Network Architectures
And Services
NET 2009-01-1

IITM WS08/09

Proceedings of the Seminar
Innovative Internet Technologies and Mobile Communications (IITM)

Winter Semester 2008 /2009

Munich, Germany, 13.11.2008 - 29.1.2009

Editors

Georg Carle, Corinna Schmitt

Organisation

Chair for Network Architectures and Services
Department of Computer Science, Technische Universität München

Technische Universität München





Network Architectures
and Services
NET 2009-01-1

IITM WS08/09

Proceedings zum Seminar
Innovative Internettechnologien und Mobilkommunikation (IITM) WS08/09

Wintersemester 2008/2009
Vorträge im Zeitraum 13.11.2008 – 29.1.2009

Editoren: Georg Carle, Corinna Schmitt

Seminar organisiert durch den
Lehrstuhl Netzarchitekturen und Netzdienste (I8),
Fakultät für Informatik,
Technische Universität München

Technische Universität München

IITM WS08/09
Seminar “Innovative Internet Technologies and Mobile Communications”
Wintersemester 2008/2009
Chair for Network Architectures and Services (I8)
Technische Universität München

Editors:

Georg Carle
Lehrstuhl Netzarchitekturen und Netzdienste (I8)
Technische Universität München
D-85748 Garching bei München, Germany
E-mail: carle@net.in.tum.de
Internet: http://www.net.in.tum.de/~carle/

Corinna Schmitt
Lehrstuhl Netzarchitekturen und Netzdienste (I8)
Technische Universität München
D-85748 Garching bei München, Germany
E-mail: schmitt@net.in.tum.de
Internet: http://www.net.in.tum.de/~schmitt/

Cataloging-in-Publication Data

IITM WS08/09
Proceedings of Seminar “Innovative Internet Technologies and Mobile Communications”
Wintersemester 2008 / 2009
München, Germany, 13.11.2008 – 29.01.2009
Georg Carle, Corinna Schmitt (Eds.)
ISBN: 3-937201-04-1

ISSN: 1868-2634 (print)
ISSN: 1868-2642 (electronic)
Network Architectures und Services NET 2009-01-1
Series Editor: Georg Carle, Technische Universität München, Germany

Vorwort

Wir präsentieren Ihnen hiermit die Proceedings zum Seminar "Innovative Internettechnologien und Mobilkommunikation" (IITM) aus dem Wintersemester 2008/2009 der Fakultät Informatik der Technischen Universität München.
In diesem Seminar wurden Vorträge zu allgemeinen und interessanten Themen im Forschungsbereich Internettechnologien und Mobilkommunikation vorgestellt. Die folgenden Themenbereiche wurden von den Vortragenden abgedeckt:

- Peering und Interconnection
- Network Address Translation
- DNS Security Extensions (DNSSEC)
- Kryptographische Protokolle SSL/TLS und Web Services
- Identity Management mit OpenID
- Identity Federation und Web-Services
- Knowledge Plane for Future Internets
- Mobilfunkstandard 3GPP Long Term Evolution (LTE)
- Home NodeB - Femto Cells
- Visualisierung von Verkehrsmessdaten
- Verkehrscharakterisierung durch Methoden des maschinellen Lernens
- Tampling the Torrents
- Sicheres Online-Banking
- Chronical Erkennungssysteme

Wir hoffen, dass diese Beiträge, die einen aktuellen Querschnitt durch Forschungen im Bereich Internet darstellen, informativ sind und zu einer weiteren Beschäftigung mit den Themen anregen.
Falls Sie weiteres Interesse an unseren Arbeiten haben, besuchen Sie doch bitte unsere Homepage http://www.net.in.tum.de für weitere Informationen.

München, Januar 2009

Georg Carle

Corinna Schmitt

Preface

We are very pleased to present you the interesting program of our graduate-level seminar on "Innovative Internet Technologies and Mobil Communications" – IITM 2008/2009.
In this seminar we deal with common and interesting topics in current research tasks in the field of internet technologies and mobile communication. Due to the fact that this seminar was hold in German, the contributions in the proceedings are also in German. The following topics are covered by this seminar:

- Peering and Interconnection
- Network Address Translation
- DNS Security Extensions (DNSSEC)
- Cryptografic protocols SSL/TLS and Web Services
- Identity Management with OpenID
- Identity Federation and Web-Services
- Knowledge Plane for Future Internets
- Standard for Mobile Communication 3GPP Long Term Evolution (LTE)
- Home NodeB - Femto Cells
- Visualisation of Traffic Measurments
- Traffic characterisation with machine learning methods
- Tampling the Torrents
- Secure Online-Banking
- Chronical Recognition System

We hope that these contributions, which represent selected topics of Internet research, inspire for further interest into these topics.
If you are more interested in our work, please visit our homepage http://www.net.in.tum.de for more information, also including offers for thesis projects and job opportunities.

Munich, January 2009

Seminarorganisation

Lehrstuhl für Netzarchitekturen und Netzdienste (I8)

Lehrstuhlinhaber

Georg Carle,
Technische Universität München, Germany

Seminarleitung

Corinna Schmitt, *Technische Universität München*, Germany

Betreuer der Beiträge

Tobias Bandh, *Technische Universität München, Wiss. Mitarbeiter I8*
Lothar Braun, *Technische Universität München, Wiss. Mitarbeiter I8*
Benedikt Elser, *Technische Universität München, DFG Emmy Noether Research Group Member*
Ali Fessi, *Technische Universität München, Wiss. Mitarbeiter I8*
Marc Fouquet, *Technische Universität München, Wiss. Mitarbeiter I8*
Holger Kinkelin, *Technische Universität München, Wiss. Mitarbeiter I8*
Andreas Müller, *Technische Universität München, Wiss. Mitarbeiter I8*
Gerhard Münz, *Technische Universität München, Wiss. Mitarbeiter I8*
Heiko Niedermayer, *Technische Universität München, Wiss. Mitarbeiter I8*
Marc-Oliver Pahl, *Technische Universität München, Wiss. Mitarbeiter I8*
Corinna Schmitt, *Technische Universität München, Wiss. Mitarbeiterin I8*

Kontakt

{carle,schmitt,bandh,braun,elser,fessi,fouquet,kinkelin,mueller,muenz,heiko,pahl}
@net.in.tum.de

Seminar-Homepage

http://www.net.in.tum.de/de/lehre/ws0809/seminare/

Inhaltsverzeichnis

Themenbereich 1: Internet-Mechanismen

Themenbereich 2: Internet-Protokolle

Themenbereich 3: Web-Services

Themenbereich 4: Knowledge Plane für das Future Internet

Themenbereich 5: Mobilkommunikation

Themenbereich 6: Verkehrsanalyse

Themenbereich 7: Anwendungen

Peering und Interconnection

Slawomir Chodnicki
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
Slawomir.Chodnicki@web.de

KURZFASSUNG

In dieser Ausarbeitung wird Struktur und Funktionsweise des Kerns des Internets behandelt. Insbesondere werden Autonome Systeme erklärt und deren Beziehung zu Internet Service Providern erläutert. Es werden unterschiedliche Geschäftsmodelle von Internet Service Providern, sowie übliche Zusammenschaltungsstrategien aufgezeigt und die daraus resultierende Praxis beim Weiterleiten von Datenpaketen im Internet beleuchtet.

SCHLÜSSELWORTE

Interconnection, Peering, Routing, BGP, policy based routing, Autonomes System, ISP

1. EINLEITUNG

Das Internet besteht aus einer Reihe von IP-Netzen. Damit jeder Teilnehmer im Internet erreichbar ist, müssen die IP-Netze miteinander verschaltet werden. Man spricht dabei von Interconnection Routing. Da die IP-Netze i.d.R. von privat geführten Organisationen, meist Internet Service Providern verwaltet werden, entwickelt sich eine markt- und interessengesteuerte Auswahl von bevorzugten Partnern bei der Netzzusammenschaltung. Die dadurch erzeugten Effekte werden nun näher erörtert.

2. AUTONOME SYSTEME (AS)

Zunächst betrachten wir eine Menge von IP-Netzen, die an eine Organisation zur Verwaltung übergeben wird. Eine solche Menge wird Autonomes System (AS) genannt. Jedes AS ist durch eine global eindeutige Nummer identifiziert. Die Menge aller AS bildet das Internet. Im Normalfall handelt es sich bei den Organisationen, welche ein oder mehrere AS verwalten, um Internet Service Provider (im folgenden kurz „ISPs“). Das Geschäftsmodell von ISPs besteht darin, Internetzugänge zur Verfügung zu stellen bzw. den im Internet aufkommenden Datenverkehr zu transportieren. Es kommt auch vor, dass ein AS-Betreiber gar kein ISP im herkömmlichen Sinne ist. Es kann für einen reinen Online-Content-Anbieter ebenfalls sinnvoll sein, ein eigenes AS zu unterhalten. Betreiber von Video-On-Demand Diensten, Internet- Suchmaschinen, kurz jede Organisation für die es sinnvoll sein kann, einen eigenen Internet-Serverpark zu unterhalten, kommt als AS-Betreiber in Frage. Im Folgenden wird jede dieser potentiellen Organisationen mit dem Begriff ISP synonym verwendet.

Da diese Ausarbeitung hauptsächlich wirtschaftliche und marktstrategische Faktoren beleuchtet, wird im Folgenden ebenfalls nicht scharf zwischen ISP und AS unterschieden. Wenn also von Routing zwischen ISPs die Rede ist, ist das Routing zwischen den entsprechenden AS gemeint.

2.1 Vergabe von AS

AS sind an IP-Netze gebunden, und IP-Netzadressen sind aufgrund der festen Adresslänge beschränkt. Folglich handelt es sich auch bei AS um eine beschränkte Ressource: Die Anzahl Autonomer Systeme hat eine obere Schranke in der Anzahl von IP-Netzen. AS werden weltweit gebraucht, und die für die globale Vergabe von AS zuständige Organisation ist die „Internet Assigned Numbers Authority“, kurz IANA [1]. IANA vergibt die AS jedoch nicht direkt an ISPs, sondern zunächst an subsidiär angesiedelte Organisationen. Für den europäischen Raum ist in zweiter Instanz RIPE zuständig, (Réseaux IP Européens) [2].

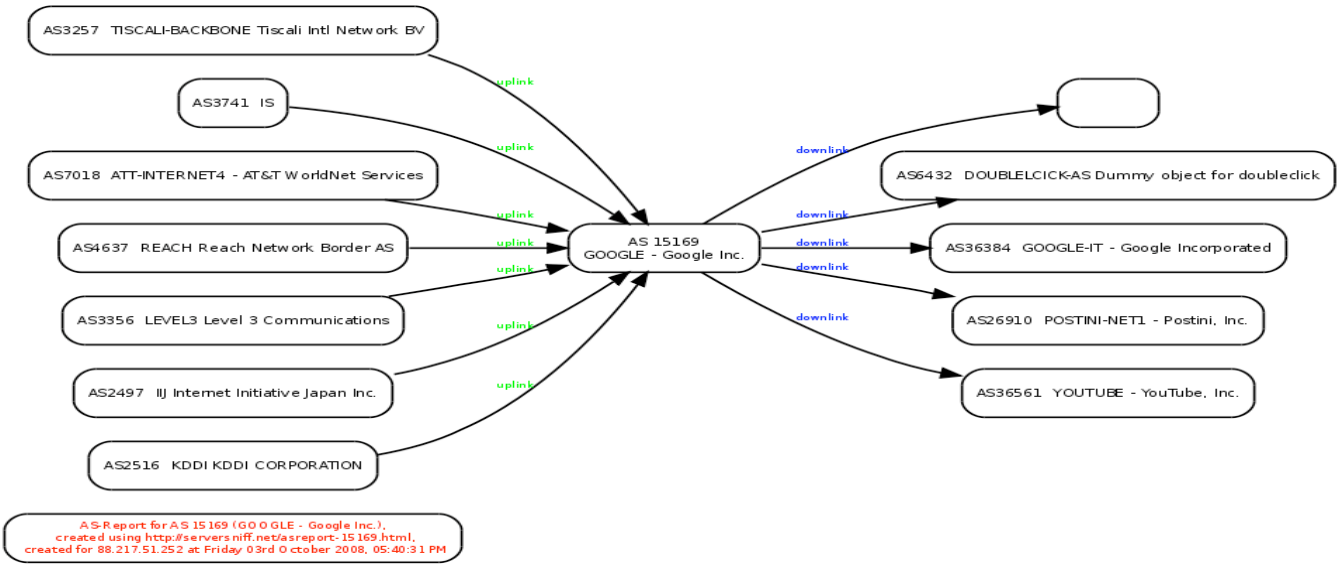


Abbildung 1. Interconnection des Google AS 15169 [6]

Prinzipiell kann jede Organisation ein AS für sich beantragen. Es müssen jedoch gewisse Voraussetzungen erfüllt werden, bevor eine AS-Nummer zugeteilt wird. So muss z.B. gewährleistet sein, dass das AS an mindestens zwei weitere AS angebunden ist, um dem Bedürfnis nach Ausfallsicherheit ein Stückweit Rechnung zu tragen.

2.2 Interconnection

AS sind miteinander verbunden, tauschen ständig Daten an den jeweiligen Netzgrenzen aus und bilden so das Internet. Die Zusammenschaltung der AS nennt man Interconnection, oder kurz IC. Prinzipiell steht es allen AS-Betreibern frei, mit welchen anderen AS sie eine Direktverbindung unterhalten wollen. Viele Verbindungen zu anderen AS bedeuten eine zuverlässigere Anbindung, aber auch höhere Betriebskosten.

Beispielhaft wollen wir in Abbildung 1 die Interconnection des AS mit der Nummer 15169 betrachten, das von der Firma Google. Inc. betrieben wird. AS 15169 ist an 12 weitere AS angebunden. Einige der Partner-AS gehören zu großen ISPs, wie etwa AS 3356 zu „Level 3“, andere gehören zu Subunternehmen der Firma Google. Inc. wie etwa der zur Youtube Inc. gehörende AS 36561.

2.3 AS-Topologie

Gegenwärtig sind über 49.000 AS-Nummern vergeben [5]. Ein vollständiger Graph mit 49.000 Knoten besitzt 1.200.475.500 Kanten. Die potentielle Größe sowie die Komplexität der AS-Topologie sind also beträchtlich.

Die Topologie Autonomer Systeme wird nicht von einer regulierenden Organisation vorgegeben und ergibt sich aus den individuellen Interessenslagen der AS-Betreiber. Sie ist demnach auch nicht statisch, sondern ändert sich in dem Maße, in dem sich die Interessen der AS-Betreiber wandeln. Folglich ist eine Strukturierung oder Systematisierung der Topologie a priori nicht möglich.

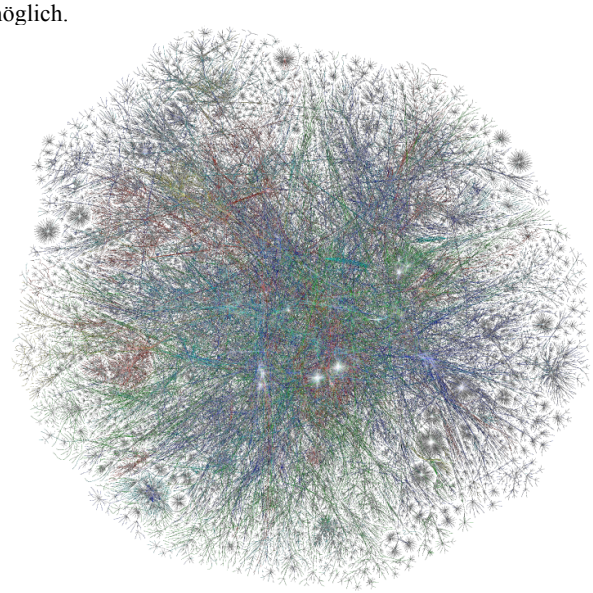


Abbildung 2. Ausschnitt der Routertopologie 01.2005 [7]

Zur etwas besseren Sichtbarkeit auf Papier wurde der ursprünglich schwarze Hintergrund ausgeblendet.

Einige Online-Projekte versuchen die Topologie des Internets auszumessen. Eine nähere Betrachtung der Meß- und Auswertungsmethoden wird an dieser Stelle nicht unternommen. Die vorliegenden Teilergebnisse des OPTE Projekts (http://opte.org) sollen hier schlicht dazu dienen, einen visuellen Eindruck zur Topologie der AS zu vermitteln. Abbildung 2 zeigt dazu einen kleinen Ausschnitt der Vernetzungsstruktur von Internetroutern, basierend auf Daten von Januar 2005.

3. AS-BETREIBER IM TIER-MODELL

Um die Interconnection-Interessen verschiedener ISPs besser zu verstehen, ist es nützlich zunächst von einem hierarchischen Routing Modell auszugehen, und dieses anschließend zu verfeinern und zu verändern. Abbildung 3 zeigt ein solches Modell.

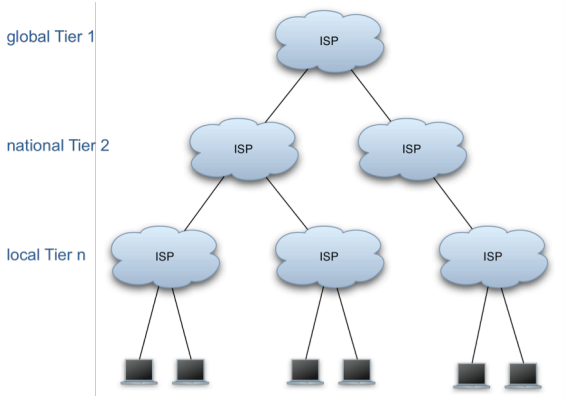


Abbildung 3. Tier Modell zur Klassifizierung von ISPs

Ganz oben in der Hierarchie, im sogenannten Tier 1 [13] befinden sich eine handvoll ISPs, die sich darauf konzentrieren, Technologie mit hohen Bandbreiten und hohen Geschwindigkeiten für den weltweiten Datenverkehr zur Verfügung zu stellen. Sie werden entsprechend auch häufig als Backbone-Betreiber bezeichnet. Als Beispiel für einen ISP des Tier 1 mag die Firma „Level 3“ dienen (http://www.level3.com).

Im Tier 2 sind ISPs angesiedelt deren Augenmerk auf dem nationalen oder auch überregionalen Datenverkehr liegt. Tier 2 ISPs gehen eine Kundenbeziehung zu Tier 1 ISPs ein, um den internationalen und ggf. interkontinentalen Datenverkehr abzuwickeln. Als Beispiel mag hier die Firma COLT Telecom dienen (http://www.colt.net/DE-de/index.htm).

Im Tier 3 sind lokale ISPs zusammengefasst die keine größere Internet Infrastruktur unterhalten und eher den Charakter eines Resellers von Internetdiensten haben. Als Beispiel mag hier die Firma KielNET dienen (http://kielnet.de/).

Da eine Resellerkette prinzipiell beliebig lang sein kann, kann man allgemein auch von n Tiern sprechen. Gängig ist jedoch eine Unterteilung in die drei genannten Tier.

