Network Resilience

Tsvetko Tsvetkov
Seminar Future Internet, SS 2010
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
tsvetko.tsvetkov@mytum.de

Kurzfassung

Dass Widerstandsfähigkeit des Internets ein aktuelles und wichtiges Thema ist, zeigen die wissenschaftlichen Diskussionen sowie die Forschungsprojekte auf nationaler und internationaler Ebene. Die Nutzung des Internets hat sich in den letzten vier Jahrzehnten sehr stark verändert. Die damaligen Entwickler konnten die rapide Weiterentwicklung nicht voraussehen und es ist nicht überraschend, dass viele an das Internet gestellte Anforderungen nicht mehr oder nur noch unzureichend erfüllt werden können. In diesem Paper werden verschiedene bekannte und zukünftige Technologien betrachtet, die viele Schwachstellen abdecken und somit die Netzwerk-Resilienz erhöhen.

Schlüsselworte

Future Internet, Netzwerk, Resilience, Fehlertoleranz, Robustheit, Sicherheit, Zuverlässigkeit

1. Einführung

Das Internet ist mittlerweile zu einem globalen und sozialen Phänomen geworden. Die Gesellschaft hängt für nahezu jeden Aspekt des täglichen Lebens von Netzwerken ab. Täglich werden Dienste wie World Wide Web, E-Mail, Dateiverwaltung, Chat verwendet und dabei ist es wichtig in nicht vorhersehbaren Situationen, die beispielsweise durch Störungen zwischen Switches/Routern oder komplett Ausfällen von Netzknoten entstehen, adäquat reagieren zu können. Aus Sicht eines Anbieters hängt die Stabilität des Dienstes von der Qualität und Absicherung des Servers ab. Hier sind neben der reinen Serverleistung Faktoren wie Standort, Sicherungsmaßnahmen und Notfallpläne (z.B. Protection Switching) wichtig.

Die Erhöhung der Widerstandsfähigkeit des Internets ist auch ein zentrales Thema für Organisationen wie die *IFIP Working Group* und der EU-Kommission durch das *ResumeNet* [1] Projekt geworden. Im Weiteren werden die Definition der Netzwerk-Resilienz (Abschnitt 2) und die Ursachen, warum das Internet nicht widerstandsfähig genug ist (Abschnitt 3), erwähnt. Zusätzlich werden im Abschnitt 4 bekannte Techniken zur Erhöhung der Resilienz betrachtet. Abschnitt 5 behandelt zukünftige Ansätze und die Ausarbeitung wird schließlich mit den Abschnitten 6 und 7 – Zusammenfassung und Literatur-referenzen abgeschlossen.

2. Definition

Der Begriff der Resilienz (engl. Resilience) wird in verschiedenen Bereichen der Wissenschaft verwendet. In der Psychologie wird beispielsweise die Resilienz als die Stärke eines Menschen bezeichnet, Lebenskrisen wie schwere Erkrankungen oder Unfälle zu überwinden.

Im Netzwerkbereich wird der Begriff der Resilienz als die Möglichkeit bezeichnet, ein hohes Leistungsniveau zu erbringen, auch wenn der Normalbetrieb durch unerwartete Faktoren gestört wird. Ein klassisches Beispiel für solche Faktoren sind u.a. Computerwürmer, die den IPv4 Bereich eines Subnetzes scannen um Sicherheitslücken bei den Endgeräten zu finden und somit Zielrechner infizieren zu können. Im Jahre 2003 wurde *W32.Blaster* [2] berühmt, der eine Sicherheitslücke in der RPC-Schnittstelle von Microsoft Windows ausgenutzt und sich somit verbreitet hat. Dabei sollte der Wurm einen gezielten DDoS-Angriff gegen die Updateseiten von Microsoft starten.

Ein weiteres Beispiel wären Topologieänderungen des Netzes. Speziell in geswitchten Umgebungen sollte man auf der Sicherungsschicht (ISO-OSI Modell) zu jedem möglichen Ziel immer nur einen Pfad haben, da ansonsten Datenpakete mehrmals eintreffen können und somit Fehlerfunktionen beim Empfänger auslösen können.

Man muss allerdings betonen, dass Resilienz als Obermenge von Begriffen, wie Überlebensfähigkeit (engl. survivability) und Verfügbarkeit (engl. availability), verwendet werden kann [3]. Dabei bezeichnet „availability“ die Möglichkeit ein bestimmtes System oder einen bestimmten Dienst zu verwenden. Ein 100 % überlebensfähiges System würde zusätzlich einen zeitgenauen Dienst anbieten können. Es würde theoretisch im Falle eines Angriffes oder Störung keine Zeit brauchen um einen „Notfallplan“ zu realisieren. Dabei werden keine Datenpakete, die in der „Planwechselphase“ eintreffen, verloren gehen.

3. Gründe für die nicht-Resilienz

Die Geschichte des Internets beginnt bereits vor 40 Jahren. Durch den Aufbau des ARPAnet, das den Zweck hatte, Universitäten und Forschungseinrichtungen zu vernetzen, wurden die Grundlagen erforscht und entwickelt. Damals wurde das *Network Control Protocol (NCP)* entwickelt, was Adressierung und Transport in sich kombiniert hat. Zusätzlich wurden die Grundbausteine der heutigen Protokolle, wie *File Transfer* und *Remote Login* gelegt. Nachdem man am 1 Januar 1983 [18] den großen Schritt gemacht hat und von NCP nach TCP/IP umgestiegen ist, wurde manchen klar, dass *Flag Days* negative Konsequenzen haben werden. Das Internet hatte damals ca. 400 Netzknoten und es war durchaus möglich einen planmäßigen Netzausfall durchzuführen. Solche Maßnahmen sind heutzutage undenkbar und erschweren somit den Entwurf von neuen Netz-Architekturen.

Allerdings darf man sich nicht darauf verlassen, dass neue Innovationen zu einem widerstandsfähigen und stabilen Internet führen werden. Obwohl TCP einen enormen Fortschritt geleistet hat, hatten die damaligen Netzwerke mit Netzüberlastungen zu kämpfen. Die Netz-Knoten waren komplett überlastet und man konnte keinen zuverlässigen Dienst anbieten. Die Lösung des Problems war die TCP Staukontrolle [23]. Dieser Schritt hat nicht nur einen Ausweg von den Stausituationen gefunden, sondern hat auch neue Hürden für alle zukünftigen Techniken gestellt: Rückwärtskompatibilität und inkrementelle Erweiterbarkeit.

Das Internet wurde so konstruiert, dass es auch nach einem Atomschlag funktionsfähig bleibt. Dass dieser Mythos jedoch nicht den Tatsachen entspricht, wurde bereits in den letzten Jahrzehnten gezeigt. Beispielsweise hatte der Stromausfall im US-Bundesstaat Virginia im Jahre 2001 sogar in Deutschland seine Auswirkungen. Dabei waren zahlreiche Web-Dienste bundesweit nicht erreichbar [22].

