	Zuverlässigkeit und Kenngrößen	128
6.1.2	Modelle zur Fehleranalyse	129
6.1.3	Ausfall einer Instanz der parallelen Packet Screen	131
6.2	Fehlermodell und Fehlersemantik	134
6.2.1	Fehlermodell für die parallele Packet Screen	134
6.2.1.1	Potentielle Hardware- und Softwarefehler	135
6.2.1.2	Fehlersemantik der parallelen Packet Screen	137
6.2.2	Bewertung der Fehlersemantik	137
6.3	Anforderungen an eine fehlertolerante Packet Screen	138
6.3.1	Anforderungen an die Fehlerbehandlung	139
6.3.2	Weitere Anforderungen	139
6.4	Abhängigkeiten der Mechanismen im Bereich der fehlertoleranten Systeme	140
6.5	Verwaltung von Gruppenzugehörigkeit	143
6.6	Das zeitgesteuerte, asynchrone Systemmodell	144
6.7	Zusammenfassung	145

7	Leichtgewichtiges, sicheres Gruppenmanagement-Protokoll	147
7.1	Funktionsbedingter Aufwand beim Gruppenmanagement	148
7.2	Entwurf eines leichtgewichtigen Gruppenmanagement-Protokolls	152
7.2.1	Dienstspezifikation Gruppenmanagement-Dienst	153
7.2.2	Technische Annahmen über die Umgebung	157
7.2.3	Automatische Erkennung eines Ausfalls	157
7.2.3.1	“Heartbeat”-Signale als “Multicasts”	159
7.2.3.2	Heuristik für die Ausfallserkennung	159
7.2.4	Semiformale Beschreibung der Funktionsweise des Protokolls	161
7.2.4.1	Verhalten bei erkennbaren Fehlern	162
7.2.4.2	Erhöhen der Robustheit I	163
7.2.4.3	Aufnahme in die Gruppe	163
7.2.4.4	Erhöhen der Robustheit II	164
7.2.5	Protokollspezifikation	166
7.2.5.1	Protokollautomat	166
7.2.5.2	Zeitlicher Ablauf	170
7.2.5.3	“Protocol Data Unit”-Format	174
7.2.5.4	SDL-Spezifikation	175
7.3	Validation vs. Sicherheitsanalyse	181
7.3.1	Validation des leichtgewichtigen Gruppenmanagement-Protokolls	181
7.3.2	Vollständigkeit	181
7.3.2.1	Syntaktische Vollständigkeit	182
7.3.2.2	Semantische Vollständigkeit	182
7.3.3	Keine Verklemmungen (“Deadlocks”)	182
7.3.4	Keine “Livelocks”	183
7.3.5	Kein ungewollter Abbruch	183
7.4	Erreichte Zuverlässigkeit	183
7.4.1	Verfügbarkeit der parallelen Packet Screen	184
7.4.2	Verfügbarkeit bei Leistungsvorgaben	184
7.5	Zusammenfassung	186

8	Sicherheitsanalyse für das leichtgewichtige GM-Protokoll	189
8.1	Sicherheitsrelevante Annahmen über die Umgebung	190
8.2	Sicherheitsanforderungen an das Gruppenmanagement-Protokoll	192
8.3	Zusätzliche Entwurfsschritte für Protokolle in überlebensfähigen Systemen	193
8.4	Sicherheitsanalyse für Protokolle	194
8.4.1	BAN-Logik und deren Weiterentwicklungen	195
8.4.1.1	Schlußregeln der BAN-Logik	196
8.4.1.2	Wichtige Ergänzungen der BAN-Logik	198
8.4.2	Automatenmodell-basierte Ansätze	199
8.4.3	Hybride Methoden	200
8.4.4	Offene Probleme und Schlußfolgerungen	200
8.5	Sicherheitsprobleme, Bedrohungspotential und erforderliche Sicherheitsme- chanismen	203
8.5.1	Zu SA1: Nur autorisierte Prozessoren dürfen in die Gruppe aufge- nommen werden	204
8.5.2	Zu SA2: Autorisierten Prozessoren darf die Aufnahme in die Gruppe nicht verweigert werden	207
8.5.3	Zu SA3: Die Integrität des Gruppenkonsens muß gewährleistet sein	212
8.5.4	“Denial of Service“-Angriff durch IP-Dateneinheiten	213
8.5.4.1	“Denial of Service“-Angriff: Erfolgsaussichten und Auswir- kungen	214
8.5.4.2	Worst-Case Szenario	216
8.5.5	“Denial of Service“-Angriff auf das Gruppenmanagement-Protokoll .	217
8.5.5.1	Erkennen des “Denial of Service“-Angriffs	219
8.5.5.2	Abwehren des Angriffs	220
8.6	Zusammenfassung der zusätzlichen Mechanismen für ein sicheres GM-Protokoll	223
8.6.1	“Keyed Message Authentication Code” für Integrität und Authentizität	223
8.6.2	Streng monoton steigende Sequenznummer	223
8.6.3	Authentifikationsanfrage an Bewerber	224
8.6.4	Alumni-Liste und stabiler Speicher	224
8.6.5	Zusätzlicher Zustand für DoS-Angriffe	224
8.7	Zusammenfassung	225