Die Unterscheidungsgrenzen sind dabei subjektiv. Es kann im Einzelfall durchaus strittig sein, ob ein bestimmter ISP eher Tier 1 oder 2 bzw. Tier 2 oder 3 zuzurechnen ist. Die betroffenen ISPs tendieren natürlich oft dazu sich in dem jeweils höheren Tier zu sehen. Das Tier Modell ist also tatsächlich nur ein Modell und hilft dabei ISPs zu klassifizieren, um spezielle Untersuchungen zu vereinfachen. Exakte Definitionen aus denen eine Zuordnung zu einem Tier hervorgeht sind nicht gebräuchlich.

3.1 ISP Interconnection

Ausgehend vom Tier Modell wird klar, dass die Interessen zur Interconnection je nach Tier unterschiedlich sein müssen. Die Betrachtung mag diesmal von unten nach oben verlaufen.

Für einen ISP im Tier 3 mag es durchaus reichen, die minimal vorgeschriebenen zwei Verbindungen zu ISPs im Tier 2 zu unterhalten. KielNET ist, wie Abbildung 4 deutlich macht, wieder als Beispiel geeignet.

Für einen ISP im Tier 2 ist es hingegen nicht mehr leicht zu entscheiden, wie eine sinnvolle Interconnection Strategie aussieht.

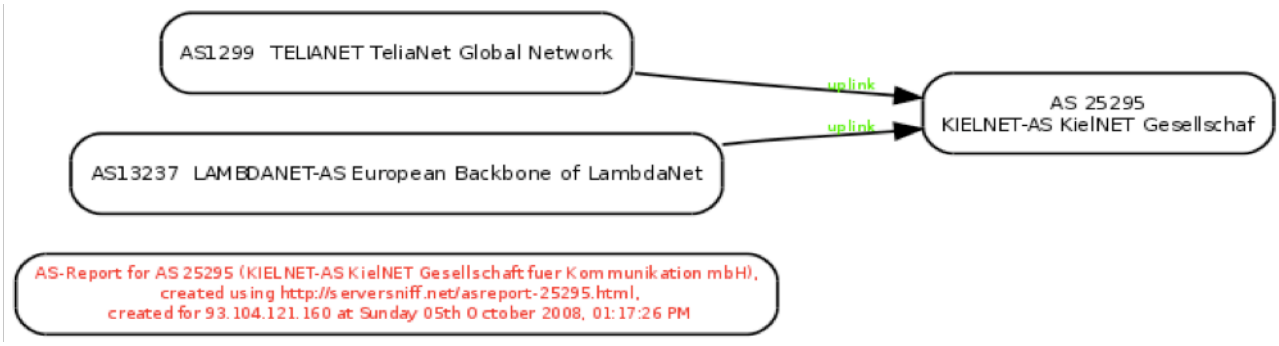


Abbildung 4. Interconnection des KielNET AS 25295 [8]

Natürlich sind direkte Verbindungen zu Tier 1 wünschenswert. Aber ob und inwieweit Verbindungen zu anderen ISPs im Tier 2 oder vielleicht Tier 3 sinnvoll sind, hängt von einer Menge von wirtschaftlichen und strategischen Faktoren ab, die von den Unternehmen, je nach strategischer Ausrichtung, durchaus verschieden bewertet werden.

Im Tier 1 ist das Kerngeschäft der Datenfernverkehr. Ein ISP im Tier 1 pflegt Verbindungen in möglichst vielen Teilen der Welt. Besonders interessant sind dabei Partnerschaften im Tier 1, um eine höhere Ausfallsicherheit zu erreichen, aber auch eine Vielzahl von vorteilhaften Anschlüssen zum Tier 2.

Bei den vorangegangenen Überlegungen wirft sich die Frage auf, wie der Netzzugang von den Vertragspartnern jeweils vergütet wird. Im Einzelfall kann die Bewertung der Nutzensituation sehr eindeutig oder aber eher strittig sein. Bei Vertragspartnern, die eindeutig unterschiedlichen Tiern angehören, ist die Bewertung im Allgemeinen einfach. Im Falle der Anbindung von KielNET dürfte klar sein, dass KielNET den höheren Nutzen aus den Verbindungen zu seinen Partnern im Tier 2 zieht als umgekehrt. Die Vertragsbeziehung ist also eindeutig eine Lieferanten- zu Kundenbeziehung bei der KielNET die Rolle des (sicher auch zahlenden) Kunden einnimmt.

Schwieriger wird es, will man die Bewertung einer Verbindung zwischen Partnern aus demselben Tier vornehmen. Wird der gegenseitige Nutzen einer gegenseitigen Verbindung als in etwa gleich angenommen, resultiert dies häufig in einer besonderen Vertragsvereinbarung, bei der jeweils keine Zugangs- oder Nutzungsgebühren erhoben werden. Die Vereinbarung gleicht also einem Tauschgeschäft zum Vorteil aller. Man spricht bei einer solchen Vertragsvereinbarung von „Peering“, da sich die beiden Vertragspartner als „Gleiche“ also als „Peers“ begegnen.

3.2 ISP Peering

Kommt es zu einer Peering-Vereinbarung zwischen ISPs, so ist diese genau so lange stabil, wie der gegenseitige Nutzen der Verbindung als gleich angenommen wird. Gerät die Nutzensituation in eine Schieflage, oder entsteht zumindest bei einem der Vertragspartner dieser Eindruck, wird die Inerconnection-Verbindung neu verhandelt. Im Einzelfall kann es durchaus auch dazu kommen, dass eine Verbindung einseitig unterbrochen wird, etwa dann, wenn sich ein Vertragspartner hintergangen fühlt.

Ein Hauptproblem bei Peering-Vereinbarungen liegt darin, dass der jeweils aus der Verbindung gezogene Nutzen für die potentiellen Partner nicht gut objektivierbar ist, und von einer Reihe von Faktoren abhängt, die in den einzelnen Unternehmen unterschiedlich gewichtet werden. Dabei sind rein technische Aspekte wie Bandbreite nur eine Kategorie unter vielen, die bei einer Peeringentscheidung eine Rolle spielen kann. Da die Verträge zwischen ISPs nicht offengelegt werden, ist es auch schwer allgemeine Usancen in diesem Berech zu benennen. Es gibt also keine einheitliche, sachliche Grundlage, auf der man Peering-Verträge stützen könnte. Es existiert keine Universalmetrik, die den Nutzen einer Peering-Verbindung für ein Unternehmen objektiviert. Infolgedessen werden Peering-Vereinbarungen häufig darauf überprüft, ob sie sich nicht in eine für den Akteur vorteilhaftere Vertragsbeziehung wandeln lassen.

Als Beispiel für gelegentliche Inkongruenz in der Nutzenwahrnehmung mag ein Vorfall im Oktober 2005 dienen, als die Peering-Verbindung zweier großer ISPs, „Level 3“ und „Cogent“, die jeweils dem Tier 1 zugerechnet werden können, einseitig unterbrochen wurde [9].

3.3 Stabiliserung / Regulierung des Peering

Der Zugang zum Internet ist ein wichtiger Stützpfeiler moderner Gesellschaften und deren ökonomischer Infrastruktur geworden. Entsprechend hat jeder moderne Staat ein besonderes Interesse an einer schnellen, möglichst flächendeckend verfügbaren und vor allem zuverlässigen Internetanbindung.

Das Beispiel von „Level 3“ und „Cogent“ macht jedoch deutlich, dass eine unregulierte Wirtschaft gelegentlich Situationen hervorbringt, die einen stabilen Internetzugang durchaus gefährden können. Immerhin handelte es sich bei den Kontrahenten um ISPs aus dem ersten Tier. Um solche ungünstigen Situationen zu vermeiden, aber vor allem um Monopolentstehung und systematische Benachteiligung Dritter zu verhindern, werden in der Regel Netz-Regulierungsstellen eingesetzt. Diese kann man, je nach Bereich, in Form von Selbstkontrollemechanismen der Wirtschaft oder amtlichen Aufsichtsbehörden vorfinden [10].

4. EXCHANGE-STRATEGIEN

Die technische Umsetzung der Interconnection muss den im vorangegangenen Abschnitt dargelegten, sich flexibel wandelnden Interessenslagen der ISPs Rechnung tragen. Den Datenaustausch an Netzwerkgrenzen nennt man auch schlicht „Exchange“. Im Folgenden werden vier technische Exchange-Konzepte vorgestellt, die neben technischen Unterschieden auch unterschiedliche Geschäftssituationen abbilden können.

Auf Protokollebene wird stets das Border Gateway Protocol verwendet (BGP), dessen kurze Charakterisierung den Beispielen vorangestellt ist.

4.1 Border Gateway Protocol

BGP ist ein Protokoll zur Verbreitung von Routingpfaden, dessen sich Router an AS-Grenzen bedienen, um mögliche Routingwege zu ermitteln und auszutauschen. Die technische Funktionsweise soll an dieser Stelle nicht näher beleuchtet werden. Von zentraler Bedeutung ist jedoch eine Eigenschaft des BGP, die andere Routingprotokolle nicht immer aufweisen und die dazu geführt

hat, dass BGP im Inter-AS Routing universell eingesetzt wird: Die Priorisierung der gewählten Routen wird vom Betreiber des Routers konfiguriert und nicht vom Protokoll vorgegeben. Diese Flexibilität eröffnet ein weites Spektrum an geschäftlichen Modellen und Vereinbarungen und wird gelegentlich auch „policy based routing“ genannt [3]. In den nachfolgend geschilderten Exchange-Strategien spielt es also durchaus einer Rolle, ob und in welchem Umfang „policy based routing“ umgesetzt werden kann.

4.2 Direct Connection Exchange

Das Konzept des Direct Connection Exchange sieht vor, dass jeder Interconnection-Teilnehmer mit jeweils allen seinen Partnern über eine feste Leitung verbunden ist. Fasst man die Router der an der Interconnection beteiligten ISPs als Knoten eines Graphen auf, so entsteht beim Direct Connection Exchange ein vollständiger Graph. Abbildung 5 zeigt eine Situation, bei der fünf ISPs über Direct Connection Exchange verbunden sind.

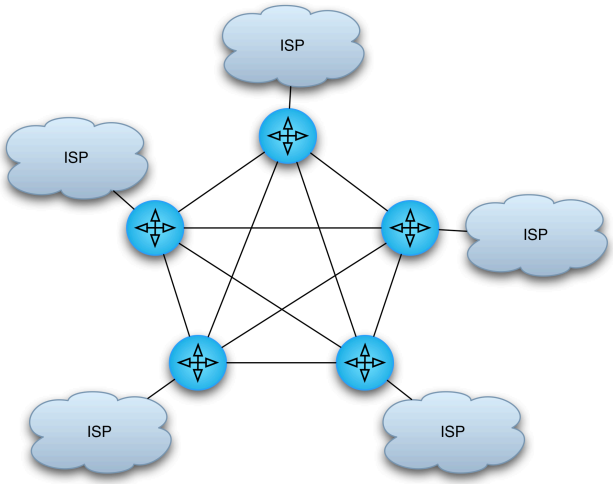


Abbildung 5. Direct Connection Exchange

Natürlich hat dieses Modell die Schwäche, dass relativ viele Leitungen unterhalten und gewartet werden müssen. Ein vollständiger Graph mit n Knoten hat n(n-1)/2 Kanten. Selbst bei einer Größe von nur 12 Knoten, eine Anzahl die laut Abbildung 1 für einen AS durchaus realistisch ist, erhält man 66 Leitungen. Folglich ist dieses Modell nicht für alle AS-Anbindungen sinnvoll. Bei strategischen und langfristigen Partnerschaften zwischen ISPs oder bei relativer örtlicher Nähe, kann es jedoch Vorteile bieten.

4.3 Exchange Router

Das Konzept des Exchange Router sieht vor, dass jeder Interconnection-Partner eine Leitung zu einem gemeinsam genutzten Router unterhält. Abbilung 6 zeigt die entstehende Konfiguration. Der Nachteil viele Leitungen unterhalten zu müssen liegt hier nicht mehr vor. Stattessen ist aber ein anderer gravierender Nachteil entstanden: Die Einhaltung von Geschäftsvereinbarungen der beteiligten ISPs kann nicht mehr garantiert werden. Einen einzelnen Router wird man in der Regel nicht so konfigurieren können, dass die Interessen aller Beteiligten in neutraler Weise gewahrt werden. Was passiert z.B. wenn zwei der ISPs gleichwertige Routen zu einem bestimmten Ziel anbieten? Darüber hinaus entsteht ein Vertrauensproblem. Welcher der Interconnection-Partner ist für die Konfiguration des Routers verantwortlich und wie kann die Konfiguration von allen eingesehen und überprüft werden?

Das Modell des Exchange Routers ist aus den genannten Gründen für viele ISPs nicht akzeptabel, und deswegen in der Praxis kaum von Bedeutung. Die Fortentwicklung der Idee eines zentralen Exchange Points, ohne die genannten Nachteile, existiert in Form der „Exchange Site“, auch „Internetknotenpunkt“ genannt.

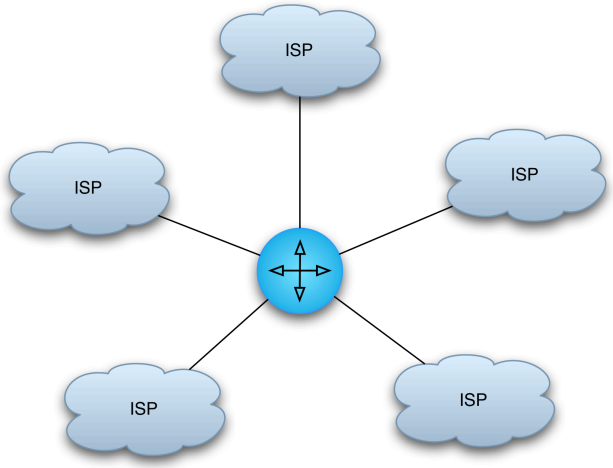


Abbildung 6. Exchange Router

4.4 Exchange Site

Das Konzept der Exchange Site sieht im Unterschied zum „Exchange Router“ nicht bloß einen Router im Zentrum, sondern ein LAN von Routern. Jeder Interconnection-Partner bringt seinen eigenen Router in das LAN der Exchange Site ein. Dieses Prinzip wird auch „colocation“ genannt. Die Anzahl der Leitungen ist reduziert, die Konfiguration der Router liegt in der Hand der jeweiligen ISPs und die Vorteile des BGP und policy based routing können voll genutzt werden.

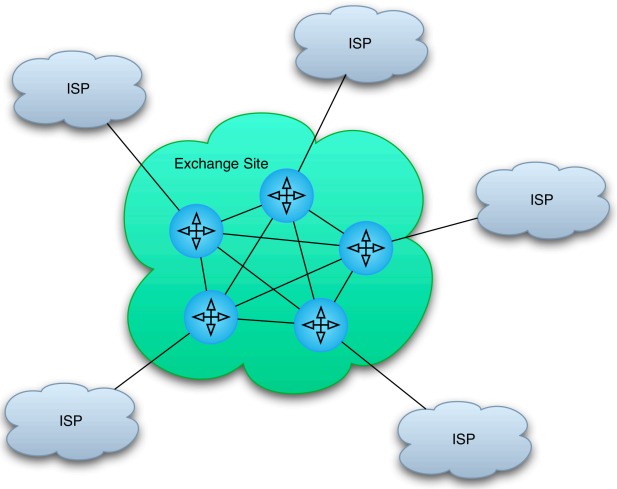


Abbildung 7. Exchange Site

Die Exchange Site wird in der Regel von einem Drittanbieter verwaltet, was einen interessensneutralen Betrieb gewährleistet.

Die Vorteile einer Exchange Site sind nicht von der Hand zu weisen. Aber auch hier entstehen neue Fragestellungen: Durch den Betrieb der Exchange Site entstehen z.B. Fremdkosten. Es muss sichergestellt werden, dass die Konfiguration aller Router im Sinne der Beteiligten ist.

In dieser mitunter etwas offeneren Umgebung gilt ein besonderes Augenmerk der korrekten Konfiguration des Routing, sowie gegebenenfalls technischen Defensivmaßnahmen gegen ungewollten Traffic.

Die dabei entstehenden routing policies werden immer länger und komplexer, je ausgefeilter die Geschäftsvereinbarungen und je mehr Partner miteinander verbunden sind, was sich in Folge negativ auf die Performance und die Zuverlässigkeit des Inter-AS-Routing auswirken kann.

Zusätzlich kommt hinzu, dass der Betreiber der Exchange Site, je nach strategischer Ausrichtung, durchaus auch eigene Interessen verfolgen kann. Die Exchange Site ist ein idealer Ort um Internet-Content zu hosten. High-Traffic-Sites, Suchmaschinenserver, DNS-Server, Usenet-Server, Video-onDemand, VoIP und IP-TV-Server würden vom Hosting an der Exchange Site profitieren. Die Rolle der Exchange Site als Knotenpunkt für das Routing kann also u.U. auch weiter gefasst werden. Abbildung 8 zeigt die erweiterte Rolle einer Exchange Site.

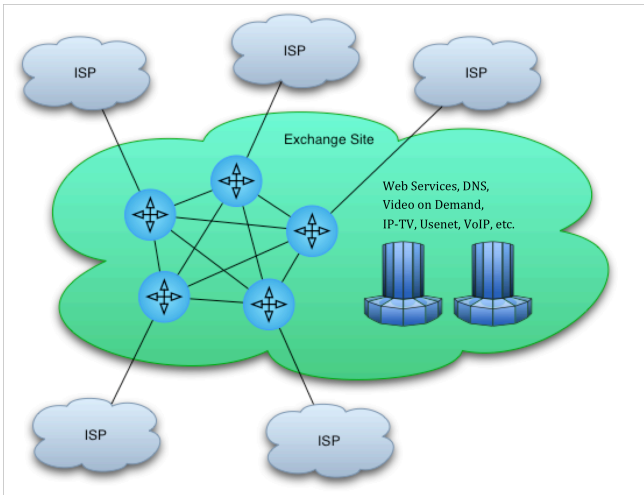


Abbildung 8. Erweiterte Exchange Site

Als Beispiel eines Betreibers von Exchange Sites mag die in Frankfurt ansässige Firma „DE-CIX“ dienen (http://www.de-cix.de/).

4.5 Distributed Exchange Network

Will man das prinzipielle Konzept einer Exchange Site beibehalten, aber die Notwendigkeit der „colocation“, also des Einbringens eines eigenen Routers in das LAN der Exchange Site eliminieren, so erhält man als Ergebnis die Idee des Distributed Exchange Network.

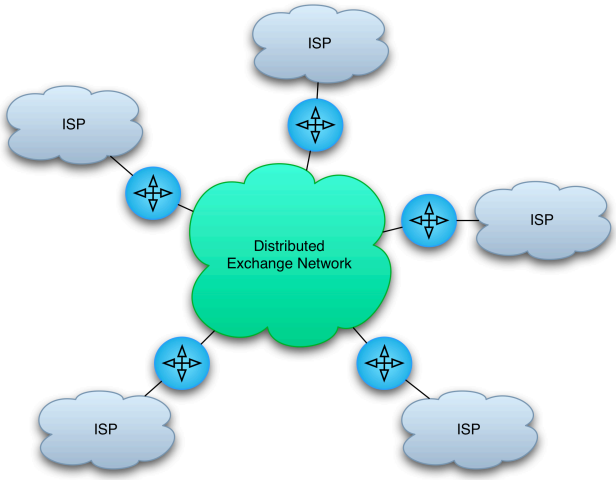


Abbildung 9. Distributed Exchange Network

In diesem Ansatz wird der Router eines jeden ISP über eine Fernleitung angebunden. Der Router verbleibt beim betreibenden ISP.