Im Folgenden werden weitere Probleme des heutigen Internets geschildert, die zur Störung der Funktionsfähigkeit führen können.

3.1 Routing

Routing wird in der Telekommunikation als das Festlegen von Wegen zur Vermittlung von Paketen in Rechnernetzen bezeichnet. Häufig wird Routing auch als die allgemeine Übermittlung von Nachrichtenpaketen betrachtet, da es meistens eine Zusammensetzung von den Begriffen *Wegbestimmung* und *Weiterleitung* (engl. Forwarding) ist. Der Begriff Forwarding beschreibt einen Entscheidungsprozess eines einzelnen Knotens. Dabei kann ein Datenpaket unter bestimmten Voraussetzungen verworfen werden. Beispielsweise könnte ein Switch das CRC-Verfahren verwenden, um fehlerhafte Pakete zu erkennen und somit deren Weiterleitung zu verbieten (z.B. Store-and-Forward Methode).

Die optimale Wegbestimmung mit Hilfe von Routing-Protokollen ist ein wesentliches Element des heutigen Internets geworden. Das bezieht sich nicht nur auf die Kommunikation innerhalb eines Internetproviders, sondern auch besonders auf den Datenverkehr zwischen *autonomen Systemen (AS)*. Aufgrund des exponentiellen Wachstums der Internet-Nutzung in der Periode zwischen 1980-2005 (Abbildung 1) können heute *Reaktionszeiten* von Routing-Protokollen zwischen Providern sogar Sekunden dauern. Zusätzlich besteht die Gefahr, dass *Fehlerkonfigurationen* von Protokollen schnell Netzwerk-Ausfällen verursachen können.

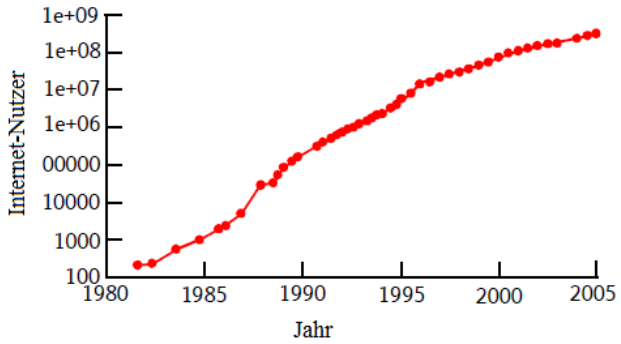

Abbildung 1: Wachstum der Internet-Nutzung, Zitat nach [18]

3.1.1 Fehlerkonfigurationen

Das Internet beinhaltet eine große Anzahl von autonomen Systemen, die unter sich Routing-Informationen austauschen und somit die besten Wege zu verschiedenen Destinationen lernen können. Derzeit wird das *Border Gateway Protocol (BGP)* [4] als Standard-Protokoll bei dem Interdomain-Routing eingesetzt und ist somit ein wichtiger Bestandteil des Alltags geworden. BGP gehört zu der Klasse der *Pfadvektorprotokolle* und dabei tauscht jeder BGP Router mit seinen Nachbarn die gesamte Liste aller *AS-Path* Informationen. Um überhaupt Routing-Informationen austauschen zu können, werden sogenannte *Peering-Sessions* aufgebaut. Dabei werden die kompletten Routing-Tabellen ausgetauscht um somit ein globales Routing zu ermöglichen. Nach dieser Anfangsphase werden meistens Update-Nachrichten verschickt, die eine mögliche Route-Änderung signalisieren sollen. Allerdings besteht die Gefahr, dass ein Router bzw. ein Routerinterface schnell zwischen „up“ und „down“ Zustand wechselt und dabei R*oute-Flapping* [5] erzeugt. Der Grund dafür wäre beispielsweise eine Fehlerkonfiguration, die zur Überschwemmung eines Routers mit Update-Nachrichten führen kann. Dabei wäre eine Überlastung durch ständiges Löschen und Einfügen von Einträgen in der Routing-Tabelle möglich.

Ein ähnliches Phänomen ist auch am 16 Februar 2009 aufgetreten, als die Inkompetenz eines Administrators des tschechischen Providers „SuproNet“ einen globalen *Buffer-Overflow* erzeugt hat [6]. Grund war eine künstliche Verlängerung des AS-Pfades, die bei vielen von Cisco IOS betriebene Router einen Verbindungsabbruch und Neustart verursacht hat.

3.1.2 Reaktionszeiten

Das Ziel der Routing-Protokolle ist nicht nur den optimalen Weg zwischen zwei Netzknoten zu bestimmen, sondern auch möglichst schnell auf Topologieänderungen zu reagieren. Mathematisch ausgedrückt soll ein Protokoll gute Konvergenzeigenschaften aufweisen und zusätzlich Skalierbar sein.

Auch wenn durch die Aufteilung des Internets in autonome Systeme eine bessere Skalierbarkeit erreicht wird, so bleiben noch einige Konvergenzprobleme des BGP bestehen, die zu untersuchen sind. Im Laufe der Zeit wurden unterschiedliche Techniken entwickelt, die die Anzahl von Update-Nachrichten verringern sollen. Beispielsweise kommt *Route-Flap-Damping* zum Einsatz um die Probleme von *Route-Flapping* zu lösen. Dabei verfügt jeder BGP-Router über einen Penalty-Wert, der die Instabilität der Route messen soll. Dieser Wert wird inkrementiert wenn z.B. eine Route ständig zwischen verfügbar und nicht verfügbar pendelt. Es wird dann auf lokaler Ebene entschieden, wann die Route wiederverwendet werden soll (z.B. mit Hilfe eines Schwellwertes). Allerdings kann *Route-Flap-Damping* ein schlechtes Konvergenzverhalten aufweisen und somit Verzögerungen bis zu einer Stunde verursachen [5].

MRAI-Timer ist ein weiterer Mechanismus, mit dem BGP versucht, die Stabilität des Routings zu erhöhen. Jedes Mal wenn ein Router ein Update bezüglich eines bestimmten Präfixes an einen Nachbarn schickt, wird der Timer gestartet. Erst wenn er abgelaufen ist, können wieder Update-Nachrichten geschickt werden. Dadurch werden *Update-Bursts* verhindert. Meistens hat der Timer einen Standardwert von 30 Sekunden. Allerdings haben Tarek Sobh, Khaled Elleithy and Ausif Mahmood [7] gezeigt, dass ein Timer-Wert von 2-5 Sekunden zu Ausfällen und somit zu schlechter Konvergenz führen kann.