9	Implementationsaspekte und Test des Gruppenmanagement-Protokolls	227
9.1	Eingesetzte Hardware und Software	228
9.2	Maßnahmen für geringen Speicherverbrauch	229
9.2.1	Erkennen und Vermeiden von Speicherlecks	229
9.2.2	Zusätzliche Maßnahmen	230
9.3	Identifikation der Prozessoren	231
9.4	Analyse eintreffender Dateneinheiten	231
9.5	Implementation der Zeitgeber	232
9.6	Test des Timings	233
9.6.1	Verwendung von “Tracing”-Techniken	234
9.6.2	Unterstützung der Fehlersuche	234
9.6.3	Konsequente Modularisierung	235
9.7	Host-Sicherheit	235
9.8	Leistungsbewertung einer abgesicherten parallelen Packet Screen	236
9.9	Zusammenfassung	237
10	Ein überlebensfähiges Perimeternetz für Hochgeschwindigkeitsnetze	239
10.1	Eine Beispiellarchitektur	240
10.2	Überlebensfähige Firewall-Komponenten	240
10.2.1	Bastionen und Proxies	240
10.2.2	“Virtual Private Network”-Gateways	242
10.2.3	“Sniffer” für “Intrusion Detection”-Systeme	242
10.2.4	Audit-Trails	243
10.3	Überlebensfähige Informationsdienste	245
10.3.1	DNS-Service	245
10.3.2	HTTP-Server	246
10.4	Zusammenfassung	247
11	Resümee und Ausblick	249
11.1	Zusammenfassung der Ergebnisse	249
11.2	Offene Fragestellungen	252

A	Spezifikation mit “Message Sequence Charts” und “Specification and Description Language”	255
A.1	Einführung in die graphische Notation der “Message Sequence Charts” und der “Specification and Description Language”	255
A.1.1	Einführung in “Message Sequence Charts”	256
A.1.2	Einführung in die “Specification and Description Language”	257
A.2	“Message Sequence Charts” als standardisierte, semiformale Methode zur Sicherheitsanalyse von Protokollen	261
A.2.1	“Message Sequence Charts” und BAN-Logik	263
A.2.2	“Message Sequence Charts” und “Strands”	264
A.2.3	“Message Sequence Charts” und Verarbeitungskosten	266
A.2.4	Gleichzeitige Integration mehrerer Annotationen	272
A.3	Zusammenfassung	274
B	Protokollspezifikation mit der “Specification and Description Language”	275
B.1	Hilfsspezifikation: “Broadcast“-Medium	275
B.2	Spezifikation eines Gruppenmanagement-Protokolls	277
B.2.1	Systemstruktur	277
B.2.2	Blockstruktur	277
B.2.3	Prozeßstruktur (Protokollinstanz)	279
B.2.4	Prozeßstruktur (Admin-Prozeß)	293
B.2.5	Diverse Hilfsfunktionen	293
B.3	Zusammenfassung	303
C	Durchführung von Messungen und Erweiterungen an Meßwerkzeugen und Gerätetreibern	305
C.1	Fallstricke bei der Meßdatenerfassung	305
C.1.1	Fehlerquelle: “Plug and Play“-Netzanschluß	306
C.1.2	Fehlerquelle: falsche und ungleichmäßige Konfigurierung von Monitor, Lastgenerator und der Systeme unter Last	306
C.1.3	Fehlerquelle: Störlast im Testnetz	307
C.1.4	Fehlerquelle: Anzahl der Prozessoren auf den Meßsystemen	307
C.2	Erweiterungen an den Meßwerkzeugen	309

C.2.1	Statistische Absicherung	309
C.2.2	Trennung der Wegwahl für gleichzeitige Kommunikationsbeziehungen	309
C.2.3	Synchronisation mehrerer Meßsysteme	310
C.2.4	Leistungsvorgaben	311
C.3	Erweiterungen an den Gerätetreibern	312
C.3.1	Erweiterungen für die verteilte Selektion	313
C.3.2	Erweiterungen an der Flußkontrolle	316
C.3.3	Erweiterungen an der Fehlererkennung	319
C.4	Zusammenfassung	322
	Abkürzungsverzeichnis	323
	Literaturverzeichnis	327