Im Distributed Exchange Ansatz gibt der Betreiber des Distributed Exchange Network die Technologie, über die eine Anbindung möglich ist, vor. Für manche ISPs kann dies ein Nachteil sein, falls die geforderte Technologievariante nicht günstig anzubinden ist. Ein wichtiger Nachteil liegt zusätzlich darin, dass Router und Switches während des normalen Switching-Vorganges auch gegenseitig Informationen austauschen, und die Wege zwischen den am Distributed Exchange Network beteiligten Routern nun wesentlich weiter sind, als im LAN einer Exchange Site. Dadurch sinkt die Switching- und Routing-Performance. Im Allgemeinen überwiegen die Nachteile im Vergleich zum klassischen Exchange Site Konzept. Das Konzept des Distributed Exchange Network ist aber oft eine gute Möglichkeit Gebiete, die fernab von Ballungszentren gelegen sind, anzubinden. Abbildung 9 veranschaulicht das Konzept des Distribted Exchange.

Als Beispiel für einen Anbieter eines Distributed Exchange Network mag „Switch And Data“ dienen, mit dem Produkt „MetroPAIX“.
(http://www.switchanddata.com/subpage.asp?navid=3&id=62)

5. DAS TRANSIT PROBLEM

Die Gründe, warum bestimmte Interconnection- bzw. Peering-Partnerschaften zustande kommen, entspringen, wie oben dargelegt, hauptsächlich unternehmerischem Kalkül. Ein grundsätzlicher Faktor, der entscheidend dazu beitragen kann Peering-Vereinbarungen zu destabilisieren, soll als Transit-Problem kurz dargelegt werden:

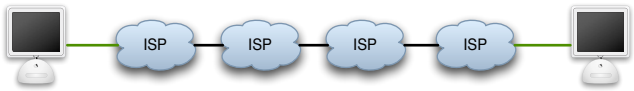


Abbildung 10. Transitverkehr im Internet

Die beiden ISPs an den Endpunkten einer IP Kommunikationsverbindung haben sicher ein ausgeprägtes Interesse daran, ein Datenpaket zuverlässig zuzustellen. Immerhin behandeln die ISPs dabei die Pakete ihrer direkten Kunden. Bei allen dazwischen liegenden und am Inter-AS-Routing beteiligten ISPs kann dieses Interesse stärker oder weniger stark vorhanden sein, je nach Interconnection-Vereinbarung mit den jeweils unmittelbaren Routing Partnern. Die ISPs an den Endpunkten der Kommunikation haben aber immer weniger Einfluss auf das Schicksal des Datenpakets, je weiter der Routingweg ist. Das Datenpaket wird so zum Spielball des unternehmerischen Kalküls, welches die Interconnection-Vereinbarungen steuert. So kann es durchaus passieren, dass Datenpakete im Transit nicht über die schnellste bzw. kürzeste Route geleitet werden, sondern über die für den jeweiligen ISP billigste.

Dies ist für alle im Internet kommunizierenden Parteien, die „Endverbraucher“, sicherlich kein wünschenswerter Zustand. Ein Hauptaspekt bei den Lösungsansätzen zum beschriebenen Transitproblem ist, dass ein mit vorhandener Technik realisierbares, einheitliches Kostenmodell, welches die Transitkosten sinnvoll aufteilt, fehlt.

6. KOSTENMODELLE

Im Folgenden werden die aktuell gängigen Kostenmodelle aufgeführt und die potentiellen Interessenskonflikte benannt.

6.1 Trafficbasierendes Kostenmodell

Ein gängiges Kostenmodell besteht darin, volumenbasiert den von fremden ISPs angenommenen Traffic zu berechnen. Je mehr fremder Traffic angenommen wird, desto teurer wird der Transit.

Varianten des Modells berücksichtigen noch andere Parameter, wie etwa die zur Verfügung gestellte Bandbreite etc. Folgende Fragestellungen werfen sich auf:

- Wie geht man mit Datenpaketen um, die nicht zugestellt werden konnten oder verworfen wurden? In aller Regel wird der Sender die Zustellung mehrfach versuchen und der Transit ISP mehrfach abrechnen.
- Welchen Anreiz hat der Transit ISP Pakete nicht absichtlich zu verwerfen, um eine Neuzustellung zu provozieren, um erneut zu kassieren?
- Volumenbasierte Abrechnung provoziert u.U. die Priorisierung technisch schlechterer Routen, die aber ökonomisch günstiger sind.

Eine (theoretische) Alternative bestünde darin, nicht den angenommenen, sondern den weitergegebenen Traffic abzurechnen. Dieses Modell hätte den Vorteil, dass ein Anreiz vorhanden wäre, Pakete zuverlässig zuzustellen. Andererseits könnten von ISP illegitime Pakete künstlich generiert, weitergegeben und abgerechnet werden. Insofern wäre dieser Ansatz auch nicht sehr überzeugend.

6.2 Sessionbasierendes Kostenmodell

Ein weiteres eher theoretisches Kostenmodell entlehnt sich aus der Welt der Telefonie. Der Verursacher einer Verbindung (Initiator einer TCP Session, bzw. Sender von UDP Paketen) würde als Kostenträger der gesamten hin und her gehenden Kommunikation identifiziert.

Dieses Modell ist kaum praxistauglich, da im Unterschied zur Technik in der Telefonie keine feste Leitung durchgeschaltet wird. So ist nicht garantiert, dass alle zu einer TCP-Session gehörigen Pakete über denselben Weg gehen wie das sessionöffnende SYN Paket. Folglich könnten nicht alle Pakete zugeordnet werden. Darüber hinaus könnte das Öffnen der Session über diverse Workaround-Techniken kaschiert werden. Entsprechend ist dieses Modell nicht von praktischer Bedeutung.

6.3 Peering-Kostenmodell

Das Peering als gegenseitiger, meist kostenneutraler Netzzugang steht in Konkurrenz zum trafficbasierenden Kostenmodell. Stabil hat sich das Peering nur zwischen gleichen Partnern gezeigt, die längerfristige strategische Partnerschaften anstreben. Grundsätzlich werden aber die Partner einer Peering-Vereinbarung versuchen, die eigene Position zu stärken und ggf. eine Zugangsgebühr auszuhandeln. Entsprechend sind viele Peering-Vereinbarungen auf Dauer eher instabil. Sie fördern darüber hinaus den „Hot-Potato“-Effekt: Falls ein ISP entsprechende Peering-Vereinbarungen hat, gibt es einen Anreiz Datenpakete möglichst schnell aus dem eigenen Netz ins Netz eines Peers zu befördern, um die eigene Netzlast zu senken. Die Datenpakete werden behandelt wie eben heiße Kartoffeln, die man schnell weiter wirft. Der Weg zur am nächsten gelegenen, als Route in Frage kommenden, Netzgrenze ist allerdings nicht unbedingt der beste. Routen können so unnötig verlängert werden.

6.4 Folgen des Transit Problems

Durch den Mangel an einem einheitlichen Kostenmodell bleibt das Transit-Problem ungelöst. Datenpakete werden häufig nicht über die besten Routen, sondern die für die jeweiligen ISP günstigsten geleitet.

Um das Transit-Problem grundsätzlich zu lösen, müsste die bestehende Technik verändert werden. Um z.B. eine Lösung wie in der Telefoniewelt üblich zu erzielen, müsste der Weg eines jeden Datenpakets vollständig bestimmt werden können. Und zwar am besten bevor das Paket diesen Weg abläuft. Dieses Problem ist mit aktueller Technik (noch) nicht im nötigen Maßstab lösbar. Es bleibt abzuwarten welche technischen Innovationen helfen können, das Problem einzudämmen.

Derweil sind die Effekte des ungelösten Problems zu beobachten. Routinginformationen und Routingtabellen werden immer länger und komplexer, um von allen möglichen Routen die ökonomisch optimale heraussuchen zu können. Neben der Tatsache, dass der ökonomisch sinnvollste Weg selten der technisch beste ist, arbeiten die Router so langsamer, und überfluten sich gegenseitig mit überkomplexen, durch Geschäftsstrategie vorgegebenen Routen.

7. ZUSAMMENFASSUNG UND AUSBLICK

IP-Netze im Internet werden zu Autonomen System (AS) zusammengefasst. AS sind nummerierte technische Einheiten, die es erlauben eine ganze Gruppe von IP-Netzen als Einheit aufzufassen und das Routing zwischen diesen Netzen mittels des BGP Protokolls zu realisieren (Interconnection). Die Menge aller AS bildet das Internet. Üblicherweise werden AS Nummern an ISPs und andere große Netzbetreiber wie etwa Google Inc. vergeben.

Technisch erfolgt die Zusammenschaltung von AS in der Regel mithilfe von Exchange Sites. Dies sind dedizierte LANs, in denen Router aller beteiligten Routingpartner zusammenkommen (Colocation). Das Betreiben eines einzigen Exchange Routers für alle Beteiligten lässt i.d.R. keine für alle Partner gleichermaßen passende und faire Konfiguration zu. Eine Exchange Site betreut daher mehrere Router und wird von einer unabhängigen Organisation betrieben, was eine faire Behandlung aller Interconnection-Partner gewährleisten soll. Aufgrund der günstigen Lage an solchen Austauschknoten, wird innerhalb des LANs von Exchange Sites gelegentlich auch Content Hosting betrieben. Man spricht in solchen Fällen von einer Extended Exchange Site.

ISPs sind weitgehend frei darüber zu entscheiden mit welchen Partner-ISPs sie ihr Netz zusammenschalten wollen. Es gibt nur minimale Vorgaben seitens der IANA wie ein AS vernetzt sein muss. Die ISPs schließen nach eigenem Ermessen Interconnection-Verträge mit ihren Partnern. Die Verträge sehen, je nach Größe des beiden ISPs, eine ein- oder beidseitig zu entrichtende Gebühr für die Netznutzung des jeweils anderen vor. Sehen sich die beiden ISPs als gleichwertige Partner, kommt oft stattdessen ein „Peering“ zustande: der jeweils gebührenfreie Zugang zum Netz des anderen. Peering-Verträge unterliegen regelmäßiger Überprüfung, ob der gegenseitige Nutzen der Vereinbarung noch als gleichwertig eingeschätzt wird. Nachverhandlungen infolge einer Überprüfung von Peeringvereinbarungen münden gelegentlich in einem Streit, der zu Zugangssperren führt und der den Internetverkehr aufgrund von blockierten Routingpfaden beeinträchtigen kann.

Da die Netzzusammenschaltung mit wirtschaftlichen Überlegungen zusammenhängt, und die bilateralen Verträge zwischen ISPs unterschiedliche und komplexe Interessenslagen erzeugen können, kann es passieren, dass Datenpakete im Internet nicht anhand der optimalen Route weitergeleitet werden. Das Routing erfolgt stattdessen anhand der für den jeweiligen ISP billigsten Route. Dieses Phänomen wird als Transit-Problem bezeichnet.

Es gibt den Ansatz das Transit-Problem anzugehen, indem neue Kostenmodelle für die Netzzusammenschaltung vorgeschlagen werden. Jedoch ist keiner der Vorschläge mit aktuell eingesetzter Technik realisierbar. Die Eigenschaften des IP-Protokolls, sowie der eingesetzten Hardware lassen eine Einführung der neuen Kostenmodelle nicht zu.

Es zeichnet sich also für die nahe Zukunft keine Lösung des Transit Problems ab. So bleibt Internetnutzern momentan nichts

anderes übrig, als sich mit der Lage abzufinden, oder im Falle von ungünstig geleiteten Paketen an ihre Provider zu appellieren. Das Transit Problem ist leider so beschaffen, dass der Provider nicht unbedingt Einfluss auf die ungünstige Abweichung von der optimalen Route hat. Insbesondere dann nicht, wenn die Routingentscheidung von einem ISP getroffen wird, mit dem keine direkte Vertragsvereinbarung existiert.

8. Literatur

[1] IANA, http://www.iana.org, 05.10.2008

[2] RIPE, http://www.ripe.net, 05.10.2008

[3] „Border Gateway Protocol“, Wikipedia Foundation, http://de.wikipedia.org/wiki/BGP, 05.10.2008

[4] „Autonomes System“, Wikipedia Foundation, http://de.wikipedia.org/wiki/Autonome_Systeme, 05.10.2008

[5] „Autonomous System (AS) Numbers“ IANA, http://www.iana.org/assignments/as-numbers/ , 05.10.2008

[6] „ServerSniff.net AS-Report Google.de“, ServerSniff.net http://serversniff.net/asreport-google.de.html, 03.10.2008

[7] „The Opte Project Maps“, The Opte Project http://www.opte.org/maps/, Bildurl: http://bitcast-a.bitgravity.com/blyon/opte/maps/static/1105841711.LGL.2D.4000x4000.png, 04.10.2008

[8] „ServerSniff.net AS-Report KielNET.de“, ServerSniff.net, http://serversniff.net/asreport-kielnet.de.html, 05.10.2008

[9] „Internet-Backbone: Betreiber zerstritten, Kunden in Gefahr“, Silicon.de, http://www.silicon.de/hardware/netzwerk-storage/0,39039015,39177119,00/internet_backbone+betreiber+zerstritten_+kunden+in+gefahr.htm, 05.10.2008

[10] „EU diskutiert Zukunft der Netz Regulierung“ , heise.de, http://www.heise.de/newsticker/EU-diskutiert-Zukunft-der-Netz-Regulierung--/meldung/40898, 05.10.2008

[11] Geoff Huston, „Interconnection, Peering and Settlements-Part I“, The Internet Protocol Journal - Volume 2, No. 1 http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_2-1/peering_and_settlements.html, 04.10.2008

[12] Geoff Huston, „Interconnection, Peering and Settlements-Part II“, The Internet Protocol Journal - Volume 2, No. 2 http://www.cisco.com/web/about/ac123/ac147/ac174/ac200/about_cisco_ipj_archive_article09186a00800c8900.html, 04.10.2008

[13] „tier 1, tier 2, tier 3 ISP (Internet service provider)“, Smart Computing®Encyclopedia, http://www.smartcomputing.com/editorial/dictionary/detail.asp?guid=&searchtype=&DicID=19260&RefType=Encyclopedia, 05.10.2008

Network Address Translation

Seminar Innovative Internettechnologien und Mobilkommunikation WS2008

Florian Kaiser
Technische Universität München
Informatik VIII
Netzarchitekturen und Netzdienste
Email: kaiserf@in.tum.de

***Kurzfassung*— Aufgrund der beschränkten Anzahl an IPv4-Adressen hat es sich eingebürgert, den Adressraum in öffentliche, global geroutete, und private, nur lokal gültige, Adressen zu segmentieren und zwischen diesen zu übersetzen – Network Address Translation (NAT). Eine öffentliche Adresse wird unter mehreren Rechnern, die ihrerseits nur über private Adressen verfügen, aufgeteilt. Diese Vorgehensweise löst das Problem der Adressknappheit, bringt jedoch eine Reihe an Problemen mit sich – jegliche Kommunikation muss von hinter der NAT aus initiiert werden, echte P2P-Kommunikation ist dadurch nicht mehr möglich. In diesem Paper wird die Funktionsweise von NAT dargelegt und damit verbundene Probleme und Lösungsansätze aufgezeigt.**

***Schlüsselworte*— NAT, Network Address Translation, NAT Traversal, Networking, Internet Architecture, Peer to Peer computing**

I. Einleitung

> *What, exactly, is the Internet? Basically it is a global network exchanging digitized data in such a way that any computer, anywhere, that is equipped with a device called a "modem" can make a noise like a duck choking on a kazoo.*
>
> - Dave Barry

Analoge Modems sind heute in der Minderheit – die Idee des Internets als einem globalen Netzwerk, in dem jeder Rechner von jedem Punkt im Netz aus erreichbar ist, hat sich jedoch nicht verändert.

Zur Entstehungszeit des Internets war der heutige Boom der Computer nicht abzusehen, weshalb man im Internet Protokoll Version 4 (IPv4 oder auch nur IP, [**?**]) – dem Rückgrat unseres heutigen Internets – damals einen Adressraum von 32 Bit vorsah. Schienen die damit möglichen 4 Milliarden Adressen noch unglaublich viel, so sind wir heute an einem Punkt, an dem diese Anzahl nicht mehr ausreicht. Diese Tatsache hat IP-Adressen zu einem kostbaren Gut gemacht. Zwar existiert mit IPv6 [**?**] bereits seit geraumer Zeit ein Protokoll mit einem drastisch erweitertem Adressraum (128 Bit), allerdings ist die Umstellung eines derart riesigen, heterogenen Netzwerks wie des Internets eine äußerst langwierige Sache.

Unter anderem deshalb wurde das Internet in zwei Klassen unterteilt [**?**]: öffentliche Adressen, die entsprechend dem Internet-Gedanken global geroutet werden, und private Adressen, die nur für die Adressierung in einem autonomen IP-Netzwerk gedacht sind.

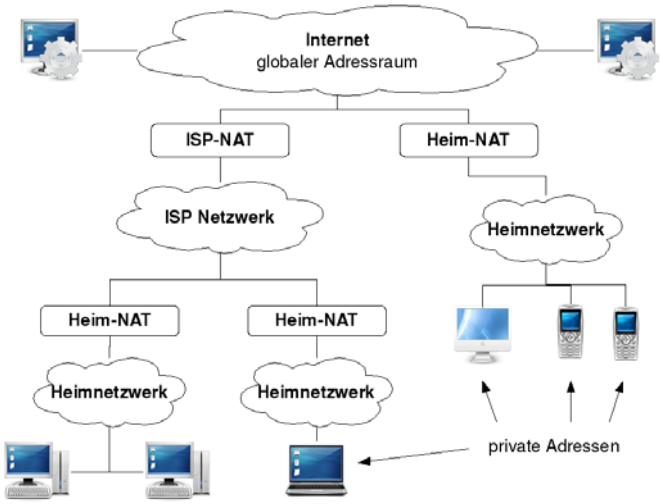


Abbildung 1. Netzwerk-Topologie mit NAT

Da diese privaten Netze unabhängig von einander sind, können Adressen mehrfach vergeben werden, was das Problem der Adressknappheit löst. Allerdings können Rechner in diesem privaten Netz per Definition erst einmal nicht mit Rechnern im Internet kommunizieren.

Um dies zu umgehen, hat es sich an vielerlei Stelle eingebürgert, eine öffentliche Adresse auf mehrere Rechner im lokalen Netz aufzuteilen. Damit sind zumindest ausgehende Verbindungen ins Internet wieder möglich (n:1-Multiplex). Diesen Vorgang bezeichnet man als Network Address Translation (zu deutsch etwa Netzwerkadressen-Übersetzung), kurz NAT.

Ein weiterer, nicht ganz so prominenter, Grund für den Einsatz von NAT kann das Bestreben sein, den Aufbau des internen Netzwerks zu verbergen (Topology Hiding). Durch NAT ist das gegeben: nach außen hin ist immer nur die Adresse des NAT-Gateways zu sehen; die Netzwerkadressen interner Hosts und damit die Netzwerk-Topologie bleiben verborgen.