3.2 Dienstabhängigkeit im Internet

Einen großen Einfluss auf die Resilienz hat auch die Dienstabhängigkeit im Internet. Beispielsweise werden Google-Dienste von mehreren hundert Millionen Menschen weltweit genutzt. Laut dem Statistischen Bundesamt bestellten 29,5 Millionen Menschen in Deutschland im ersten Quartal 2009 Waren und Dienstleistungen über das Internet. Dabei ist Google eine der Hauptquellen für weitergeleiteten Verkehr im Internet, und ein längerer Ausfall der Seite würde viele Geschäfte beeinflussen.

Ein weiterer Faktor wäre die Positionierung von DNS-Servern im Netz eines Unternehmens. Ein klassisches Beispiel ist die DoS-Attacke auf die Router, die für den Datenverkehr zu den Microsoft-Webseiten zuständig waren. Da im Jahre 2001 die DNS-Server von Microsoft im gleichen IP-Subnetz standen (die IP-Adressen 207.46.138.11, 207.46.138.12, 207.46.138.20 und 207.46.138.21), wurde dadurch der Zugang zu einigen Internet-Angeboten, darunter microsoft.com, microsoft.de, msn.com und WindowsMedia.com teilweise verhindert.

3.3 Sicherheit

Im Internet kommt der Sicherheit des eigenen Netzwerks oder des eigenen Rechners eine besondere Bedeutung zu. Während früher nur Netzwerke der Universitäten und Forschungseinrichtungen anfällig an Angriffen waren, ist heute praktisch jeder der Gefahr ausgesetzt, durch fehlende Sicherheitskonzepte Opfer zu werden - mit oft fatalen Folgen.

In LANs, in denen das *Address Resolution Protocol (ARP)* eingesetzt wird, sind sehr viele wirkungsvolle Spoofing-Angriffe möglich. Es reicht dann meist schon, die Kontrolle über einen der Rechner im LAN zu bekommen, um das gesamte Netz zu kompromittieren. Dabei kann es zum sogenannten *Man-in-the-middle-Angriff* kommen, bei dem der Angreifer vollständige Kontrolle über den Datenverkehr zwischen zwei oder mehreren Netzknoten erlangen kann. Mögliche Angriffsstrategien wären das Modifizieren der ARP-Tabellen der Opfersysteme oder das Vorspielen eines falschen DHCP-Servers. Neulich hat das Unternehmen „RedTeam Pentesting“ es geschafft beim Online-Banking Überweisungen unbemerkt auf dritte Konten umzuleiten [8].

Ein weiteres Problem wären die fehlenden Autorisierungsmechanismen bei BGP. Ein BGP-Router würde beim Eintreffen eines Datenpackes seine Routingtabelle nach der besten Route zum Ziel durchsuchen. Der Router wird den kürzesten Weg auswählen und danach die Daten schicken. Allerdings vertrauen Router darauf, dass ihnen alle anderen Router die Wahrheit sagen. Wenn ein Router mitteilt, dass er die kürzeste Route zu einem bestimmten Ziel kennt, dann entspricht es der „Wahrheit“, und die Pakete werden diesem Router übergeben. Möchte ein Angreifer den Verkehr zu einem bestimmten System umleiten, muss er nur die Routing-Einträge manipulieren, indem er behauptet eine kürzere Route zum Ziel zu kennen. Diese Strategie ist noch in der Welt der Netzwerke als *IP-Hijacking* bekannt. Im Februar 2008 wurde von der „Pakistan Telecom“ eine Anweisung zum Sperren des Zugriffs auf YouTube umgesetzt, indem auf dem Border-Gateway eine spezifischere Route zu den Servern von YouTube angekündigt wurde. Die empfangenen Pakete wurden einfach verworfen. Allerdings übernahm der Upstream-Provider von „Pakistan Telecom“ diese Information und verbreitete sie im Internet. Danach wurden alle an YouTube adressierten Daten an das Null-Device eines Rechners von „Pakistan Telecom“ ausgeliefert [9].

4. Resilienz-Techniken

Obwohl viele Internet-Nutzer das heutige Internet als anständig funktionierend einschätzen, wurde im Abschnitt 3 klar gezeigt, dass es leider nicht als Robust und Widerstandsfähig eingestuft werden kann. Um ein angemessenes QoS-Niveau garantieren zu können, wurden zahlreiche Resilienz-Techniken erfunden, die Thema dieses Abschnittes sind.

Resilienz-Techniken können in zwei Klassen aufgeteilt werden. Einerseits werden proaktive Mechanismen eingesetzt, die eine zukünftige Störung oder Ausfall vermeiden sollen. Dabei wäre folgende hypothetische Situation denkbar: jeder Netzknoten erstellt eine Routing-Tabelle, in der ein Weg zu jedem möglichen Zielknoten steht. Es werden periodisch Tabelleninformationen zu den Nachbarn des jeweiligen Knotens geschickt. Auf dieser Weise ist eine Route zu jedem Netzteilnehmer bekannt. Andererseits sind auch reaktive (on-demand) Techniken ein möglicher Ausweg aus Fehlersituationen: ein Weg zu einem Knoten wird nur gesucht, wenn auch ein Datenpaket verschickt werden soll.

4.1 Protection Switching

Protection Switching wird in der Welt der Netzwerke als die Möglichkeit bezeichnet, in Fehlersituationen auf redundante (engl. backup) Pfade umschalten zu können. Dabei zählt sie zu den proaktiven Mechanismen auf Schicht 2 des ISO-OSI Modells und kann sowohl bei Leitungsvermittelnde (engl. circuit switching) Netzen, als auch bei Paketvermittelnde (engl. packet switching) Netzen eingesetzt werden.

Bei dieser Technologie unterscheidet man hauptsächlich zwischen *Hauptpfaden*, durch denen die eigentlichen Daten geschickt werden, und *Ersatzpfaden*, die im Fall eines Linkausfalls oder Störung als Backup verwendet werden. Dabei würde man meistens eine Ring-Topologie aufbauen, da im Falle eines Fehlers schnell in der entgegengesetzten Richtung umgeschaltet werden kann.

Ein mögliches Arbeitskonzept zur Realisierung des Mechanismus wäre das Einfügen von sogenannten Domänen, die eine Ring-Struktur aufweisen. Dabei besitzt jede Domäne einen Master-Knoten und zahlreiche Transit-Knoten. Jeder Knoten ist mit zwei Ports an dem Ring angeschlossen, wobei der eine als *Primary-Port* und der andere als *Secondary-Port* gekennzeichnet ist. Im normalen Betrieb blockiert der Master-Knoten seinen Secondary-Port für alle Datenpakete mit Ausnahme der Prüfpakete um somit eine Schleifenbildung zu verhindern. Wenn eine Transit-Station einen Linkausfall entdeckt, so wird es den Master-Knoten mitgeteilt. Dabei wird der Secondary-Port des Masters geöffnet um somit der Richtungswechsel zu ermöglichen. Durch den Richtungswechsel wird ein anderer Pfad genommen, der noch als Backup bezeichnet wird.