Mit NAT gehen jedoch prinzipbedingte Probleme einher – Peer-to-Peer-Kommunikation ist nicht mehr möglich. Immer mehr Anwendungen setzen jedoch auf P2P-Mechanismen, z.B. VoIP (Voice over IP, "Telefonieren über das Internet"), Instant-Messenger (Kurznachrichten), Online-Spiele, Dateiübertragung und klassische Filesharing-Börsen.

Dieses Paper beschäftigt sich zunächst mit der grundlegenden Funktionsweise von NAT (II) und den damit verbundenen Problemen (III). Anschließend werden exemplarisch einige der wichtigsten Lösungsansätze (IV) zusammen mit Zahlen zu ihrer Unterstützung in derzeit eingesetzten NATs (IV-I) beschrieben. Zum Abschluss folgt eine kurze Zusammenfassung sowie ein Ausblick auf die Zukunft von NAT und NAT Traversal (V), gerade auch im Hinblick auf IPv6.

II. Funktionsweise

A. Idee

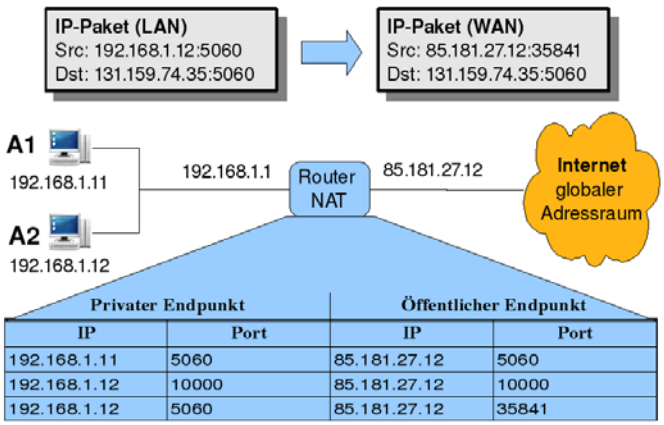
Die NAT zugrunde liegende Idee ist einfach: Jeder Host wird über eine IP-Adresse identifiziert. Damit auf einem Rechner jedoch mehrere Anwendungen gleichzeitig auf das Netzwerk zugreifen können, verwenden die beiden wichtigsten Transport-Protokolle, TCP und UDP, eine weitere Unterteilung in 65536 Ports pro Host – in der Praxis wird diese Anzahl jedoch nicht voll ausgeschöpft werden. Das legt die Idee Nahe, nur einen Host im lokalen Netzwerk über eine öffentliche IP-Adresse mit dem Internet zu verbinden und sämtliche Anfragen von Rechnern in diesem Netz über den Internet-Host zu leiten. In der Praxis wird diese Funktionalität oft von der gleichen Maschine übernommen, die auch das Routing in das Internet bereitstellt. Oft wird auch dieses Gerät, das die Netzwerkadressen-Übersetzung durchführt, als NAT bezeichnet.

In so gut wie jedem privaten Netzwerk, in dem mehr als ein Rechner mit dem Internet verbunden sein soll, wird dieser Mechanismus eingesetzt. Aufgrund der – historisch bedingten – weltweiten Verteilung von IPv4-Adressen bleibt in stark wachsenden Schwellenländern [**?**] auch manchen ISPs nichts anderes übrig, als private IPs hinter einer NAT an ihre Kunden zu vergeben. Natürlich kann weiterhin auch der Kunde selbst eine NAT betreiben, was zu mehreren Ebenen von NAT führt (siehe Abbildung 1: Netzwerk-Topologie mit kaskadierter NAT).

B. Realisierung

Eine Sitzung ist definiert durch die Endpunkte der beiden Kommunikationspartner. NAT geschieht auf Vermittlungs- und Transportschicht – für jedes Paket werden IP-Adresse und Port übersetzt.

Das NAT-Gerät befindet sich im Datenstrom (z.B. in Form eines Gateways) und übersetzt zwischen öffentlichen und privaten Endpunkten im Paket-Header: Für ein ausgehendes Paket wird der private Endpunkt nach einem gewissen Schema (siehe Abschnitt II-C) durch einen öffentlichen Endpunkt ersetzt und der Übersetzungsvorgang (Mapping) in einer Tabelle festgehalten. Man spricht hier vom Erstellen eines Bindings. Trifft nun eine an den öffentlichen Endpunkt gerichtete Antwort ein, so schlägt die NAT in dieser Tabelle das Mapping nach und überschreibt den öffentlichen Ziel-Endpunkt wieder durch den zugehörigen privaten Endpunkt, so dass das Paket im lokalen Netzwerk geroutet werden kann (grafische Darstellung des Mapping-Vorgangs: siehe Abbildung 2).



Privater Endpunkt		Öffentlicher Endpunkt	
IP	Port	IP	Port
192.168.1.11	5060	85.181.27.12	5060
192.168.1.12	10000	85.181.27.12	10000
192.168.1.12	5060	85.181.27.12	35841

Abbildung 2. Mapping-Vorgang

Bei dieser Methode, die in der Form heute in fast jeder NAT zum Einsatz kommt, handelt es sich eigentlich um Network Address Port Translation (NAPT). Es hat sich jedoch eingebürgert, auch in diesem Fall von NAT zu sprechen. In dem gesamten Artikel ist NAT immer als Synonym zu NAPT zu verstehen. Ebenfalls synonym wird oft auch der Begriff IP Masquerading verwendet.

Der Vorteil dieser Vorgehensweise ist, dass sich mit einer öffentlichen IP-Adresse mehrere öffentliche Endpunkte realisieren lassen – in der Theorie nur durch die Anzahl der verfügbaren Ports beschränkt.

Für Umgebungen, in denen ein Rechner die Übersetzung nicht bewältigen kann, existieren Ansätze für Distributed NAT [**?**]. In der Praxis wird jedoch meistens dennoch nur ein einziges NAT-Gerät mit einem öffentlichen Endpunkt eingesetzt.

C. Implementierungs-Typen

NAT kann auf verschiedene Arten implementiert werden. Es gibt keinen offiziellen Standard, so dass sich die Implementierungen von Hersteller zu Hersteller und sogar von Modell zu Modell unterscheiden.

Es existieren Versuche, das Verhalten von NATs zu standardisieren [**?**], allerdings kommt man aufgrund der vielen sich bereits im Einsatz befindenden Geräte – gerade bei Heimbenutzern – nicht umhin, sich mit den Eigenheiten verschiedener Implementierungen auseinander zu setzen.

RFC 3489 [**?**] teilt NAT-Implementierungen in vier Klassen auf (Symmetric, Port Restricted Cone, Restricted Cone, Full Cone) und definiert einen Mechanismus, um zu bestimmen, in welcher Klasse eine NAT liegt. Diese Klassen haben sich mittlerweile als Quasi-Standard zu Beschreibung des Verhaltens von NATs etabliert.

Wichtig ist es allerdings zu beachten, dass sich nicht alle Implementierungen genau in diese Klassen einteilen lassen – es gibt beispielsweise NATs, die sich in manchen Fällen wie eine Cone NAT verhalten, in anderen Fällen wie symmetrische NAT. Für eine genauere Untersuchung der Typen und Eigenschaften von NATs, die sich ebenfalls auf den in RFC 3489 definierten Klassen aufbaut, sei auf [**?**] verwiesen.

1) Cone NAT: Eine Cone NAT übersetzt den selben privaten Endpunkt immer in den selben öffentlichen Endpunkt, ungeachtet des Kommunikationspartners.

Eingehende Pakete an den öffentlichen Endpunkt werden von jedem Host (*Full Cone NAT*), nur von einem Host, an den bereits ein ausgehendes Paket verschickt wurde (*Address Restricted Cone NAT*) oder nur von einem Endpunkt (Adresse und Port), an den bereits ein Paket verschickt wird (*Port Restricted Cone NAT*) akzeptiert.

2) Symmetric NAT: Wie Port Restricted Cone NAT, zusätzlich wird jedoch für jede Sitzung ein neuer öffentlicher Endpunkt erzeugt, auch wenn der private Quell-Endpunkt identisch ist. Dies macht es unmöglich, den öffentlichen Endpunkt des Hosts hinter der NAT von einem zu einem anderen Rechner weiterzugeben. Genau das ist jedoch die Voraussetzung für Hole Punching (IV-A.2).

D. Hairpin Translation

Gerade bei einer durch den ISP eingesetzten NAT kann es vorkommen, dass sich zwei Rechner – ohne sich dessen bewusst zu sein – hinter der selben NAT befinden. Es ist im Allgemeinen für den Client nicht möglich, diese Situation zuverlässig festzustellen.

Deshalb ist es wichtig, dass die NAT auch ausgehende Pakete untersucht, erkennt, wenn diese an einen von der NAT verwalteten öffentlichen Endpunkt adressiert sind, und diese übersetzt und gleich wieder ins private Netzwerk zurück leitet (wie eine gekrümmte Haarnadel, daher der Name Hairpin Translation).

Nur ein Bruchteil der zurzeit eingesetzten NATs unterstützt jedoch Hairpin Translation (siehe IV-I).

III. Probleme mit NAT

Bedingt durch den n:1-Multiplex (n private Adressen werden auf eine öffentliche Adresse übersetzt) muss jede Sitzung von hinter der NAT aus initiiert werden – schließlich müssen in der NAT erst die Bindings erzeugt werden, bevor eingehende Pakete an die öffentliche Seite der NAT einem Host im lokalen Netz zugeordnet werden können. Aufgrund ihrer begrenzten Ressourcen kann die NAT Bindings nur endlich lange aufrecht erhalten, was auch einem eigentlich zustandslosen Protokoll wie UDP gewissermaßen einen Zustand aufzwingt: Findet in einer Sitzung eine gewisse Zeit lang keine Kommunikation statt, so wird in der NAT das Binding gelöscht – die Sitzung bricht zusammen.

IV. Lösungsansätze

Um das Problem der eingehenden Verbindungen zu umgehen, gibt es mehrere Möglichkeiten.

Wollen zwei Rechner, die sich potentiell jeweils hinter einer NAT befinden, kommunizieren, so ist eine Möglichkeit, dass sich beide Rechner zu einem dritten Server verbinden und dieser die Daten zwischen den beiden Clients weiterleitet (IV-B). Diese Variante funktioniert mit jeder Form von NAT, allerdings läuft sämtliche Kommunikation nicht direkt zwischen den Clients ab, sondern eben über den zusätzlichen Server. Dies kann die Verbindungsqualität (Geschwindigkeit, Latenz) durchaus beeinträchtigen. Desweiteren verursacht das Betreiben des Servers hohe Kosten, da dieser den gesamten Traffic durchschleusen muss.

Eine subtilere Lösung ist es, zwar einen zentralen Server zu betreiben, zu dem sich alle Clients eingangs verbinden, diesen aber nur dazu zu verwenden, den Clients jeweils den (öffentlichen) Endpunkt ihres Partners mitzuteilen und die eigentlichen Daten dann mit Hilfe von Hole Punching (IV-A.2) direkt zu übertragen (Rendezvous-Server).

Die bisher beschriebenen Lösungen setzen voraus, dass der Service Provider einen Server bereitstellt, der sich um NAT Traversal kümmert.

Es ist jedoch auch möglich, dass sich der Client selbst um NAT Traversal bemüht, indem er z.B. das Binding in der NAT manuell erzeugt (IV-F).

Der Vorteil einer serverseitigen Lösung ist, dass der Client nicht angepasst werden muss. Dafür muss im Server jede Anwendung individuell implementiert werden. Ein clientseitiger Ansatz hingegen erspart dem Service Provider viel zusätzlichen Aufwand, ist allerdings nicht immer praktikabel.

Im Folgenden eine Übersicht über die verbreitetsten NAT Traversal-Techniken.

In der Praxis ist natürlich eine Kombination der genannten Verfahren möglich und wünschenswert. Ein Beispiel für eine Anwendung, die eine ausgeklügelte Sammlung von NAT Traversal Techniken einsetzt ist das Skype-Protokoll. [**?**]

A. Rendezvous-Server

1) Connection Reversal: Befindet sich nur einer der Kommunikationspartner hinter einer NAT, so kann über einen Rendezvous-Server einfach die Umkehrung der Sitzungs-Richtung vereinbart werden:

Möchte beispielsweise Client A per VoIP Client B, der sich hinter einer NAT befindet, anrufen, so veranlasst er via Server S, an dem beide Kommunikationspartner registriert sind, einen Rückruf seitens B. Für B handelt es sich bei dem Anruf nun um eine ausgehende Sitzung, die problemlos durch die NAT möglich ist.

Zusammen mit Hole Punching ist auch eine Erweiterung des Szenarios auf den Fall, dass sich nur ein Client hinter einer symmetrischen NAT befindet, möglich.

2) Hole Punching: Ist der öffentliche Kommunikations-Endpunkt des Partners bekannt – z.B. durch einen Rendezvous-Server, so ist für viele Protokolle, darunter UDP und TCP, sogenanntes Hole Punching möglich. Das bedeutet, der Client versendet ein Paket an die Zieladresse. Dieses Paket wird möglicherweise von der NAT des Partners verworfen, passiert aber in jedem Fall zuvor die lokale NAT und führt dazu, dass ein neues Binding angelegt wird (ein “Loch” in der NAT). Der Partner führt genau den gleichen Vorgang durch und öffnet dadurch ein “Loch” in seiner NAT. Nun ist eine Kommunikation in beide Richtungen möglich (grafische Darstellung: Abbildung 3). Es muss lediglich noch darauf geachtet werden, dass periodisch Keep-Alive Pakete (Pakete, die keine sinnvollen Daten transportieren, sondern nur dem

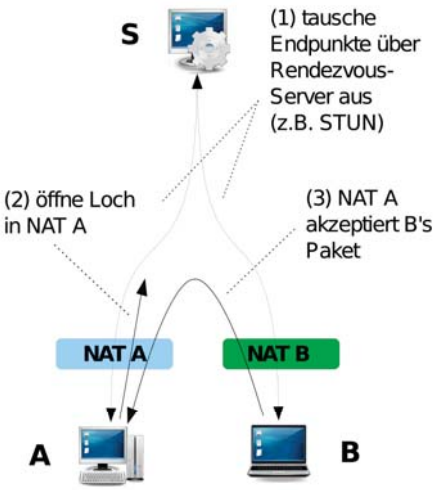


Abbildung 3. Hole Punching-Vorgang

Zweck dienen, die Verbindung aufrecht zu erhalten) verschickt werden, bevor eine der NATs die Bindings löscht.

Am weitesten verbreitet ist Hole Punching für das UDP-Protokoll, z.B. in Form von STUN (IV-A.3) für VoIP via SIP (*Session Initiation Protocol*, [**?**]). Prinzipiell lässt sich die Technik genauso auf andere Protokolle wie TCP anwenden, was allerdings in der Praxis relativ wenig eingesetzt wird und auch weniger zuverlässig funktioniert.

Hole Punching ist in vielen Fällen eine elegante und praktikable Lösung, funktioniert jedoch nicht mit allen Typen von NAT (siehe II-C, IV-I).

Für Näheres zu Hole Punching via UDP und TCP sei auf [**?**] verwiesen.

3) STUN: STUN (*Simple Traversal of UDP Networks*, [?]) ist ein von der IETF (*Internet Engineering Taskforce*) spezifiziertes Protokoll, das es einem Host ermöglicht, den Typ der NAT, hinter der er sich befindet, sowie den verwendeten öffentlichen Endpunkt zu ermitteln (NAT-Typen: siehe II-C). Das Protokoll definiert hierfür einen STUN-Server, der sich auf der anderen Seite der NAT – im Regelfall im öffentlichen Internet – befinden muss.

Im Falle von symmetrischen NATs ist der zurückgelieferte öffentliche Endpunkt jedoch unbrauchbar, da nur für die Kommunikation mit dem STUN-Server gültig.

STUN nach RFC 3489 ist also keine Komplettlösung für NAT Traversal, sondern lediglich ein Hilfsmittel. Dies wurde in der ursprünglichen Fassung von STUN jedoch nicht deutlich, so dass STUN in RFC 5389 [**?**] (Draft) neu formuliert und in "Session Traversal Utilities for NAT" umbenannt wurde.

STUN ist, wie der ursprüngliche Name bereits besagt, nur für den Aufbau von UDP-Sitzungen ausgelegt. Es gibt jedoch einen Versuch, den Ansatz auch auf TCP zu erweitern: STUNT (*Simple Traversal of UDP through NATs and TCP too*, [**?**]).

B. Relay-Server

1) SOCKS: Das SOCKS-Protokoll (Abkürzung für *SOCKetS*, [**?**]) wurde ursprünglich entwickelt, um eine Möglichkeit für Hosts hinter einer Firewall zu schaffen, mit externen Hosts zu kommunizieren, ist jedoch auch auf NATs anwendbar.

Sämtliche Daten werden von der SOCKS-fähigen Anwendung über einen Tunnel an den SOCKS-Server gesendet und von dort ins öffentliche Netz weiter geroutet.

SOCKS ist für Client/Server-Anwendungen ausgelegt, es unterstützt nur entweder eingehende oder ausgehende Verbindungen. Die aktuelle Version 5 bietet Unterstützung für die Protokolle TCP und UDP sowie Multicast und Authentifizierung.

Aufgrund der Client/Server-Beschränkung funktionieren nicht alle Protokolle mit SOCKS – und bei jenen, die es tun, würde oft auch reines NAT ausreichen.

Ein positiver Aspekt von SOCKS ist, dass sich der Einsatz eines SOCKS-Proxies durch eine Lösung wie tsocks [**?**] für die Anwendung völlig transparent gestalten lässt.

2) TURN: Einen IETF-Draft für einen Relay Server stellt das TURN-Protokoll (*Traversal using relay NAT*, [**?**]). Es erlaubt es einem Host hinter einer NAT, eingehende Verbindungen über TCP- oder UDP-Verbindungen zu erhalten. TURN ist allerdings nicht für den Betrieb eines Servers hinter einer NAT ausgelegt.

Interessant wird TURN vor allem im Zusammenhang mit ICE als letzter Ausweg, falls die Kommunikation mittels Hole Punching fehlschlägt.

C. ICE

ICE (*Interactive Connectivity Establishment*, [**?**]) kann als eine Art Meta-Protokoll zur Aushandlung von NAT Traversal Techniken angesehen werden.

Es setzt auf IETF-Standards wie STUN und TURN auf und ermöglicht eine stufenweise Eskalation der Traversal-Techniken. So kann z.B. zuerst eine direkte Verbindung zum privaten Endpunkt des Hosts versucht werden, anschließend Hole Punching und zu guter Letzt, falls auch dieses versagt, auf Relay via TURN ausgewichen werden.

Die ICE-Protokollsuite stellt ein vielversprechendes, standardisiertes Werkzeug im Umgang mit NATs jeglichen Typs dar. Der Preis dafür ist allerdings eine relativ hohe Komplexität.

Bis jetzt hat ICE den Status eines Internet-Drafts und ist erst in wenigen Anwendungen – Hardware wie Software – implementiert.

D. Application Layer Gateway

Ein Application Layer Gateway (ALG) könnte man als NAT für die Anwendungsschicht bezeichnen. Es inspiziert die Anwendungsschicht-PDU und übersetzt dort Endpunkte.

Dies setzt allerdings voraus, dass dem ALG das Protokoll bekannt ist.

In vielen NATs ist ein ALG für verbreitete Protokolle wie FTP (*File Transfer Protocol*, [**?**]) implementiert. Im Falle von FTP sucht das ALG beispielsweise nach einem PORT-Kommando. Entdeckt es ein solches, wird es den übergebenen privaten Endpunkt in einen freien öffentlichen Endpunkt übersetzen und für diesen in der NAT ein Binding erzeugen.

Versagen hingegen wird ein ALG bei neuen, noch nicht implementierten Protokollen. Weiterhin ist die Inspektion einer so hohen Schicht im OSI-Modell [**?**] mit erheblichem rechnerischen Aufwand verbunden. Für komplexe P2P-Protokolle, bei denen unter Umständen mehrere hundert Verbindungen pro Sekunde aufgebaut werden, übersteigt die Komplexität die Rechenkraft heutiger "Heim-NATs".

E. Port Prediction / Symmetric NAT Traversal

Viele NATs weisen Ports nach einem vorhersagbaren Schema zu – z.B. durch einfaches Inkrementieren des letzten benutzten Ports. Es gibt Ansätze, sich dies für die Kommunikation durch symmetrische NATs zunutze zu machen, in dem man versucht, den verwendeten öffentlichen Endpunkt – basierend auf zuvor zugewiesenen Endpunkten – zu "erraten".

Ein solches Verfahren, aufbauend auf STUN, wird in [**?**] beschrieben.

Allerdings sollte ein solch heuristisches Verfahren nur als letzter Ausweg betrachtet werden, um in einigen Fällen selbst eine symmetrische NAT umschiffen zu können. Je mehr mögliche Endpunkte vom Kommunikationspartner durchprobiert werden müssen, desto länger dauert der Verbindungsaufbau – und es gibt keinerlei Garantie, dass der richtige Endpunkt schlussendlich auch getroffen wird.

F. Port Forwarding

1) Manuelles Portforwarding: Fast jede NAT bietet heute die Möglichkeit, über ein Webinterface von Hand Bindings einzutragen. Dies wird als Portforwarding oder Static NAT (SNAT) bezeichnet. Diese statischen Bindings bleiben dauerhaft bestehen, bis sie manuell wieder entfernt werden.

Solange eine Anwendung nur bereits manuell weitergeleitete Ports verwendet, ist die NAT für sie weitgehend transparent. Allerdings ist diese Vorgehensweise in der Praxis nur für eine kleine Anzahl an Ports, die sich nicht (oft) verändern, praktikabel.

2) Automatisiertes Portforwarding: Portforwarding lässt sich natürlich auch automatisieren. Mit UPnP-IGD (*Universal Plug and Play – Internet Gateway Device*, [**?**]) und NAT-PMP (*NAT Portmapping Protocol*, [**?**]) / Bonjour existieren zwei Protokolle, die es Anwendungen ermöglichen, dynamisch Portforwardings einzurichten.

Allerdings sind diese Protokolle nur auf einem geringen Anteil der eingesetzten NATs aktiv. Weiterhin macht der Einsatz nur in einem privaten Heimnetzwerk Sinn – ein ISP wird kein Interesse daran haben, dass der Kunde seine Netzwerk-Infrastruktur beeinflussen kann.

G. Tunneling

Manche Protokolle werden von NATs nicht übersetzt. Ein Grund kann sein, dass das Protokoll zu neu und deshalb nicht in der NAT implementiert ist (z.B. SCTP [**?**], DCCP [**?**]). Manche Protokolle wie IPSec (Internet Protocol Security, [**?**]) können aber auch prinzipbedingt nicht direkt übersetzt werden, da z.B. die Payload verschlüsselt ist.

Dieses Problem kann man umgehen, in dem man die zu übermittelnden Pakete durch ein bekanntes Protokoll, z.B. UDP, tunnelt. Jedes Paket wird also mit z.B. einem UDP-Header versehen, welcher von NATs übersetzt werden kann. Allerdings muss auch die Gegenstelle dieses Verfahren unterstützen und den UDP-Header wieder entfernen. Für IPSec ist dieses Vorgehen weit verbreitet und wird von vielen NATs unterstützt.

H. Realm Specific IP

RSIP ist ein experimentelles IETF-Framework [**?**] und -Protokoll [**?**] und soll eine Alternative zu NAT darstellen. Ein RSIP Host "leiht" sich (idr. öffentliche) Endpunkte von einem RSIP-Gateway. Möchte der Host Daten über einen geliehenen Endpunkt versenden, so überträgt er diese über einen Tunnel versehen mit privatem und geliehenem Absender-Endpunkt an das Gateway. Dieses entfernt den privaten Endpunkt und routet das Paket weiter. Antworten werden auf die gleiche Weise wieder zurück zum Host geleitet.

Vorteilhaft an diesem Verfahrens ist, dass die Ende-zu-Ende-Integrität der übermittelten Pakete erhalten bleibt.

Auch ein Ansatz für IPSec via RSIP existiert [**?**].

Allerdings scheint RSIP bis jetzt nicht über den experimentellen Status hinausgekommen zu sein.

I. Einige Zahlen zu NAT Traversal

Eine Untersuchung von Ford, Srisuresh und Kegel aus dem Jahr 2005 [?] führte zu dem folgenden Ergebnis:

Von den getesteten 380 NATs unterstützen

- 82% UDP Hole Punching
- 64% TCP Hole Punching
- 24% UDP Hairpin Translation
- 13% TCP Hairpin Translation

Getestet wurden NATs von 68 verschiedenen Herstellern sowie die eingebaute NAT-Funktionalität von 8 verschiedenen Betriebssystemen (bzw. Betriebssystem-Versionen).

Aus diesen Zahlen lässt sich ablesen, dass UDP Hole Punching eine interessante Technik ist, die in den meisten Fällen funktioniert – in der Tat wird sie auch an vielen Stellen wie z.B. VoIP (STUN) eingesetzt. Dennoch existiert mit 18 % ein nicht zu vernachlässigender Anteil an Clients, bei denen Hole Punching nicht möglich ist und deshalb auf andere Lösungen wie z.B. einen Relay-Server zurückgegriffen werden muss.

Aus einem Feldtest von A. Müller [**?**] mit 400 NATs geht weiterhin hervor, dass Portforwarding via UPnP-IGD nur in 13 % der Fälle unterstützt wird. Diese doch recht geringe Zahl dürfte allerdings auch daher rühren, dass in vielen Heim-Routern UPnP-Unterstützung zwar vorhanden, aus Sicherheitsgründen aber standardmäßig deaktiviert ist.

V. Zusammenfassung und Ausblick

NAT ist ein weit verbreitetes Verfahren, um eine öffentliche IP-Adresse unter mehreren Hosts zu teilen. Notwendig gemacht wird dieses Vorgehen in erster Linie durch die Adressknappheit bei IPv4. Ein weitere Grund für den Einsatz von NAT kann das Bestreben sein, die interne Netzwerk-Topologie des eigenen Netzwerks zu verbergen. Als Resultat des Übersetzungsvorgangs muss jede Sitzung von einem Host hinter der NAT initiiert werden – P2P-Kommunikation ist nicht möglich. Es existieren verschieden Lösungsansätze, um die mit NAT verbundenen Probleme zu umschiffen, die sich stark in Aufwand und Zuverlässigkeit unterscheiden. Prinzipbedingt kann es jedoch kein Verfahren geben, das NAT in jedem Fall völlig transparent macht.

Es muss daher individuell für jede Anwendung, die mit NATs funktionieren soll, abgewogen werden, welches bzw. welche Verfahren eingesetzt werden sollen. ICE [?] stellt einen vielversprechenden Ansatz zur Koordinierung der verschiedenen Traversal-Techniken dar, ist jedoch bis jetzt nur ein Internet-Draft und in Hardware wie Software wenig verbreitet.

IPv6 könnte NAT, zumindest was den Adressraum betrifft, überflüssig machen. Dennoch ist der Einsatz von NAT z.B. zum Verbergen der internen Netzwerk-Topologie wahrscheinlich. Auch für IPv6 wurden nicht global geroutete Adressbereiche definiert [**?**]. Ein IETF-Draft für NAT mit IPv6 existiert ebenfalls bereits [**?**].

Literatur

[1] J. Postel, "Internet Protocol," RFC 791 (Standard), Sep. 1981, updated by RFC 1349. [Online]. Available: http://www.ietf.org/rfc/rfc791.txt

[2] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460 (Draft Standard), Dec. 1998, updated by RFC 5095. [Online]. Available: http://www.ietf.org/rfc/rfc2460.txt

[3] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," RFC 1918 (Best Current Practice), Feb. 1996. [Online]. Available: http://www.ietf.org/rfc/rfc1918.txt

[4] IP2Location.com, "IP2Location Internet IP Address 2008 Report." [Online]. Available: http://www.ip2location.com/ip2location-internet-ip-address-2008-report.aspx

[5] M. S. Borella, D. Grabelsky, I. Sidhu, and B. Petry, "Distributed Network Address Translation," in *Internet Draft*, 1998.

[6] F. Audet and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP," RFC 4787 (Best Current Practice), Jan. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4787.txt

[7] J. Rosenberg, J. Weinberger, C. Huitema, and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)," RFC 3489 (Proposed Standard), Mar. 2003, obsoleted by RFC 5389. [Online]. Available: http://www.ietf.org/rfc/rfc3489.txt

[8] A. Müller, G. Carle, and A. Klenk, "Behavior and classification of NAT devices and implications for NAT traversal," *Network, IEEE*, vol. 22, no. 5, pp. 14–19, September-October 2008.

[9] S. A. Baset and H. Schulzrinne, "An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol," 2004.

[10] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," RFC 3261 (Proposed Standard), Jun. 2002, updated by RFCs 3265, 3853, 4320, 4916, 5393. [Online]. Available: http://www.ietf.org/rfc/rfc3261.txt

[11] B. Ford, "Peer-to-peer communication across network address translators," in *In USENIX Annual Technical Conference*, 2005, pp. 179–192.

[12] J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session Traversal Utilities for NAT (STUN)," RFC 5389 (Proposed Standard), Oct. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5389.txt

[13] S. Guha and P. Francis, "Simple traversal of UDP through NATs and TCP too (STUNT)." [Online]. Available: http://nutss.gforge.cis.cornell.edu

[14] M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, and L. Jones, "SOCKS Protocol Version 5," RFC 1928 (Proposed Standard), Mar. 1996. [Online]. Available: http://www.ietf.org/rfc/rfc1928.txt

[15] "tsocks, a transparent SOCKS proxying library." [Online]. Available: http://tsocks.sourceforge.net

[16] J. Rosenberg, C. Huitema, and R. Mahy, "Traversal using relay NAT (TURN)," 2003.

[17] J. Rosenberg, "Interactive connectivity establishment (ICE)," 2003.

[18] J. Postel and J. Reynolds, "File Transfer Protocol," RFC 959 (Standard), Oct. 1985, updated by RFCs 2228, 2640, 2773, 3659. [Online]. Available: http://www.ietf.org/rfc/rfc959.txt

[19] N. Budhiraja, K. Marzullo, F. B. Schneider, and S. Toueg, "CCITT Recommendation X.200, Reference Model of Open Systems Interconnection for CCITT Applications," in *Revision: 2.0.0 Page 124 August 20, 1991 OSI Work Group*, 1984, pp. 81–86.

[20] Y. Takeda, "Symmetric NAT Traversal using STUN," 2003. [Online]. Available: http://www.cs.cornell.edu/projects/stunt/draft-takeda-symmetric-nat-traversal-00.txt

[21] UPnP Forum, "Internet Gateway Device (IGD) V 1.0," 2001. [Online]. Available: http://upnp.org/standardizeddcps/igd.asp

[22] S. Cheshire, M. Krochmal, and K. Sekar, "NAT Port Mapping Protocol (NAT-PMP)," Internet-Draft / Standards Track, 2008. [Online]. Available: http://files.dns-sd.org/draft-cheshire-nat-pmp.txt

[23] R. Stewart, "Stream Control Transmission Protocol," RFC 4960 (Proposed Standard), Sep. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4960.txt

[24] E. Kohler, M. Handley, and S. Floyd, "Datagram Congestion Control Protocol (DCCP)," RFC 4340 (Proposed Standard), Mar. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4340.txt

[25] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol," RFC 2401 (Proposed Standard), Nov. 1998, obsoleted by RFC 4301, updated by RFC 3168. [Online]. Available: http://www.ietf.org/rfc/rfc2401.txt

[26] M. Borella, J. Lo, D. Grabelsky, and G. Montenegro, "Realm Specific IP: Framework," RFC 3102 (Experimental), Oct. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3102.txt

[27] M. Borella, D. Grabelsky, J. Lo, and K. Taniguchi, "Realm Specific IP: Protocol Specification," RFC 3103 (Experimental), Oct. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3103.txt

[28] G. Montenegro and M. Borella, "RSIP Support for End-to-end IPsec," RFC 3104 (Experimental), Oct. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3104.txt

[29] A. Müller, "Network Address Translator - Tester." [Online]. Available: http://nattest.in.tum.de

[30] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," RFC 4291 (Draft Standard), Feb. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4291.txt

[31] M. Wassermann and F. Baker, "IPv6-to-IPv6 Network Address Translation (NAT66)," 2008. [Online]. Available: http://tools.ietf.org/html/draft-mrw-behave-nat66-01

DNS Security Extensions (DNSSEC)

Seminar Innovative Internettechnologien und Mobilkommunikation WS2008/2009

Ralf Glauberman
Lehrstuhl Netzarchitekturen und Netzdienste
Fakultät für Informatik
Technische Universität München
Email: glauberm@in.tum.de
Betreuer: Andreas Müller

***Kurzfassung*— In dieser Arbeit geht es um die DNS Sicherheitserweiterungen *DNSSEC*. Dabei werden der Grund für die Einführung von DNSSEC, die sich durch DNSSEC ergebenden Änderungen am Aufbau des DNS, neue Möglichkeiten, aber auch Probleme bei der Einführung von DNSSEC aufgezeigt. Ebenfalls wird auf die historische Entwicklung von DNSSEC und den bisherigen Stand der Verbreitung eingegangen.**

***Schlüsselworte*— Domain Name System Security Extensions DNS DNSSEC**

I. Einleitung

Alle Nutzer des Internets sind es heute gewohnt, Namen für Server zu verwenden. Die Übersetzung der Namen in für Computer nutzbare IP-Adressen passiert dabei in der Regel automatisch und ist für die meisten so selbstverständlich, dass sich kaum noch jemand Gedanken macht, wie dies eigentlich funktioniert. Seit langer Zeit dient das *Domain Name System (DNS)* der Auflösung von Namen in IP-Adressen und umgekehrt. Die Zeiten, in denen Host-Dateien für diese Auflösung manuell gepflegt werden mussten, sind längst vergessen. Obwohl eine funktionierende Namensauflösung für die Funktion des gesamten Internets unabdingbar ist, macht sich kaum jemand Gedanken, wie diese eigentlich abläuft. Um so bedrohlicher wirken die in letzter Zeit zunehmenden Berichte über Verwundbarkeiten im DNS-System (vgl. [**?**]). Zwar sind Angriffe auf das DNS nichts Neues (siehe [**?**]), jedoch scheinen die Bedrohungen durch Manipulationen heute, in einer Zeit da Phishing-Angriffe an der Tagesordnung sind und eine funktionierende Namensauflösung für viele Personen und auch große Firmen unabdingbar ist, wesentlich gefährlicher als noch vor einigen Jahren zu sein. Den meisten ist dabei nicht bekannt, dass seit längerer Zeit an einer Lösung gearbeitet wird, die Manipulationen am DNS effektiv ausschließen soll: die *DNS Sicherheitserweiterungen DNSSEC*. Dabei ist es durchaus sinnvoll, sich mit diesem neuen Standard einmal genauer zu beschäftigen und aufzuzeigen, welche Probleme dadurch genau behoben werden können.

II. Aufbau des DNS

Die Informationen des DNS sind hierarchisch in einer verteilten, dezentralen Datenbank gespeichert. Als einer der ältesten Dienste des Internets - die Anfänge des DNS gehen bis 1983 zurück - spielte Sicherheit bei der Konzeption keine Rolle. Alle Informationen sind öffentlich und weder Anfragen noch Antworten sind vor Manipulationen geschützt. Logisch ist das DNS in Zonen aufgeteilt, welche als Container für andere Zonen und Einträge, so genannte *Resource Records (RRs)* dienen. Jede Zone hat einen eindeutigen Besitzer, untergeordnete Zonen können einen anderen Besitzer haben. Resource Records (RRs) haben einen Namen, einen Typ, einen Inhalt (Wert) und Attribute. Verschiedene Typen sind bekannt, z.B. A für IPv4-Adressen, AAAA für IPv6-Adressen, CNAME für Alias-Namen oder PTR für die Rückwärtsauflösung von IP-Adressen in Namen. Das Format des Wertes ist abhängig vom Typ.

III. Sicherheitsprobleme beim bisherigen DNS

Zu der Zeit, als das DNS entwickelt wurde, spielte Sicherheit bei Computersystemen noch keine große Rolle. Daher wurde DNS nicht als sicheres System konzipiert, was eine Reihe von Angriffsmöglichkeiten eröffnet. Vorweg sei gesagt, dass DNSSEC nicht alle diese Angriffe verhindern kann.

A. Angriffe auf das DNS

Eine der wichtigsten Angriffsmöglichkeiten ist die gezielte Manipulation des Inhalts von Anfragen oder Antworten. Dadurch ist es möglich, für einen angefragten RR einen anderen Inhalt zurückzuliefern, die Existenz eines RR zu verleugnen oder die Existenz eines in Wirklichkeit nicht existierenden RRs vorzuspiegeln. Des weiteren besteht eine Gefahr durch *Denial of Service (DoS)* Angriffe auf DNS-Server. Auch können Informationen über den Inhalt einer Zone unbeabsichtigt offengelegt werden, also z.B. eine Auflistung aller RRs und Unterzonen einer Zone ermöglicht werden. Dies ist nach der ursprünglichen Ansicht, nach der alle Informationen im DNS grundsätzlich öffentlich sind, keine Bedrohung, allerdings aus verschiedenen Gründen, die im Kapitel **??** erläutert werden, heute oftmals unerwünscht.

B. Angriffe mit Hilfe des DNS

Es gibt auch Angriffe, bei denen das DNS nicht Ziel des Angriffs ist, sondern lediglich als Hilfsmittel zur Durchführung des Angriffs genutzt wird. Dazu gehören z.B. Phishing-Angriffe, bei denen das Opfer durch manipulierte DNS-Antworten auf einen falschen Server umgeleitet wird. Auch lassen sich durch das DNS DoS-Angriffe auf andere Systeme

ausführen. DNS arbeitet in der Regel mit dem verbindungslosen Protokoll UDP. Anfragepakete an einen Server sind in der Regel deutlich kleiner als von dem Server verschickte Antwortpakete. Wenn ein Angreifer also einem Server eine große Menge von Anfragen mit gefälschter Absenderadresse schickt, wird dieser den Besitzer der Absenderadresse mit einer um ein vielfaches höheren Datenmenge überfluten. DNS-Server können also zur Verstärkung von DoS-Angriffen genutzt werden (*DNS Amplification Attack*)

IV. Geschichte von DNSSEC

Das folgende Kapitel soll einen kurzen Überblick über die geschichtliche Entwicklung von DNS und DNSSEC geben. Diese Übersicht basiert größtenteils auf [**?**] und [?], wo weiterführende Informationen verfügbar sind.