Allerdings ist es möglich, in einem Netz von Punkt A bis Punkt B mehrere Ersatzpfade zu haben oder sogar Haupt- und Ersatzpfad gleichzeitig zu verwenden. Hierbei unterscheidet man zwischen fünf Schutzmechanismen:

1+1 Protection: Bei diesem Schutzmechanismus wird ein Hauptpfad durch einen knoten- und kantendisjunkten Ersatzpfad geschützt. Dabei werden die Daten gleichzeitig über beide Pfade geschickt um im Falle einer Störung des Hauptpfades können diese am schnellsten zum Ziel gelangen. Allerdings kann man den Ersatzpfad nur für den Backup-Traffic verwenden und man hat beispielsweise nicht die Möglichkeit Traffic niedrigerer Priorität dadurch zu transportieren. Zusätzlich ist ein Overhead von mindestens 100 % garantiert.

1:1 Protection: Bei diesem Konzept wird wieder ein Hauptpfad durch einen knoten- und kantendisjunkten Ersatzpfad geschützt. Allerdings wird der Ersatzpfad für den Backup-Traffic nur dann verwendet, wenn eine Störung vorliegt. Wie man sich leicht vor Augen führen kann, kann der Backup-Pfad für den Transport niedrig priorisierter Daten verwendet werden. Wenn aber ein Link- oder Knotenausfall vorkommt, wird das Senden des Datenstroms angehalten um somit für den hoch priorisierten Verkehr „Platz zu schaffen“.

M:N Protection: Der M:N Mechanismus wird als eine Weiterentwicklung des 1:1 betrachtet. Hierbei werden N Hauptpfade durch M Ersatzpfade geschützt mit der Nebenbedingung, dass meistens $M \leq N$ gilt. Im Falle, dass mehr als M Hauptpfade gestört werden, kann eine sichere und korrekte Datenübertragung nicht garantiert werden. Es besteht auch die Möglichkeit für N Hauptpfade nur einen Backup-Pfad zu reservieren. Dies entspricht dem 1:N Mechanismus, der noch als Spezialfall des M:N Mechanismus zu betrachten ist. Allerdings ist es nicht möglich den einen Ersatzpfad von mehreren Hauptpfaden gleichzeitig benutzen zu lassen.

Demand-Wise Shared Protection (DSP): Dieses Konzept eignet sich gut für optische Netzwerke, da man leicht die Bandbreite einer Verbindung aufteilen und für verschiedene Zwecke reservieren kann [10]. Sein Vorteil basiert auf einer effizienteren Ausnutzung der Netzstruktur, so dass die Anzahl notwendiger Ersatzwege reduziert werden kann. Allerdings hat man keine Separation zwischen Haupt- und Ersatzpfaden, sondern vielmehr eine Unterteilung der Bandbreite. Es wäre also möglich bei Bedarf einen Pfad gleichzeitig für den Haupt- und Backuptraffic zu verwenden. Im Fall eines Knoten- oder Linkausfalls wird der entsprechende Datenverkehr über den anderen Pfaden verteilt.

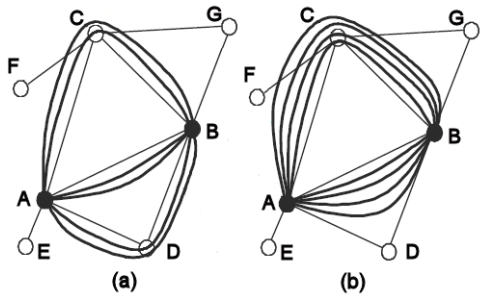

Abbildung 2: Demand-Wise Protection und 1+1 Protection

Ein Vergleich zwischen *1+1* und Demand-Wise Shared Protection ist in der Abbildung 2 zu sehen. Es müssen insgesamt vier Dateneinheiten von A bis B gleichzeitig transportiert werden. Allerdings muss sichergestellt werden, dass jeder datentransportierende Pfad gegen einzelne Link- und Knotenausfälle geschützt ist. Im Falle der DSP-Strategie (a) werden insgesamt 6 Pfade benötigt um die Anforderungen zu erfüllen. Bei der *1+1* Technik (b) werden 8 Pfade benötigt, was zu höheren Vernetzungskosten führen kann. Wie in Referenz [11] diskutiert wird, sind die beiden Techniken zusätzlich im Punkt Dienstverfügbarkeit vergleichbar.

Protection Cycles: Diese Sicherheitstechnik wird noch *P-Cycles* genannt und wird meistens in ringförmigen optischen Netzwerken verwendet, da diese oft an Linkausfälle anfällig sind. Wie in der Abbildung 3(a) zu sehen ist, wird im Voraus ein P-Cycle konfiguriert um somit eine gewisse Anzahl von Kanten und Knoten zu schützen.

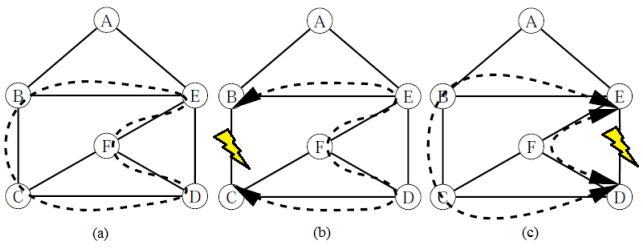

Abbildung 3: Protection Cycles

Die Links, die im Zyklus B-C-D-F-E-B zu finden sind, werden noch *On-Cycle-Links* genannt. Im Falle eines Ausfalls (Abbildung 3(b)) wird es immer noch möglich sein den Verkehr auf den Rest des Zyklus C-D-F-E-B weiterzuleiten. Zusätzlich werden Links, deren Endpunkte auf dem Zyklus liegen geschützt. Die werden noch *Straddling-Links* genannt. Ein Beispiel dazu ist in der Abbildung 3(c) zu sehen, bei dem der Link E-D ausgefallen ist. Damit der Datenaustausch zwischen den Knoten E und D funktionieren soll, werden die alternativen Routen E-F-D oder E-B-C-D verwendet.

4.2 Multiprotocol Label Switching

Im Abschnitt 3.1.2 wurde bereits die Problematik geschildert, dass das Routing heute zwar auf Knoten- und Linkausfälle reagieren kann, aber dabei schlechte Konvergenzeigenschaften aufweisen kann. Zusätzlich muss man betonen, dass die heutige Netzarchitektur komplexe Entscheidungen, Auswertungen und Funktionen den Endgeräten überlassen hat. Dabei wurde die Leistungsfähigkeit eines Zwischenknotens (Router, Switch) beschränkt, was auch zu Problemen führen kann. Da das heutige Internet als verbindungsloses Netzwerk arbeitet, muss jeder Router selber Entscheidungen treffen wie ein gewisses Paket weitergeleitet werden soll. Meistens werden Pakete in *Forwarding Equivalence Classes (FEC)* eingeteilt (z.B. mit gemeinsamer Zieladresse) um dann im nächsten Schritt alle mögliche Folgerouter auf diese FEC abzubilden. Auf dem Weg durch ein Netzwerk wird das Paket an jedem Router erneut untersucht und einer FEC zugeordnet.