- **1983:** DNS wird erfunden und der erste Server implementiert
- **1988:** DNS beginnt eine Rolle im Internet zu spielen
- **1990:** schwere Sicherheitslücken im DNS werden entdeckt, die Informationen werden zurückgehalten, da quasi alle bisherigen Dienste dadurch unsicher sind
- **1995:** der Artikel von 1990 wird veröffentlicht, in der IETF beginnt man über Sicherheitserweiterungen nachzudenken
- **1999:** DNSSEC scheint bereit zum Deployment zu sein, Implementierung existiert
- **2001:** DNSSEC stellt sich als für den praktischen Einsatz unbrauchbar heraus, da es bei großen Zonen schlecht skaliert. Ein neuer Standard, *DNSSECbis*, wird entworfen aber neue Implementationen sind erforderlich. Da die alte Version von DNSSEC keine praktische Rolle spielt oder gespielt hat, wird im weiteren nur noch von DNSSEC gesprochen, wenn DNSSECbis gemeint ist.
- **2002-2003:** weitere Tests zeigen, dass DNSSECbis jetzt einsatzfähig ist
- **Oktober 2005:** Schweden führt für die TLD .se DNSSEC ein, dies ist damit die erste ccTLD[1] mit DNSSEC
- **2008:** NSEC3 wird spezifiziert um weitere Probleme zu beheben (siehe **??**)

V. Funktionsweise von DNSSEC

Um die Funktionsweise von DNSSEC zu verstehen muss man sich zunächst mit den zugrunde liegenden Ideen, Konzepten und den angestrebten Zielen vertraut machen.

A. Ziele von DNSSEC

Bei DNSSEC soll eine Ende-zu-Ende Sicherheit zwischen dem Besitzer der DNS-Zone und der abfragenden Instanz, dem *DNS-Resolver*, erreicht werden. Dabei soll sowohl vor manipulierten DNS-Anfragen und Antworten, z.B. durch einen Man-in-the-middle oder manipulierte DNS-Caches, als auch vor manipulierten DNS-Servern geschützt werden. Kein Ziel von DNSSEC ist es hingegen, die übertragenen Daten geheim zu halten (also zu verschlüsseln), DoS-Angriffe zu verhindern oder die Offenlegung von DNS-Zonen zu verhindern.

[1]Länder Top Level Domains, Country Code Top Level Domain, ccTLD, im Gegensatz zu generischen TLDs (gTLD) wie .com oder .net)

B. Ideen hinter DNSSEC

DNSSEC arbeitet mit digitalen Signaturen basierend auf asymmetrischer Cryptographie. Dabei erstellt der Zonenbesitzer ein Paar aus öffentlichem und privatem Schlüssel. Der öffentliche Schlüssel wird der DNS-Zone als neuer RR hinzugefügt und ist somit öffentlich abfragbar, der private Schlüssel wird nicht auf dem DNS-Server gespeichert. Für jeden RR der Zone wird nun eine digitale Signatur mit dem privaten Schlüssel erzeugt, die erstellten Signaturen werden ebenfalls der Zone als neue RRs hinzugefügt. Die gesamte Signatur der Zone kann offline erfolgen, anschließend kann die Zonendatei auf die (möglicherweise nicht einmal vertrauenswürdigen) DNS-Server (*Nameserver*) übertragen werden. Der private Schlüssel wird nicht auf den Servern benötigt. Ein DNS-Resolver, der einen Namen auflösen will, kann aus dem DNS nicht nur den entsprechenden Wert ermitteln, sondern auch die zugehörige Signatur und den öffentlichen Schlüssel, womit er die Authentizität des empfangenen Wertes bestätigen kann.

Allerdings muss auch die Authentizität des öffentlichen Schlüssels bestätigt werden, bevor diesem getraut werden kann. Dazu existiert in der übergeordneten Zone neben dem NS-Resource Record, der die eigentliche Delegation der untergeordneten Zone darstellt, ein *Delegation-Signer (DS)* RR. Dieser erhält einen Hash des öffentlichen Schlüssels der untergeordneten Zone. Der DS-Eintrag gehört zur übergeordneten Zone und ist daher mit deren öffentlichem Schlüssel signiert. Kann ein Resolver also dem öffentlichen Schlüssel einer Zone vertrauen, so kann er durch Nutzung der DS-Einträge auch allen öffentlichen Schlüsseln von untergeordneten Zonen vertrauen. Im Idealfall kann ein Resolver also ausgehend vom öffentlichen Schlüssel der Root-Zone (.) allen anderen Zonen vertrauen, da er nach und nach über die DS-RRs die öffentlichen Schlüssel aller untergeordneten Zonen verifizieren kann. Der öffentliche Schlüssel der Root-Zone kann allerdings nicht anderweitig bestätigt werden und muss dem Resolver daher bekannt sein. Selbstverständlich können auch weitere Schlüssel dem Resolver bekannt gemacht werden, um die Auflösung zu vereinfachen.

C. Weitere Abstraktionen

Bei DNSSEC muss ein Kompromiss bezüglich der Schlüssellänge eingegangen werden. Kürzere Schlüssel ermöglichen es, Signaturen schneller zu erstellen und zu überprüfen. Darüber hinaus benötigen die damit erzeugten Signaturen weniger Speicherplatz in den Zonen und verursachen weniger Datenverkehr bei Abfragen. Leider sind kurze Schlüssel weniger sicher gegen Angriffe und müssen daher öfter gewechselt werden. Der Wechsel von Schlüsseln ist aufwendig, da auch der Verwalter der übergeordneten Zone über den neuen Schlüssel informiert werden und den DS-RR aktualisieren muss. Bei DNSSEC wird daher für jede Zone nicht nur ein Paar aus öffentlichem und privatem Schlüssel gebildet, sondern zwei Paar. Ein Paar wird als *Key Signing Key (KSK)* bezeichnet, das andere als *Zone Signing Key (ZSK)*. Der in der übergeordneten Zone existierende DS-RR verweist dabei auf den KSK. Mit diesem wird der

ZSK signiert (beide öffentlichen Schlüssel sowie die mit dem KSK erzeugte Signatur des ZSK befinden sich in der untergeordneten Zone). Mit dem ZSK werden nun wiederum alle anderen Einträge der Zone signiert. Dieses Vorgehen bietet einige entscheidende Vorteile:

- Der KSK kann sehr lang und sicher sein und muss daher nicht oft gewechselt werden.
- Der ZSK kann kürzer sein, dadurch wird weniger Platz benötigt und große Zonen können schneller signiert werden.
- Durch die hohe Lebensdauer des KSK muss der DS-RR nicht oft aktualisiert werden.
- Der ZSK kann oft gewechselt werden, da keine weitere Instanz am Schlüsseltausch beteiligt werden muss. Der KSK bleibt gleich und damit kann der neue ZSK signiert werden. Mit diesem wiederum werden alle Zoneneinträge neu signiert.
- Bei dynamisch aktualisierten Zonen kann der private Schlüssel des ZSK auf dem Server verbleiben um Änderungen automatisch zu signieren, der private Schlüssel des KSK hingegen bleibt offline. Nach einer Kompromittierung des Servers kann die Sicherheit durch Wechsel des ZSK schnell wieder hergestellt werden, der KSK kann dabei beibehalten werden.

D. Ablauf einer Schlüsselüberprüfung durch den Resolver

1) Der Resolver fragt einen Eintrag www.example.com ab (A-Record)
2) von den Nameservern für die Rootzone (.) erhält er einen NS-Eintrag für die com. Zone und einen DS-Eintrag für den dortigen KSK sowie den ZSK und die Signatur für die .-Zone
3) der KSK für die .-Zone ist bekannt, damit wird der .-ZSK bestätigt, damit wiederum der DS-Eintrag
4) von dem com.-*Nameserver (NS)* wird der dortige KSK, ZSK, sowie NS- und DS-RRs für example.com. abgefragt
5) mit dem vorher durch den DS-RR bestätigten .com-KSK wird der dortige ZSK, damit der DS-RR bestätigt
6) von dem example.com-NS wird KSK, ZSK, www.example.com-A-RR und dessen Signatur abgefragt
7) Diese werden der Reihe nach bestätigt

E. Wem vertraut der Resolver

Eine der Kernfragen bei DNSSEC ist die Frage, wem ein Resolver vertrauen muss und wem nicht. Wie aus dem oben erläuterten Ablauf hervorgeht muss der Resolver bei der Auflösung des Eintrags www.example.com genau folgenden Instanzen und Gegebenheiten vertrauen:

- seinem einprogrammierten .-Schlüssel, da der Schlüssel für die .-Zone nicht anderweitig bestätigt werden kann
- den Besitzern der .-Zone, com.-Zone und example.com.-Zone, da der Besitzer einer Zone den dazugehörigen privaten Schlüssel kennt und daher alle darin enthaltenen RRs und Delegierungen auf untergeordnete Zonen ändern kann.
- der Sicherheit der eigenen Ausführung. Wenn der Resolver selber durch Viren oder ähnliches manipuliert wurde ist selbstverständlich keine Sicherheit mehr möglich.

Der Resolver muss jedoch keiner der folgenden Instanzen vertrauen und ist daher vor allen Manipulationen von diesen geschützt:

- einem Serverbetreiber für einen der beteiligten Nameserver
- den Nameservern seines Providers
- anderen zwischengeschalteten DNS-Caches
- der Sicherheit des Kommunikationsweges mit den Servern (z.B. können Antworten, die durch einen Man-in-the-Middle manipuliert wurden erkannt werden)

Einen Sonderfall bilden so genannte Stub-Resolver, die in vielen Betriebssystemen eingebaut sind. Diese sind keine vollwertigen Resolver, die Nameserver rekursiv abfragen und die Antworten per DNSSEC verifizieren können, stattdessen werden alle Anfragen lediglich an einen anderen, vollwertigen Resolver weitergeleitet und von diesem die fertige Antwort erhalten. In diesem Fall, wenn also der Stub-Resolver die Antwort nicht selber verifiziert, muss dieser zusätzlich dem von ihm verwendeten Resolver und der Sicherheit des Kommunikationskanals zu diesem vertrauen. Letzteres kann z.B. mittels IPSec oder TLS sichergestellt werden, dies ist aber nicht Thema dieser Arbeit.

F. Authenticated denial of existence

Eine Besonderheit bei DNSSEC ist die so genannte *authenticated denial of existence*. Wie bisher gezeigt wurde, kann ein Resolver einen Eintrag effektiv auf Authentizität überprüfen. Dadurch können sowohl Manipulationen am Inhalt eines Eintrags als auch das vorspiegeln von nicht existierenden Einträgen verhindert werden. Jedoch bleibt das Problem, dass ein Angreifer die Existenz eines in Wahrheit existierenden RRs verleugnen kann. Der Resolver kann dies erst einmal nicht erkennen, da er ja keine Signatur für das Nicht-Existieren eines RRs erhalten kann. Bei DNSSEC wurde jedoch auch dieses Problem gelöst. Dazu werden zunächst alle Einträge einer Zone alphabetisch sortiert, anschließend wird zwischen je zwei RRs mit unterschiedlichem Namen ein weiterer neuer RR eingefügt. Dieser hat den Namen des direkt davor stehenden Eintrags und als Wert den Namen des folgenden Eintrags und gibt an, dass zwischen diesen beiden Einträgen in alphabetischer Ordnung keine weiteren Einträge liegen. Als Typ wird der neue Typ *NSEC* verwendet. Diese neu erstellten Einträge werden nun wie alle anderen Einträge der Zone signiert. Wenn ein Server nun eine negative Antwort verschickt (also aussagt, dass ein angefragter Name nicht existiert), so schickt er mit dieser Antwort den NSEC-RR, in dessen Bereich der angeforderte Name liegen müsste, würde er existieren und dessen Signatur. Der anfragende Resolver überprüft nun die Signatur und ob der NSEC-Eintrag wirklich den angeforderten Schlüssel enthalten müsste. Ist dies erfolgreich kann der Resolver sicher sein, dass der angefragte Eintrag wirklich nicht existiert, da sonst kein gültiger NSEC-Eintrag für den Bereich existieren könnte.

Wenn beispielsweise die beiden Einträge a.example.com und c.example.com existieren, aber b.example.com angefordert wird, so teilt der Server dem Resolver mit, dass auf den Eintrag a.example.com direkt c.example.com folgt. Daher weiß der Resolver, dass b.example.com nicht existiert. Ebenso ist durch den NSEC-Eintrag bewiesen, dass a1.example.com, bcde.example.com, etc. nicht existieren.

VI. Änderungen am bisherigen DNS

Obwohl DNSSEC weitgehend abwärtskompatibel ist, sind doch einige Änderungen am bisherigen DNS erforderlich. Wie im Abschnitt **??** erwähnt, kennt das DNS verschiedene Typen von Resource Records. Weit verbreitete Typen sind z.B. A für die Abbildung von Namen in IPv4-Adressen, AAAA für die Abbildung von Namen in IPv6 Adressen, CNAME für Aliase, PTR für die Abbildung von IP-Adressen in Namen, TXT für beliebige Texte sowie eine Reihe weiterer, mehr oder weniger oft verwendeter Typen. DNSSEC führt nun eine Reihe neuer Typen ein, im Detail sind das die folgenden:

- **RRSIG** speichert Signaturen für andere RRs
- **DNSKEY** speichert die öffentlichen Schlüssel von KSK und ZSK, mit denen die Signaturen verifiziert werden können
- **DS** speichert einen Zeiger auf den DNSKEY einer untergeordneten Zone (in der übergeordneten Zone)
- **NSEC** speichert, welche Namen in einer Zone nicht existieren (siehe Abschnitt **??**)
- **NSEC3** ist ein sichererer Nachfolger von NSEC (siehe Abschnitt **??**)
- **NSEC3PARAM** speichert Hilfsinformationen für NSEC3

Darüber hinaus sind einige kleinere Änderungen am Protokoll erforderlich, da durch die Übermittlung von Schlüsseln und Signaturen die Pakete deutlich größer werden als bisher. Dies führt dazu, dass alle Komponenten, also Nameserver, Resolver, Caches, etc. aktualisiert werden müssen um DNSSEC zu unterstützen. Allerdings kann diese Aktualisierung schrittweise erfolgen, da neues und altes System zu einander hinreichend kompatibel sind, jedoch entsteht für alte Komponenten kein Sicherheitsgewinn durch DNSSEC.

A. Neue Schlüsseltypen im Detail

1) RRSIG: Ein Beispiel für einen RRSIG-RR könnte wie folgt aussehen:

```
host.example.com. 86400 IN RRSIG A 5 3
     86400 20030322173103 (
     20030220173103 2642 example.com.
     oJB1W6WNGv+ldvQ3WDG0MQkg5IEhjRip8WTr
     PYGv07h108dUKGMeDPKijVCHX3DDKdfb+v6o
     B9wfuh3DTJXUAfI/M0zmO/zz8bW0Rznl8O3t
     GNazPwQKkRN20XPXV6nwwfoXmJQbsLNrLfkG
     J5D6fwFm8nN+6pBzeDQfsS3Ap3o= )
```

host.example.com. ist dabei der Name des RR, 86400 ist die Gültigkeitsdauer (TTL) dieses Eintrags, IN gibt an, dass es um die DNS-Klasse Internet geht. RRSIG ist der Typ des RR, A besagt, dass dies eine Signatur für einen oder mehrere A-Records ist. 5 steht für das verwendete Verschlüsselungsverfahren, 3 ist der Labelcount, welcher für Wildcard-RRs[2] benötigt wird. 86400 gibt noch einmal die TTL an. Dies ist erforderlich, weil der TTL-Wert z.B. durch zwischengeschaltete Proxys verändert werden kann und die Signatur dadurch ungültig werden würde. Daher wird die TTL zum Zeitpunkt des signierens hier noch einmal wiederholt. Anschließend sind die Zeitstempel angegeben, bis wann und ab wann die Signatur gültig ist. example.com. ist der Name dessen, der die Signatur ausgestellt hat, hier also der Besitzer der Zone example.com., wo auch nach dem passenden öffentlichen Schlüssel gesucht werden muss, welche durch den Schlüsselidentifizierer 2642 erkannt werden kann. Anschließend folgt die eigentliche Signatur in Base64-Kodierung.

2) DNSKEY: Als nächstes soll das Format des RR-Typs DNSKEY vorgestellt werden, welches zur Speicherung von öffentlichen Schlüsseln dient.

```
example.com. 86400 IN DNSKEY 256 3 5
     ( AQPSKmynfzW4kyBv015MUG2DeIQ3
       Cbl+BBZH4b/0PY1kxkmvHjcZc8no
       kfzj31GajIQKY+5CptLr3buXA10h
       WqTkF7H6RfoRqXQeogmMHfpftf6z
       Mv1LyBUgia7za6ZEzOJBOztyvhjL
       742iU/TpPSEDhm2SNKLijfUppn1U
       aNvv4w==  )
```

Hierbei lassen sich auf Anhieb gewisse Ähnlichkeiten mit dem obigen Beispiel erkennen. Name, TTL und Klasse sind wieder wie oben, der Typ ist diesmal DNSKEY. die folgenden drei Zahlen geben Informationen zum Verschlüsselungsverfahren, Schlüssellänge, etc. Danach folgt der eigentliche Schlüssel, wieder in Base64-Kodierung.

3) DS: Ein weiterer wichtiger Typ ist der Delegation Signer DS.

```
dskey.example.com. 86400 IN DNSKEY 256 3 5
     ( AQOeiiR0GOMYkDshWoSKz9Xz
       fwJr1AYtsmx3TGkJaNXVbfi/
       2pHm822aJ5iI9BMzNXxeYCmZ
       DRD99WYwYqUSdjMmmAphXdvx
       egXd/M5+X7OrzKBaMbCVdFLU
       Uh6DhweJBjEVv5f2wwjM9Xzc
       nOf+EPbtG9DMBmADjFDc2w/r
       ljwvFw==
     ) ;  key id = 60485

dskey.example.com. 86400 IN DS 60485 5 1
     ( 2BB183AF5F22588179A53B0A
       98631FAD1A292118 )
```

In diesem Beispiel sind zwei RRs aufgeführt, zuerst einmal wieder ein DNSKEY-RR, der in der untergeordneten Zone gespeichert ist. Aus dem Schlüssel lässt sich nach einem

[2]Die genaue Funktionsweise von Wildcard-RRs kann hier nicht erklärt werden, dafür sei auf weiterführende Literatur verwiesen

ebenfalls im DNSSEC-Standard festgelegten Verfahren die Schlüssel-ID 60485 berechnen. In der übergeordneten Zone wird nun der DS-RR eingefügt. Dieser hat natürlich auch wieder einen Namen, TTL, Klasse und den Typ DS. Anschließend folgt die Key-ID und weitere Informationen zum Schlüssel. Zuletzt folgt eine Prüfsumme des Schlüssels in Hexadezimaldarstellung.

4) NSEC: Besonders einfach ist der Aufbau von NSEC-Einträgen.

```
alpha.example.com. 86400 IN NSEC
     host.example.com.
     (A MX RRSIG NSEC TYPE1234 )
```

Auch hier sind wieder die üblichen Felder vorhanden, der Typ ist NSEC. Der Name des RR ist in diesem Fall alpha.example.com., der Wert ist host.example.com., dieser NSEC-Eintrag gibt also an, dass es keine RRs gibt, die in alphabetischer Ordnung zwischen alpha und host liegen würden. Anschließend folgt noch eine Aufzählung der Typen, für die RRs mit Namen alpha.example.com. existieren, in diesem Fall sind das A, MX, RRSIG, NSEC und TYPE1234. Mit Hilfe dieser Informationen lässt sich auch feststellen, ob ein bestimmter RR-Typ für einen Namen nicht existiert.

5) NSEC3 und NSEC3PARAM: Abschließend seinen noch die Formate der Schlüsseltypen NSEC3 und NSEC3PARAM erwähnt, weitere Informationen zur Verwendung dieser Schlüssel finden sich im Abschnitt **??**.

```
example.com. 86400 IN NSEC3PARAM 1 0 12
     aabbccdd

0p9mhaveqvm6t7vbl5lop2u3t2rp3tom.example.com
     86400 IN NSEC3
     1 1 12 aabbccdd (
     2t7b4g4vsa5smi47k61mv5bv1a22bojr
     MX DNSKEY NS SOA NSEC3PARAM RRSIG )
```

Der RR-Typ NSEC3PARAM gibt nur einige Hilfsinformationen zur verwendeten Hashfunktion sowie den verwendeten Salt[3] aabbccdd an und muss nicht weiter beschrieben werden. Der RR-Typ NSEC3 hat große Ähnlichkeit mit dem Typ NSEC, allerdings werden hier nicht direkt die Namen dieses und des nächsten Schlüssels verwendet, sondern Hashwerte von diesen. Als Typ ist NSEC3 angegeben, anschließend folgen Informationen zum verwendeten Hash-Verfahren, der Salt und Informationen zur Behandlung von Wildcard-RRs. Genauere Informationen hierzu finden sich in [**?**]. Auch hier werden wieder die RR-Typen angegeben, die bei dem RR mit dem Hash 0p9mhaveqvm6t7vbl5lop2u3t2rp3tom existieren. 2t7b4g4vsa5smi47k61mv5bv1a22bojr ist der Hash des nächsten existierenden Namens.

VII. Bisherige Verbreitung von DNSSEC

Bisher ist DNSSEC noch nicht sonderlich weit verbreitet, von den Länder Top Level Domains (ccTLDs) verwenden bisher lediglich die folgenden DNSSEC:

- .bg (Bulgarien)
- .br (Brasilien)
- .cz (Tschechien)
- .pr (Puerto Rico)
- .se (Schweden)

Noch schlechter sieht es bei den *generischen Top Level Domains (Generic Top Level Domain, gTLD)* aus, von denen DNSSEC bisher lediglich bei .museum Verwendung findet. Allerdings ist fest geplant, dass die Zone .gov ab Januar 2009 DNSSEC verwendet, bis Ende 2009 sollen des weiteren die Zone .mil sowie alle Unterzonen von .mil und .gov DNSSEC verwenden. Zur Zeit testet die IANA elf IDN-TLDs (internationalisierte Top Level Domains), bei all diesen wird auch DNSSEC getestet.

Darüber hinaus existierten in den Zonen .com, .net, .arpa TLDs Testprojekte für DNSSEC, die aber inzwischen alle wieder eingestellt wurden. Auch in untergeordneten Zonen, also Second- und Third-Level Domains oder noch tieferen Zonen wird DNSSEC verwendet oder wurde bereits damit getestet.

VIII. Probleme von DNSSEC

Leider bringt DNSSEC auch einige Probleme mit sich. Diese Probleme haben verschiedenste Ursachen. Zunächst einmal ist DNSSEC nicht in der Lage alle Probleme des bisherigen DNS zu beheben. Denial of Service (DoS) Angriffe sind weiterhin möglich und werden durch die zusätzlichen Berechnungen für die kryptographischen Überprüfungen und die größeren Datenmengen sogar noch verschärft. Ebenso sind DNS Amplification Attacks weiterhin möglich und durch die bei DNSSEC deutlich größeren Antwortpakete wird auch dieses Problem noch verschlimmert. Darüber hinaus gibt es bei der Einführung von DNSSEC auch diverse politische Probleme. Historisch bedingt haben die USA eine große Kontrolle über das Internet und besonders das DNS. Diese wollen die USA nicht aufgeben, viele andere Nationen hingegen wünschen keine einseitig amerikanische Dominanz des Internets mehr. Diese Problematik trifft DNSSEC, wenn es darum geht, wer die privaten Schlüssel für die Root-(.)-Zone erhält, ebenso ist die Verwaltung der Schlüssel für die generischen Top Level Domains, vor allem .com und und .net noch nicht geklärt. Hierzu gab es bereits viele Vorschläge, diverse US-Behörden, die IANA, die UN und weitere wurden diskutiert. Auch wurde darüber nachgedacht, die Schlüssel auf mehrere Länder aufzuteilen, so dass eine Signierung nur gemeinsam erfolgen kann. Eine Lösung zeichnet sich indes nicht ab. Auch technisch gesehen sind längst nicht alle Probleme aus der Welt. Durch die NSEC-Einträge und die Signaturen steigt die Größe der Zonendateien um 100-500%, was dazu führt, dass die Nameserver gerade für die großen Zonen deutlich mehr Leistung benötigen. Auch wurden bisher im DNS nur relative Zeiten in Form der TTL-Werte verwendet, DNSSEC verwendet bei den Signaturen aber absolute Zeitstempel für die Gültigkeitszeiträume. Dadurch werden erstmals auf allen beteiligten Systemen korrekt funktionierende Rechneruhren vor-

[3]Zur Bedeutung von Salt-Werten für die Sicherheit von kryptographischen Hashes sei hier auf Literatur zum Thema Kryptographie verwiesen

ausgesetzt. Darüber hinaus wurden bei vielen ADSL-Routern schwere Fehler bei der Implementation von DNS gemacht, was dazu führt, dass die DNS-Komponenten dieser Router nicht mehr funktionieren, wenn in den Antworten DNSSEC-Daten enthalten sind. Auch organisatorische Probleme sind zu lösen. Während bisher Zonen, an denen wenig Änderungen durchgeführt werden mussten, teilweise über Jahre ohne Veränderungen und Wartungen auskamen, muss bei DNSSEC jede Zone in regelmäßigen, nicht zu langen Zeitintervallen mit jeweils neuen Schlüsseln neu signiert werden. Für diese regelmäßige Wartung, d.h. den Wechsel des ZSK, muss ein Ablauf zuverlässig etabliert werden. Auch der seltenere Tausch des KSK muss zuverlässig funktionieren, da bei Fehlern hierbei ganze Zweige des DNS unerreichbar werden können, wenn eine einzige Signatur nicht verifiziert werden kann. Auf Seite der Resolver muss eine sichere Verteilung des Root-Schlüssels gewährleistet werden, sowohl erstmalig, als auch bei eventuell notwendigen Änderungen. Auf zwei besondere Probleme von DNSSEC soll im folgenden noch genauer eingegangen werden:

A. Problem des Schlüsselwechsels

Da DNSSEC keinerlei Möglichkeit vorsieht, einmal erstellte Signaturen zurück zu rufen oder vor dem Ablauf ihrer Gültigkeit für ungültig zu erklären, muss die Gültigkeitsperiode für Schlüssel und Signaturen recht gering gewählt werden, damit im Falle einer Kompromittierung eines Schlüssels die zugehörigen Signaturen nicht mehr allzu lang gültig bleiben. Dies führt dazu, dass Schlüssel regelmäßig geändert und neue Signaturen erzeugt werden müssen. Da DNS jedoch keine einfache, zentrale Datenbank ist, sondern Caches verwendet, kann es passieren, dass RRs noch an Resolver ausgeliefert werden, wenn sie in der authoritativen Version der Zone schon nicht mehr vorhanden sind. Dies führt zu Problemen, wenn z.B. ein Schlüssel im Cache vorhanden ist, nicht aber die zugehörige Signatur, da diese evtl. nicht mehr erhältlich ist. Das gleiche Problem tritt auf, wenn eine Signatur im Cache vorhanden, der Schlüssel aber nicht mehr erhältlich ist und führt im Endeffekt dazu, dass die Überprüfung der Signatur fehlschlagen würde. Somit wären alle untergeordneten Einträge und Domänen für DNSSEC beachtende Resolver nicht mehr erreichbar! Für dieses Problem gibt es nur eine Lösung: neue Schlüssel und Signaturen werden in der Zonendatei eine Zeit lang parallel mit den alten vorgehalten. Somit können zu allem, was evtl. noch in Caches vorhanden ist, die zugehörigen Daten weiterhin abgerufen werden und die Cache-Einträge werden nach und nach durch die jeweils neueren Versionen ersetzt. Der Nachteil an dieser Lösung ist natürlich, dass alle Signaturen mehrfach existieren und bei einer Anfrage immer alle Versionen von Signaturen und Schlüsseln ausgeliefert werden müssen. Dadurch steigen Transfervolumen und Zonengröße noch einmal deutlich.

B. Zone enumeration

Ein weiterer sehr wichtiger Fehler von DNSSEC besteht darin, dass der Inhalt von Zonen offengelegt wird. Wie bereits erläutert wurde, ist es möglich, über die NSEC-Einträge nicht nur zu erfahren, dass ein gewisser Eintrag nicht existiert, sondern auch, wie der nächste existierende Eintrag heißt. Ausgehend von diesem Namen kann man nun wiederum den nächsten existierenden Namen mittels der NSEC-Einträge ermitteln. Durch wiederholen dieses Vorgangs ist es möglich, alle Einträge einer Zone mittels der verknüpfenden NSEC-Einträge aufzuzählen und somit den gesamten Inhalt der Zone zu ermitteln. Dies wird als *Zone enumeration* oder *Zone walking* bezeichnet. Während dies zwar dem ursprünglichen Konzept des DNS, nach dem alle Informationen öffentlich sein sollten, nicht widerspricht, ist es heute aus einer Reihe von Gründen oft unerwünscht. Durch die Kenntnis aller Einträge einer Zone lassen sich oftmals Informationen über den internen Aufbau eines Netzwerks, die darin vorhandenen Systeme und deren Rollen und ähnliches ableiten. Dieses Wissen kann für einen späteren Angriff auf dieses Netzwerk sehr hilfreich sein. Auch entstehen in vielen Ländern rechtliche Probleme. So meint zum Beispiel die DENIC, DNSSEC sei unvereinbar mit deutschen Datenschutzgesetzen (siehe [**?**]) und DNSSEC sei daher für die .de-Zone nicht einsetzbar. Viele andere (vor allem europaische) Länder-Registries teilen diese Bedenken. Dieses Thema wird oft kontrovers diskutiert, eben weil eigentlich alle Informationen im DNS öffentlich sind. Jedoch gibt es einen Unterschied, ob nur einzelne Datensätze abgefragt werden können, oder eine Liste aller Datensätze erstellt werden kann. Oft hört man auch das Argument ‘Domain Names können auch anders, z.B. durch Wörterbuchangriffe gefunden werden’, jedoch lässt sich auch das leicht entkräften, wenn man sich die von der DENIC veröffentlichten Zahlen ansieht (siehe [?]). So reichen das deutsche Wörterbuch, das englische Wörterbuch und die Wörterlisten des bekannten Passwortknackers ‘John the Ripper’ gerade einmal aus, um etwa ein Prozent der .de-Zone zu erraten. Selbst ein durchprobieren aller Domainnamen mit einer Länge von acht Zeichen liefert nur dreizehn Prozent der .de-Zone. Selbst wenn man im Besitz der größten Zone des Internets, .com, wäre, könnte man damit nur 42 Prozent der recht kleinen .nl-Zone durch ausprobieren herausfinden. Es scheint also keinen gangbaren Weg zu geben, den Inhalt einer DNS-Zone auch nur annähernd vollständig zu ermitteln. Zwar sind auch heute schon viele Zonen durch falsch konfigurierte Nameserver öffentlich einsehbar, jedoch scheint der Trend eher zu einer Absicherung der Zonendaten zu gehen, wie ich bei meinen Tests der Top-Level-Domains ermitteln konnte. Während im Dezember 2004 noch 141 von 258 TLDs und damit etwa 55% das Herunterladen der Zonendateien von mindestens einem authoritativen Nameserver oder dem im SOA-Eintrag angegebenen Server erlaubten, sind dies vier Jahre später im Dezember 2008 nur noch 78 von 280 TLDs. Nimmt man dazu noch die sechs produktiven Zonen, die DNSSEC verwenden und die elf IDN-Testzonen, welche allesammt die Auflistung der Einträge durch Zone enumeration ermöglichen, sinkt der Prozentsatz der offenliegenden Zonen um 21% auf nun etwa 34%. Auch sind die ungeschützten Zonen kleiner geworden, während 2004 noch große Zonen wie .be mit etwa 42MB und etwa 900.000 RRs und .fi mit etwa 11MB und etwa

235.000 RRs öffentlich waren, sind die größten ungesicherten Zonen heute .sg, .ma und .kz mit 5MB, 4,5MB und 4,2MB.[4]

C. Die Lösung: NSEC3

Zum Glück existiert seit März 2008 eine Lösung für das Problem der Zone enumeration, die es ermöglicht Zonendaten geheim zu halten, dabei aber trotzdem die Nicht-Existenz eines RRs nachzuweisen. Diese Erweiterung von DNSSEC wird als *hashed authenticated denial of existence* oder auch als *NSEC3* bezeichnet. Dabei wird fast genauso wie bei NSEC verfahren, jedoch wird von jedem Namen zuerst ein kryptographischer Hash gebildet, anschließend werden diese Hashes sortiert und die NSEC3-Records erstellt und signiert. Auch mit den Hashes lässt sich nachweisen, dass ein RR nicht existiert, aufgrund der Unumkehrbarkeit der Hash-Funktionen ist es aber nicht möglich, die ursprünglichen Namen wieder zu ermitteln. Dabei finden auch Salts Verwendung, um das Knacken der Hashes zu erschweren, ebenso gibt es Methoden, um Hash-Kollisionen zu vermeiden. NSEC3 ermöglicht es unter anderem der DENIC, DNSSEC einzusetzen, ohne dabei den Datenschutz zu vernachlässigen und bereitet somit den Weg für eine weite Verbreitung von DNSSEC.

IX. Zusammenfassung

Zusammenfassend lässt sich feststellen, dass eine DNS-weite Einführung von DNSSEC technisch möglich ist. Essentiell dabei ist eine Einführung von DNSSEC für die Root-(.)-Zone, da ansonsten für jeden Resolver eine Liste mit einer Vielzahl von vertrauenswürdigen öffentlichen Schlüsseln gepflegt werden muss. Der Aufwand für die Umstellung aller existierenden Systeme (vor allem aller Resolver) ist sehr hoch, dies ist jedoch kein Hinderniss, da die Abwärtskompatibilität vorhanden und eine schrittweise Einführung somit möglich ist. Dabei profitieren zwar ältere Resolver nicht unmittelbar, bis DNSSEC weiträumig eingeführt ist dürften allerdings viele davon ohnehin durch neuere Systeme ersetzt worden sein.

X. Ausblick

Leider steht gerade mit Blick auf die Rootzone zu befürchten, dass politische Probleme die Verbreitung von DNSSEC weiter verzögern dürften. Gerade die jüngere Entwicklung (siehe [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**] und [**?**]) macht auch keine allzu großen Hoffnungen auf eine baldige Einigung. Des Weiteren benötigt die Aktualisierung der Kern-Infrastruktur des DNS, also aller großen Nameserver und Caches, Zeit. Allerdings existieren massive Anstrengungen zur Einführung von DNSSEC, z.B. bei .gov und .mil. Clientseitig hat Microsoft angekündigt, dass Windows 7 und Windows Server 2008 SP2 DNSSEC unterstützen werden. Wie bei allen neuen Technologien, welche weit verbreitete und bewährte Technik ersetzen sollen, besteht bei der Einführung auch hier ein Henne-Ei Problem, solange kaum Zonen DNSSEC verwenden ist das Interesse gering, die Resolver aufzurüsten, so lange kaum Resolver davon profitieren scheuen viele Zonenbesitzer den zusätzlichen Aufwand von DNSSEC. Allerdings eröffnet DNSSEC auch ganz neue Möglichkeiten, das DNS zu nutzen. So ist es z.B. denkbar, E-Mail Zertifikate für Benutzer im DNS zu speichern und somit zuverlässig global zu verbreiten, davon könnte die Verwendung von verschlüsselten und signierten E-Mails stark profitieren und deren Verbreitung zunehmen. Auch sind ganz neue Möglichkeiten der Spam-Bekämpfung denkbar.

[4]Die Größe der DNSSEC geschützten Zonen wurde nicht bestimmt. Insbesondere könnte .se um einige größer sein.

Literatur

[1] B. Claise, "IPFIX protocol specifications," Internet-Draft, draft-ietf-ipfix-protocol-07, December 2004.

[2] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer, "Hash-based IP traceback," in *ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 2001.

[3] A. Belenky and N. Ansari, "IP traceback with deterministic packet marking," *IEEE Communications Letters*, vol. 7, no. 4, pp. 162–164, 2003.

[16] M. Ermert and V. Briegleb. (2007, November) Igf: Politische und technische probleme bei dnssec. heise online. [Online]. Available: http://www.heise.de/netze/IGF-Politische-und-technische-Probleme-bei-DNSSEC--/news/meldung/99000

[17] M. Ermert and A. Wilkens. (2007, Juni) Icann soll rasch rootzone mit dnssec signieren. heise online. [Online]. Available: http://www.heise.de/netze/ICANN-soll-rasch-Rootzone-mit-DNSSEC-signieren--/news/meldung/91501

[18] M. Ermert and J. Kuri. (2007, March) Department of homeland security will den masterschlüssel fürs dns. heise online. [Online]. Available: http://www.heise.de/netze/Department-of-Homeland-Security-will-den-Masterschluessel-fuers-DNS--/news/meldung/87620

[19] M. Ermert and V. Briegleb. (2007, May) Icann soll signatur der dns-rootzone übernehmen. heise online. [Online]. Available: http://www.heise.de/netze/ICANN-soll-Signatur-der-DNS-Rootzone-uebernehmen--/news/meldung/89730

[20] M. Ermert and V. Briegleb. (2007, May) Rootzone-sicherung sorgt weiter für debatten [update]. heise online. [Online]. Available: http://www.heise.de/newsticker/Rootzone-Sicherung-sorgt-weiter-fuer-Debatten-Update--/meldung/89997

[21] M. Ermert and A. Wilkens. (2006, March) Igf: Diskussion über den masterschlüssel für die dns-aufsicht. heise online. [Online]. Available: http://www.heise.de/security/IGF-Diskussion-ueber-den-Masterschluessel-fuer-die-DNS-Aufsicht--/news/meldung/80479

[22] B. Müller. (2008, Juli) Improved dns spoofing using node re-delegation. SEC Consult Unternehmensberatung GmbH. [Online]. Available: http://www.sec-consult.com/files/Whitepaper-DNS-node-redelegation.pdf

[23] S. M. Bellovin. (2005) Using the domain name system for system break-ins. AT&T Bell Laboratories. [Online]. Available: http://citeseer.ist.psu.edu/bellovin95using.html

[24] R. Glauberman. (2004, December) [full-disclosure] again: zone transfers, a spammer's dream? Mailing List. [Online]. Available: http://www.derkeiler.com/Mailing-Lists/Full-Disclosure/2004-12/0812.html

Kryptographische Protokolle SSL/TLS und Web Services

Foued Jaibi
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
jaibi@in.tum.de

ABSTRACT (Kurzfassung)
Die Idee dieser Arbeit ist es, die Grundlagen von Web Sevices, kryptographischen Protokollen SSL/TLS und entsprechende Anwendungen zu charakterisieren und zu bewerten.