Multiprotocol Label Switching (MPLS) bietet seit Ende der 1990er die Möglichkeit, Router zu entlasten um somit die verfügbaren Bandbreiten des Netzes besser auszulasten. Die Grundidee ist es, nicht mehr Pakete Hop-by-Hop durch ständigen Tabellen-Lookup weiterzuleiten, sondern diese an einem Eingangspunkt (Ingress-Router) auf einem vorbestimmten Datenpfad zu schicken und erst wieder an einem Ausgangspunkt (Egress-Router) das Standard-Forwarding zu verwenden. Meistens wird MPLS in Netzen verwendet, die auf einer IP-basierten Struktur aufweisen. Dabei besteht die Möglichkeit ein ganzes AS als eine MPLS-Domäne zu betrachten um somit Traffic schnell innerhalb Providergrenzen zu routen.

Typescherweise verlässt man sich bei der Topologiebestimmung auf ein *Interior Gateway Protocol (IGP)*, wie IS-IS oder OSPF. Danach werden *Label Switched Paths (LSPs)* aufgebaut, deren Anfang ein Ingress-Router und Ende ein Egress-Router ist. Dabei werden meistens ein Anfangs- und ein Endknoten eines AS gewählt. Jetzt kann beispielsweise eine FEC Zuordnung nur ein Mal vom Ingress-Router durchgeführt werden. Sobald ein Paket ein MPLS-Netz betritt, wird es mit einem 32 Bit MPLS-Header [17] versehen, der u.a. den Weg durch das Netz bestimmen soll (Abbildung 4).

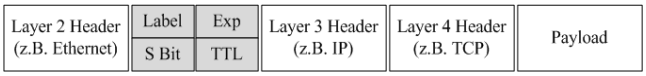

Abbildung 4: Header-Reihenfolge bei MPLS

Alle entlang eines LSP liegenden Router (Label Switch Router, LSR) treffen anhand des im Header befindbaren Labels ihre Forwarding-Entscheidung. Hier liegt auch der große Unterschied zu IP, da durch die Beförderung entlang eines LSP keine lokal bestimmte Wegewahl mehr stattfindet. Beim Forwarding wird an einem Router nun der Wert des jeweiligen Labels betrachtet um zu einer Entscheidung zu gelangen. Es ist nicht mehr nötig wie bei dem Longest Prefix Match-Forwarding den ganzen IP-Header

zu empfangen, der in der Regel mehr Informationen als gebraucht enthält. Dabei ist das Label nur eine Informationseinheit, die beispielsweise ein Paket zur jeweiligen FEC zuordnet.

MPLS bietet zusätzlich den *Fast-Reroute-Mechanismus (MPLS-FRR),* der bei Bedarf Traffic umleiten kann um somit die Resilienz des Netzwerkes zu erhöhen. Um die negativen Konsequenzen eines Link- oder Knotenausfalls zu vermeiden, werden neben den LSPs an jedem Knoten zusätzliche Backup-Pfade aufgestellt, über welche die Datenpakete umgeleitet werden sollen. Dabei wird der Traffic von dem jeweiligen Knoten (Point of Local Repair, PLR) auf den unmittelbaren Nachbarn weitergeleitet. Der Merge-Point ist frei wählbar und kann beispielsweise der Egress-Router sein (Abbildung 5).

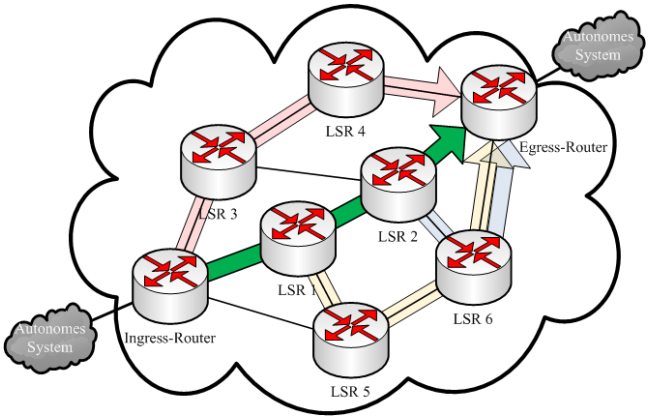

Abbildung 5: MPLS-FRR mit grünem LSP und rotem, gelbem, blauem Backup

Es existieren zwei verschiede Arten, den FRR-Mechanismus zu realisieren. In der Praxis unterscheidet man hauptsächlich zwischen den *One-to-One* und den *Many-to-One* Ansatz. Wie bei Protection Switching, kann hier die Anzahl der Backup-Pfade vom Provider bestimmt werden um ein akzeptables Niveau zwischen Backup-Aufwand und angebotene Leistung zu finden. Bei dem One-to-One Konzept verwalten die PLR-Knoten separate Backup-Pfade für jeden LSP, der durch denen passiert. Um allerdings einen drohenden Verkehrsengpass [12] zu verhindern, wird meistens die Many-to-One Methode eingesetzt, bei der ein Ersatzpfad mehrere LSPs schützen darf. Dabei wird die Anzahl der Update-Nachrichten, die zur Verwaltung der Backups nötig sind, enorm reduziert.

Ein komplett anderer Lösungsweg wäre auch die Benachrichtigung des Ingress-Routers im Falle einer Störung. Dann wäre es durchaus möglich eine neue LSP zu wählen um die Datenpakete zum Ziel zu bringen. Allerdings ist es für Provider eine ungünstige Situation, da sich im Rahmen eines Netzes mehrere tausend Knoten befinden können. Der FRR-Mechanismus würde in einer Fehlersituation maximal 50 ms [13] brauchen, da man nur auf den unmittelbaren Nachbarn den Datenverkehr umleiten muss.

4.3 Content Distribution Network

Auch wenn heutzutage die Internet-Nutzer immer schnellere Zugriffsmöglichkeiten bekommen und die Backbone-Netzen sich rapide verbessert haben, kann es beim Zugriff auf Webinhalte zu hohen Verzögerungen kommen. Die Server sind teilweise überlastet oder überhaupt nicht mehr zu erreichen und die Dienstgüte ist nur begrenzt anzubieten. Manchmal werden diese negativen Auswirkungen auch ironisch als *World Wide Wait* bezeichnet.

Um dem entgegenzuwirken wird eine zuverlässige, skalierbare und hohes QoS-Niveau garantierende Technik benötigt. Ein möglicher Losungsweg wäre der Einsatz von *Content Distribution Networks (CDNs)*, die eine resiliente und schnelle Auslieferung von Webinhalten bieten können. Allgemein enthält ein solches Netz mehrere Server, die weltweit an verschiedenen Orten verteilt sind. Deren Anzahl und Positionierung hängt allerdings alleine vom CDN-Provider ab. Der weltweit größte Anbieter für Auslieferung und Beschleunigung von Online-Anwendungen „Akamai“ beinhaltet beispielsweise 40.000 Server, die in 70 Länder zu finden sind [14]. Dabei beinhalten alle CDN-Knoten dieselben Datenkopien und sind strategisch im Netz platziert um niedrigere Latenzzeiten, hohe Robustheit und Zuverlässigkeit anzubieten. Das CDN hat dann die Möglichkeit, einem Endnutzer zu einem für ihn am besten geeigneten Server umzuleiten, anstatt auf einen Hauptserver zuzugreifen. Den großen Erfolg haben CDNs allerdings den *Content Delivery* und *Content Routing* Komponenten zu verdanken, die im Folgenden untersucht werden.

Content Delivery (Abbildung 6) beschäftigt sich mit der Kodierung, Speicherung, Bereitstellung und Auslieferung der Inhalte. Um Überlastsituationen zu vermeiden, wurde eine Struktur mit Cache-Server, die an strategischen Punkten im Netz (Points-of-Presence, POPs) zu finden sind, entwickelt. Es existieren zwei verschiedene Methoden, die die Versorgung eines Servers mit Dateninhalten ermöglichen. Man unterscheidet zwischen der *Pre-* und *Just-in-Time-Caching*. Besonders elegant ist die Pre-Caching Methode, da die Inhalte im Voraus vom Hauptserver ausgeliefert werden. Meistens wird diese Verteilung zu Niedriglastzeiten durchgeführt, um somit keinen Einfluss auf die Auslieferungszeiten zu haben. Man kann sich allerdings auch auf das Just-in-Time-Caching verlassen, dass den Dateninhalt nur dann besorgt, wenn auch eine Anfrage von einem Kunden vorliegt.

Der Einsatz des Internet Cache Protokolls (ICP) [15] wäre im Zusammenhang mit den genannten Techniken auch denkbar. Wenn beispielsweise ein Server eine Anfrage für ein Objekt erhält, dass nicht verfügbar ist, kann er mit Hilfe des Protokolls die gewünschten Daten von anderen Cache-Server empfangen, ohne den Hauptserver zu befragen.

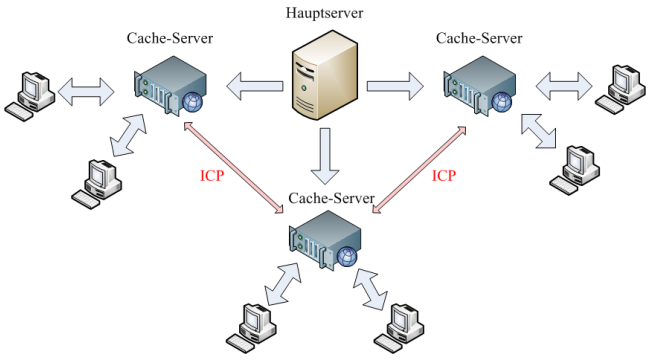

Abbildung 6: Cache-Struktur bei CDN

Content Routing (Abbildung 7) befasst sich allerdings damit, den möglichst besten Cache-Server zu finden, um Dateninhalte an den Kunden auszuliefern. Wenn ein Kunde einen Dateninhalt von der Seite eines Content-Anbieters herunterladen will, wird er normalerweise auf den DNS-Server eines CDNs weitergeleitet.

Der Client-Computer kann nun an einen für ihn am besten geeigneten Cache-Server weitergeleitet werden.

Dieses Forwarding kann nach unterschiedlichen Gesichtspunkten geschehen. Beispielsweise kann die geographische Nähe oder Antwortzeit als Hauptkriterium genommen werden. Ein anderer möglicher Lösungsweg wäre auch die Wahl des Servers, der im Subnetz des Client-Computers liegt.

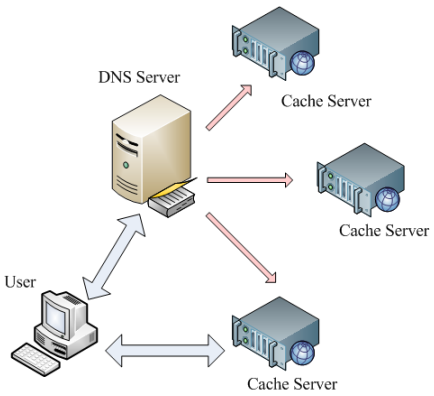

Abbildung 7: CDN-Routing

Eine weit verbreitete Methode, die ein *Load-Balancing* auf Server mit gleichem oder ähnlichem Inhalt erreicht, ist der Einsatz von sogenannten Layer 4-7 Switches. Bei einem CDN-Netzwerk heißt dies, dass hinter einem Point-of-Presence mehrere Knoten stehen, die mit identischem oder gleichem Dateninhalt ausgestattet sind. Der Switch fungiert als Proxy für alle Anfragen an diese Server und stellt eine virtuelle IP-Adresse für den gesamten POP zur Verfügung. Wenn der Switch plötzlich merkt, dass in einer Anfrage mehrere Objekte angefordert werden, darf er der Reihe nach die Server befragen. Falls ein Objekt auf mehreren Servern zu finden ist, wird der Switch den am wenigsten ausgelasteten Server auswählen und von diesem die gewünschte Dateien anfordern.

4.4 Peer-to-Peer-Netze

Peer-to-Peer-Netze (P2P) haben sich in den letzten Jahren eine rasant wachsende Popularität erfahren. Der größte Vorteil dieses Konzepts gegenüber den bekannten Client-to-Server Modell liegt in der möglichen Skalierbarkeit zur Unterstützung von Tausenden von Nutzern sowie Erzielung von Verlässlichkeit durch Dezentralität. Meistens wird P2P als ein sich selbst organisierendes System gleichberechtigter Einheiten (Peers) bezeichnet, dass ohne Nutzung zentraler Ressourcen operiert. Dabei werden Bandbreite, Speicherplatz, Rechenkapazität möglichst gleichmäßig verteilt genutzt.

Im Unterschied zu den zentralen Server-Architekturen, bei denen der Speicherort einer Datei bekannt ist, können Daten in dezentralen Systemen an zahlreichen Stellen im Netz gespeichert werden. In den letzten Jahren haben sich zur Lösung dieses Problems zwei Richtungen entwickelt, die Thema dieses Abschnittes sein werden.

Zum einen wurden die sogenannten *unstrukturierten P2P-Systeme* entwickelt, die heutzutage als die erste P2P-Generation bezeichnet werden. Meistens verlassen sich solche Systeme auf einem Hauptknoten, der die Lokation jeder Datei im Netz kennt. Damit können Peers erst nach einer Lokationsanfrage miteinander kommunizieren. Es existieren auch andere Ansätze, die das Prinzip des Flutens nutzen. Da im unstrukturierten P2P prinzipiell jeder Knoten jeden anderen Knoten als Nachbarn wählen kann, wird eine Suchanfrage weiterpropagiert um möglichst viele Peer-Knoten, die im Besitz der Datei sind zu finden. Allerdings kann es bei einer großen Anzahl von Zwischenknoten zu Überlastsituationen kommen. Zusätzlich ist keine Garantie vorhanden, dass die Suchaktion erfolgreich sein wird. Normalerweise werden Pakete mit einem Hop-Zähler ausgestattet um ein endloses Herumirren und unnötiger Ressourcenverbrauch zu vermeiden. Anschließend lässt sich sagen, dass solche Systeme auch anfällig an DoS-Attacken sind. Wie in Referenz [20] diskutiert, können bösartige Knoten andere erzwingen, einen fremden Host anzusprechen um somit Zugriff auf unerwünschte Dateien zu ermöglichen.

Zum anderen wurden *strukturierte P2P-Systeme* entwickelt, die die Probleme der Skalierbarkeit und Effizienz lösen sollen. Dabei besitzen diese eine logische Struktur und verwenden meistens den Ansatz der *Distributed Hash Tables (DHT)*, der das Auffinden einer Datei mit einer Komplexität von O(log N) ermöglicht. Da zentralisierte Tabellen oftmals langsam und anfällig gegen Angriffe sind, wird eine Aufspaltung der Hash-Tabelle durchgeführt und von verschiedenen Knoten verwaltet. Dabei wird jeder Datensatz durch einen eindeutigen Schlüssel identifiziert, der mittels einer Hash-Funktion berechnet wird. Wenn beispielsweise eine Datei gespeichert werden soll, wird mit Hilfe dieser Funktion ein solcher Wert generiert und zum verantwortlichen DHT-Knoten propagiert. Damit kann dieser Knoten in seiner Tabelle einen Zusammenhang zwischen der Datei und ihren Hash-Wert herstellen. Dieser Ansatz würde somit ein schnelles Auffinden ermöglichen, da wenn ein Peer eine bestimmte Datei haben will, kann er leicht die globale Hash-Funktion verwenden und mit Hilfe des resultierenden Wertes die verschiedenen DHT-Knoten befragen. Dabei wäre jeder DHT-Knoten in der Lage schnell eine Antwort zu generieren ob die gewünschte Datei in seinem „Bezirk“ ist.

Manche Verfahren [21] bieten auch die Möglichkeit, die verschiedenen Hash-Werte an der Netz-Topologie anzuordnen. Um das zu realisieren, wird jedem Peer eine ID zugeordnet, die beispielsweise ein Hash-Wert der IP-Adresse sein kann. Dabei existiert eine injektive Abbildung zwischen den Schlüssel-Raum und die Menge aller IDs, die die Grundlage bei einer zukünftigen Suche sein wird. Meistens werden einem Peer die Schlüssel zugeordnet, die nah an seiner ID sind. Dadurch werden Antwortzeiten auf einer Anfrage senken, da nur ein Vergleich mit der eigenen ID nötig wäre.

5. Ausblick und weitere Tendenzen

Das Internet ist mittlerweile zu einer großen Baustelle geworden. Viele behaupten, dass es sich sogar in der Krise befindet. Adressknappheit, Sicherheitsprobleme, Skalierungsprobleme beim Routing, Spam, Cyber-Kriminalität sind zur Motivation von Entwicklungsgruppen geworden komplett neue Ansätze zu entwickeln und einzusetzen. Somit wurde das Einführen von neuen Protokollen und Techniken, die die Netzwerk-Resilienz deutlich verbessern sollen, zum Thema vieler Forschungsarbeiten [16].

Ziel dieses Abschnittes ist den Leser einen kleinen Ausblick über zukünftige Methoden und Tendenzen, die sich heutzutage noch im Entwicklungsstadium befinden, zu geben.

Mittlerweile ist die Einführung des *IPv6-Protokolls* zum Lieblingsthema vieler Forscher und Entwickler geworden. Man hat schnell gemerkt, dass in Regionen, in denen sehr viele Menschen wohnen und wo gleichzeitig ein gewaltiger Wirtschaftsaufschwung stattfindet, es zu einer Knappheit im

Bereich der IPv4 Adressen kommen kann. Das IPv6 Protokoll wird diese Probleme lösen können, da es eine Erweiterung des Adressraums von 32 auf 128 Bit anbietet. Mit dieser Technologie werden ca. 340 Sextillionen zur Verfügung stehen, was auch wahllose Scans über den gesamten Adressbereich vermeiden wird. Der Nachfolger von IPv4 wird auch neue Eigenschaften wie Mobile IPv6, QoS-Unterstützung, IPsec-Verschlüsselung, Multicast und allgemeine Verbesserungen des Protokollrahmens mit sich bringen können [13].

DNSSEC ist eine weitere Tendenz, die auch als Erweiterung des Domain Namen Systems (DNS) bezeichnet wird. Sie dient dazu, die Authentizität und die Vollständigkeit der Daten von DNS-Antworten sicherzustellen. DNSSEC ist eine Art Versicherung, die einem Benutzer garantieren kann, dass nur diejenige Webseite angezeigt wird, die er aufrufen will. Dabei werden die DNS-Anfragen und -Antworten mit Hilfe kryptografischer Unterschriften gegen Verfälschungen gesichert [13]. Es ist auch möglich die Echtheit des kontaktierten Servers durch das SSL (Secure Socket Layer) Protokoll zu garantieren. Allerdings soll DNSSEC verhindern, dass man nicht schon auf einem falschen Server landet, bevor die Verbindung durch SSL gesichert wird.

Das *Stream Control Transmission Protokoll (SCTP)* kann als möglicher Nachfolger für das TCP und teilweise UDP angesehen werden [19]. Eine essenzielle Eigenschaft des Protokolls ist die Unterstützung von Multihomed-Knoten. Diese können unter verschiedene IP-Adressen erreicht werden und man hat die Möglichkeit Pakete über mehrere Wege zu schicken. Dabei erhöht man die Netzwerk-Resilienz, da auch im Falle einer Leitungsstörung die SCTP Pakete beim Empfänger ankommen können.

6. Zusammenfassung

Das Internet ist mittlerweile zur Grundlage der modernen Industriegesellschaft und der globalen Wirtschaft geworden. Gleichzeitig besitzt es eine große Bedeutung für den privaten Bereich. Es basiert jedoch größtenteils auf Techniken und Algorithmen, die in den 70er und 80er Jahren entwickelt wurden, die allerdings die heutigen Anforderungen an Sicherheit, Flexibilität und Zuverlässigkeit teilweise oder komplett nicht erfüllen können. Viele behaupten, dass es sogar zum Opfer seines Erfolgs geworden ist: Telefonate und Videos, Live-Übertragungen und Software-Downloads belasten das Kommunikationsnetz. Schlecht konzipierte Übertragungsnetze können schwerwiegende Folgen für die Dienstgüte haben. Manchmal kann die Wiederherstellung des Dienstes statt nur Sekunden mehrere Stunden in Anspruch nehmen. Längere Netzausfälle führen zu Dienststörungen, die die heutigen Geschäfte beeinflussen können. Es existieren leider zahlreiche technische Gründe, die im Abschnitt 3 erwähnt wurden, die gegen ein Widerstandsfähiges Internet sprechen.

Dennoch funktionieren die heutigen Netzwerke erstaunlich gut. Dies ist allerdings den verschiedenen Resilienz-Techniken zu verdanken, die ein angemessenes QoS-Niveau garantieren können. Manche Strategien, wie CDN und P2P-Netze, versuchen von der ossifizierten und standardmäßigen Netz-Architektur abzuweichen um somit ein gegen unerwartete Störungen widerstandsfähiges Netzwerk zu konzipieren. Dabei wurde auch das Thema Performanz und Effizienz angesprochen, die mittels geeigneter Last-Balance und Verteilung der Daten erheblich verbessert werden können. Andere Techniken, wie Protection Switching, verlassen sich auf die Redundanz, die im Falle eines Pfadausfalls ein schnelles Umleiten des Datenverkehrs ermöglicht. Man darf zusätzlich Tunneling-Mechanismen nicht unterschätzen, die zur Erhöhung der Resilienz und der Performanz führen können. Techniken, wie MPLS, bieten nicht nur ein schnelles Umleiten des Datenverkehrs, sondern auch eine Entlastung der Zwischenknoten.

Anschließend lässt sich sagen, dass Network Resilience auch im zukünftigen Internet ein aktives Forschungsthema sein wird. Techniken, wie IPv6, DNSSEC und SCTP, sind nur ein kleiner Teil der heutigen Forschungsgebiete, die allerdings eine spannende Zukunft versprechen.

7. Referenzen

[1] FP7 – 224619*, Resilience and Survivability for Future Networking*, http://www.resumenet.eu/, eingesehen am 15.03.2010.

[2] MS03-026, *Microsoft Security Bulletin*, http://www.microsoft.com/technet/security/bulletin/MS03-026.mspx, eingesehen am 15.03.2010.

[3] J. P. G. Sterbenz and D. Hutchison, *ResiliNets Wiki*, https://wiki.ittc.ku.edu/resilinets_wiki/index.php/, eingesehen am 16.03.2010.

[4] T. Schwabe, *IP-Netze mit Interdomain-BGP-Routing: Konvergenzverhalten, Dienstqualität und Dimensionierung*, Technische Universität München, September 2006, Seite 7-16.

[5] Z. Mao, R. Govindan, G. Varghese and R. Katz, *Route Flap Damping Exacerbates Internet Routing Convergence*, SIGCOMM, Pittsburgh – USA, August 2002.

[6] Renesys Blog, *Reckless Driving on the Interne*, http://www.renesys.com/blog/2009/02/the-flap-heard-around-the-worl.shtml, eingesehen am 17.03.2010.

[7] T. Sobh, K. Elleithy and A. Mahmood, *Novel Algorithms and Techniques in Telecommunications and Networking,* Chapter 17: Improving BGP Convergence Time via MRAI Timer, Springer 2010.

[8] RedTeam Pentesting GmbH, *Man-in-the-Middle-Angriffe auf das chipTAN comfort-Verfahren im Online-Banking*, Aachen, November 2009.

[9] RIPE NCC, *YouTube hijacking: A RIPE NCC RIS case study*, http://www.ripe.net/news/study-youtube-hijacking.html, eingesehen am 15.03.2010.

[10] A. Zymolka, A. Koster and R. Wessäly, *Transparent optical network design with sparse wavelength conversion*, ZIB-Report 02–34, October 2002.

[11] R. Hülsermann, M. Jäger, A. Koster, S. Orlowski, R. Wessäly and A. Zymolka, *Availability and Cost Based Evaluation of Demandwise Shared Protection*, in ITG Workshop on Photonic Networks, VDE Verlag 2006.

[12] S. Salsano, A. Botta, P. Iovanna, M. Intermite and A. Polidoro, *Traffic engineering with OSPF-TE and RSVP-TE: Flooding reduction techniques and evaluation of processing cost*, Computer Communications Volume 29, Issue 11, July 2006, Pages 2034-2045.

[13] S. Ioannidis, G. Apostolopoulos, K. Anagnostakis, N. Nikiforakis, A. Makridakis and C. Gkikas. *Resilience of communication networks: Resilience features of IPv6, DNSSec and MPLS*, Technical report, ENISA 2009.

[14] S.Triukose, Z. Al-Qudah and M. Rabinovich, *Content Delivery Networks: Protection or Threat*, EECS Department, Case Western Reserve University, 2009.

[15] D. Wessels and K. Claffy, *Internet Cache Protocol (ICP) version 2*, Internet Engineering Task Force, RFC 2186, September 1997.

[16] N. Kammenhuber, A. Fessi und G. Carle, *Resilience: Widerstandsfähigkeit des Internets gegen Störungen - Stand der Forschung und Entwicklung*, Technische Universität München, Januar 2010.

[17] E. Rosen, D. Tappan, G. Fedorkow and others, *MPLS Label Stack Encoding*, Network Working Group, RFC 3032, January 2001.

[18] M. Handley, *Why the Internet only just works*, BT Technology Journal, Vol 24 No 3, July 2006.

[19] A. Jung, *SCTP for beginners*. http://tdrwww.exp-math.uni-essen.de/inhalt/forschung/sctp_fb/, eingesehen am 22.03.2010.

[20] E. Athanasopoulos, K. Anagnostakis and E. Markatos, *Misusing Unstructured P2P Systems to Perform DoS Attacks: The Network That Never Forgets*, Foundation for Research & Technology Hellas (FORTH), 2006.

[21] U. Bischof, *Strukturierte P2P Netze*, BTU Cottbus- Seminar P2P Networking, 2005.

[22] Welt Online, *Deutsche AOL-Kunden wegen Stromausfall stundenlang offline*, Artikel 444174, 7 April 2001.

[23] V. Jacobson, *Congestion avoidance and control*, ACM SIGCOMM, Stanford, 1988.