Keywords (Schlüsselworte)
SSL/TLS, Rekord protocol, Handshake protocol, Web Services, Security.

1. EINFÜHRUNG
In der Vergangenheit wurden Verschlüsselungstechniken fast nur im militärischen oder diplomatischen Bereich zur Sicherung und Geheimhaltung der Kommunikation genutzt. In der heutigen Zeit hat der „Information“ Begriff eine grosse Bedeutung für die Wirtschaft, insbesondere für deren verschiedene Bereiche wie Medizin, Industrie oder Dienstleistungen erlangt. Die Information ist zu einem unverzichtbaren Produktionsfaktor geworden und stellt sich vor allem als ein Wirtschaftsgut wie jedes andere absatzstarke Produkt dar. Dies ist ein deutlicher Grund dafür, dass gerade im Bereich der IT-Sicherheit intensiv geforscht wird.

2. SSL/TLS

2.1 Geschichte von SSL/TLS
Das Secure Sockets Layer SSL Protokoll wurde von der Firma Netscape entwickelt und zuerst als Version 2.0 im Jahre 1994 publiziert [8]. Netscape wollte damals ihren neuen und kryptographiefähigen Webserver besser auf dem Markt absetzen, indem sie einen freien Client - den „Netscape Navigator“ - zur Verfügung stellten. Dieser Web Browser hatte sicherlich auch die gleichen kryptographischen Protokolle unterstützt wie der entwickeleten Server. Seitdem erschienen auf dem Markt verschiedene neue Spezifikationen. Beispielsweise hat Microsoft ein ähnliches Protokoll, das PCT 1.0 im Jahr 1995 gebracht. Dies wurde in der ersten Version von Internet Explorer integriert. PCT 1.0 hat gegenüber SSL 2.0 einige Vorteile:“Die Umlauf und die Nachrichtenstruktur waren beträchtlich kürzer und einfacher“ [9]. Die Internet Engineering Task Force IETF entwickelte im Jahr 1999 auf Basis von SSL den Standard Transport Layer Security TLS. Im August 2008 erschien mit RFC 5246 die Version 1.2 von TLS, welche somit RFC 4346 (http://www.ietf.org) ersetzt hat. Als wesentliche Änderung steht jetzt keine direkte Abhängigkeit zwischen den Pseudozufallsfunktionen und den MD5/SHA-1 Algorithmen. Stattdessen wurde die Pseudorandom-Funktion neu definiert ([10],[11]).

2.2 Positionierung im Protokollstack
SSL wurde hauptsächlich entwickelt um sichere Internet Verbindungen zu ermöglichen. Es hat als Ziel, Information vor Manipulation, Missbrauch und Zugriff durch einen Unbefugten zu schützen. SSL ist oberhalb der Transportschicht (Bsp. TCP) und unter der Anwendungsschicht (Bsp. HTTP) angesiedelt und ist somit für andere Netzwerkprotokolle geeignet (siehe Abbildung 1) Man trifft besonders beim „Online Banking“ auf Seiten mit „https“. Dies ist nicht anders als in der Spezifikation beschriebene „HTTP over SSL“ Variante [6].

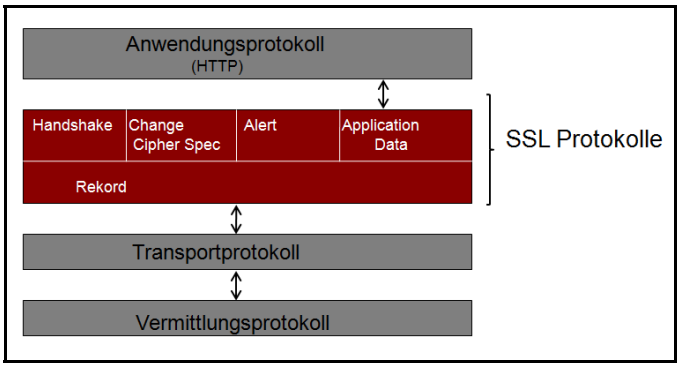


Abbildung 1: SSL im Protokollstack

2.3 Technische Ansatz

2.3.1 Das Rekord Protokoll
Das Rekord Protokoll stellt die untere Ebene des SSL Protokolls dar. Es stellt die Vertraulichkeit und Integrität von Nachrichten als Sicherheitsdienste bereit. Hierbei werden die zu übertragenden Anwendungsdaten in einzelene Pakete fragmentiert und danach kompremiert. In der Spezifikation von SSL/TLS [10] findet man keine explizite Beschreibung zur Kompressionsmethoden. Allerdings wird dort erwähnt, dass die Kompression keinen Verlust der Daten verursachen soll. Bei kleinere Datenmengen kann es zu einer Vergrösserung der Länge kommen. Hier darf die Länge der Daten nicht um mehr als 1024 Bytes vergrössert werden.
Zur Sicherung der Vertraulichkeit werden die zu übertragenden Daten verschlüsselt. Hierzu wird ein geheimer Schlüssel zwischen Client und Server vereinbart. Im Vergleich dazu wird zur Sicherung der Integrität ein MAC (Message Authentification Code) berechnet. Die genaue Funktionsweise vom Rekord Protokoll kann man in der folgenden Abbildung veranschaulichen:

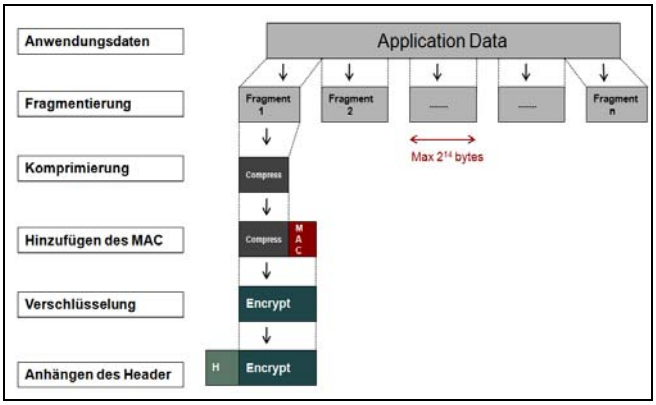


Abbildung 2: Funkionsweise des Rekordprotokolls

2.3.2 Das Handshake Protokoll

Durch das Handshake Protokoll wird zwischen dem Server und dem Client der Modus der Verschlüsselung, die Art der Nachrichtenauthentifizierung, der Schlüssel und alle zur Sicherung der Kanäle notwendigen Details vereinbart. Dieses Protokoll kann daher in vier wichtigen Phasen unterteilt werden [11]:

Phase 1: Festlegung der Ressourcen

Der Client schickt zum Server ein Client_Hello, woraufhin der Server mit einem Server_hello antwortet. Client_Hello kann dabei auch die Antwort auf ein Hello_request sein. Beim Nachrichtenaustausch müssen folgende Parameter bestimmt werden: die Version, eine Zufallzahl, eine Session ID und die zu verwendeten Cipher Suite.

In einer bestehenden SSL-Verbindung sorgt diese Phase außerdem für eine Neuverhandlung der Sicherheitsparameter.

Phase 2: Server Authentifizierung (optional)

Der Server identifiziert sich gegenüber dem Client. Hier wird auch das X509v3-Zertifikat [12] zum Client übermittelt. Außerdem kann der Server ein CertificateRequest an den Client schicken.

Phase 3: Client Authentifizierung(optional)

Hier identifiziert sich der Client gegenüber dem Server. Besitzt der Client kein Zertifikat, so antwortet er mit einem „NoCertificateAlert“. Der Client versucht außerdem, das Zertifikat, das er vom Server erhalten hat, zu verifizieren. Bei Misserfolg wird die Verbindung abgebrochen. Dieses Zertifikat enthält den öffentlichen Schlüssel des Servers. Wird die Cipher-Suite RSA verwenden, so wird das vom Client generierte pre-master-secret mit diesem öffentlichen Schlüssel verschlüsselt und kann vom Server mit dem nur ihm bekannten privaten Schlüssel wieder entschlüsselt werden. Alternativ kann hier auch das Diffie-Hellman-Verfahren[13] verwendet werden, um ein gemeinsames pre-master-secret zu generieren.

Phase 4: Beendigung des Handshake

Hier wird der Handshake beendet. Die change_cipher_spec veranlasst den Server, die gerade ausgehandelten Parameter für die weitere Sitzung zu übernehmen. „finished“ wird dann mit neuen Parametern verarbeitet. Der Server bestätigt mit „change_cipher_spec“ und „finished“.

2.3.3 Das ChangeCipherSpec Protokoll

Das ChangeCipherSpec Protokoll wird benutzt um von einem Verschlüsselungsalgorithmus zu einem anderen zu wechseln. Client und Server verhandeln über eine neue *CipherSpec* und einen neuen *Schlüssel*. Jede Entität schickt eine *CipherSpec* Nachricht, die den Beginn des Kommunikationsprozesses mit neuer *CipherSpec* und neuem *Schlüssel* veranlässt. Im Normalfall ändert sich das CipherSpec am Ende des SSL/TLS handshake. Allerdings kann diese Änderung jeder Zeit stattfinden.

2.3.4 Das Alert Protokoll

Alerts sind spezifische Arten von Nachrichten. Sie werden durch das SSL Record Layer gesendet. Alerts bestehen aus den zwei Teilen AlertLevel und AlertDescription. Beide Teile sind in einer „single-8-bit-number” kodiert.

SSL 3.0 spezifiziert zwei Arten von Alerten, wie Abb.3 zeigt [3].

Alert level	Level name	Meaning
1	Warning	SSL warnings indicate a problem that is not fatal.
2	Fatal	SSL fatal alerts immediatly terminate the current SSL session.

Abbildung 3: Alert levels

3. WEB SERVICES

Web Services stellen heutzutage einen modernen Ansatz zur Realisierung von verteilten Anwendungen dar. Sie sind ziemlich komplizierte Software-Anwendungen, die mit Hilfe von web-basierten und miteinander zu verknüpfenden Standards entwickelt und aufgebaut werden.

Im Folgenden werden Web Services und Web Standards näher beschrieben.

3.1 Was ist ein Web Service

Ein Web Service ist ein Dienst, der über das Internet verfügbar und mit einem eindeutigen Uniform Ressource Identifier (URI) identifizierbar ist [2]. Web Services benutzen einen standardisierten XML-Nachrichtenaustausch-Mechanismus. Dies ermöglicht bei der Kommunikation eine Unabhängigkeit von bestimmten Betriebssystemen oder Programmiersprache. Veranschaulicht wird diese Tatsache in Abb.4 .

Web Services haben zusätzlich zwei wichtige Eigenschaften: Sie sind selbst-beschreibend und einfach entdeckbar. Mit der ersten Eigenschaft ist gemeint, dass der Web Service eine öffentliche Schnittstelle bereitstellen soll. Diese Schnittstelle könnte beispielsweise eine für humane Benutzer verständliche Dokumentation des Dienstes sein. Ebenfalls muss ein Web Service einfach und schnell gefunden werden, damit interessierte Parteien wie Entwicklern oder andere Web Dienste problemlos auf ihn zugreifen können.

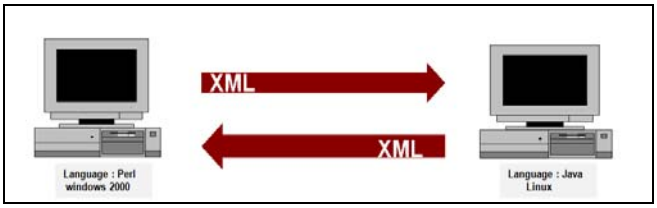


Abbildung 4: Standardisierter XML-Nachrichtenaustausch zwischen zwei verschiedenen Systemen

3.2 Die Architektur von Web Services

Web Services basieren auf einer Service orientierten Architekur (SOA)[2], Dies ist ein sehr verbreiteter Ansatz im Bereich der verteilten Anwendungen, der die Bereitstellung von bestimmten Diensten und Funktionalitäten ermöglicht. Web Services kombinieren daher eine Anzahl von verteilten und objektorientieren Standards um den Autausch von Nachrichten zwischen genau definierten Rollen vorzusehen. Außerdem bietet der sogenannte *Web Service Protocol Stack* (Abb. 6) mit seinen verschiedenen Schichten ein grundlegendes Architektur-modell an, der die einzusetzenden Technologiestandards formal beschreibt.

3.2.1 *Rollen und Aktivitäten*

Wie in der SOA Spezifikation beschrieben ist, setzt auch die Web Service Architektur die Präsenz von genau drei Rollen mit deren entsprechenden Aktivitäten voraus (siehe Abb.5).

Der Dienstanbieter (Service Provider) ist der Anbieter vom Web Service. Er implementiert seinen Dienst, publiziert ihn auf einem über das Internet erreichbaren Server und dokumentiert ihn anhand von einer öffentlichen und von diversen Applikationen lesbaren Schnittstelle damit die Dienstnutzer leicht auf ihn zugreifen können.

Das Dienstverzeichnis (Service Repository) enthält eine logische Beschreibung zu den veröffentlichen Schnittstellen beispielsweise in Form eines Katalogs. Hier werden die verschiedenen Angaben verwaltet.

Der Dienstnutzer (Service Requestor) ist der Dienstkonsument. Er interagiert mit dem Service Verzeichnis mittels XML-basierten Nachrichten. Im Sinne von Web Service Prinzip sind die Dienstnutzer nichts anderes als reine Softwaresysteme.

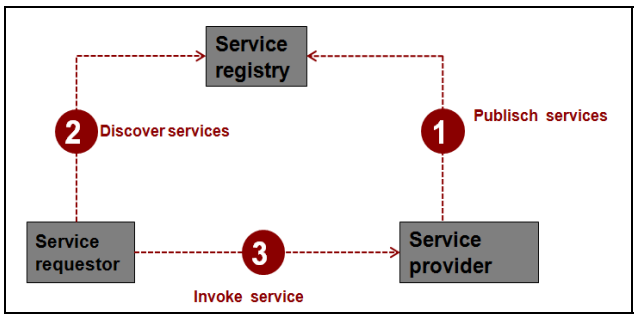


Abbildung 5: Rollen und Aktivitäten in der WS Architektur

3.3 Standards und Technologien

Wie bereits erwähnt bilden die Basistechnologien wichtige Bausteine für die Realisierung von Web Sevices. Diese Standards basiern auf XML und sind im Web Service Protokoll Stack wie folgt beschrieben.

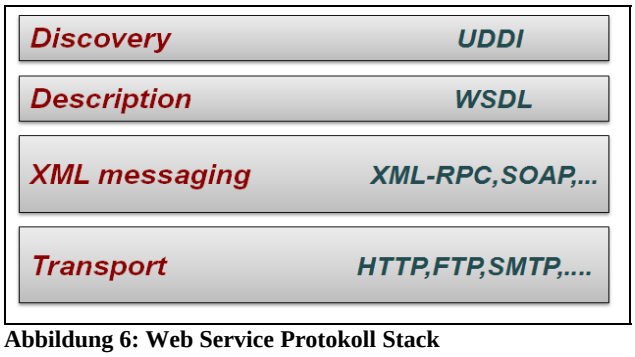


Abbildung 6: Web Service Protokoll Stack

Das Simple Object Access Protocol (SOAP) ist ein Netzwerkprotokoll, das dem Austausch der Daten zwischen zwei verschiedenen Systemen dient. In der Regel wird SOAP für das Remote Procedure Calls durch HTTP verwendet, allerdings sind andere Kommunikationsalternativen möglich (*XML-RPC).* Mittels SOAP kann eine Client-Applikation sehr einfach mit einem Service verbunden werden. Danach kann diese die entsprechend entfernte Methode aufrufen. Es gibt verschiedene andere Technologien (CORBA, DCOM, Java RMI), die die gleiche Funktionalität anbieten [14]. Allerdings besteht die Stärke von SOAP darin, dass die SOAP Nachrichten komplett in XML formuliert sind. Eine totale Plattform- und Programmiersprachen-unabhängigkeit ist daher gewährleistet, welches in Abb. 4 dargestellt ist.

Die Web Services Description Language (WSDL) ist eine Spezifikation zur Beschreibung von Web Services anhand einer allgemeinen XML-Syntax. Mit Hilfe der Metasprache WSDL können also alle wichtigen Informationen zum Aufruf eines Dienstes beschrieben werden. Diese Informationen enthalten die bereitgestellten Funktionen, Daten, Datentypen und Austauschprotokolle eines Web Service. Hauptsächlich wird WSDL in Kombination von SOAP und XML-Schema verwendet.

Die Universal Description, Discovery and Integration (UDDI) [16] ist eine technische Spezifikation zur Beschreibung, Entdeckung und Integration von Web Services. Innerhalb des Web Service Protocol Stack spielt UDDI eine sehr wichtige Rolle. Es erlaubt den Unternehmen, Dienste zum Publizieren und zum Auffinden zu nutzen. Wegen der Verwendung von XML hat ein UDDI Verzeichnis einen Baum als interne Datenstrukur. In dieser Baumstruktur gibt es verschiedene Elemente wie Business-Entities, Busniess-Service, Binding-Template und technische Modelle.

4. Sicherheit für Web Services

Bei der Entstehung des Internets war die Sicherheit keine kritische Frage. In der heutigen Zeit hat das Web viel an Wert gewonnen, besonders im wirtschaftlichen Kontext. Man findet tausende von Geschäftsprozessen einschließlich E-Kommerz, Zahlungsverkehr, Aufträge, Buchungen und viel andere die über das Internet laufen. Allein im Jahr 2008 sind die Umsätze der Online-Händler auf einen Rekordwert von rund 10 Milliarden Euro gestiegen. Daher ist die Gewährleistung der Vertraulichkeit, Verfügbarkeit und Integrität der Dienste erforderlich. Dies erfolgt nur durch die Anwendung von bestimmten Sicherheitmechanismen.

4.1.1 *Sicherheit in Web Services*

In der SOAP Spezifikation [17] wurden keine expliziten Sicherheitsanforderungen erwähnt. Allerdings gibt es für die Umsetzung von Sicherheit in Web Services eine gute Anzahl von definierten Standards [1]: Darunter findet man den bekannten „Web Service - Security“. Dieser Standard wurde von OASIS entwickelt und liegt zurzeit in der Version 1.2 vor [18]. Es existieren außerdem noch die sogenannte Security Pattern wie Message Inspector, Secure Message Router, Front Door [Steel., 2006]. Wegen der intensiven Nutzung von XML Dokumenten bei der Kommunikation zwischen den beteiligten Systemen kommen auch innovative Methoden zum Einsatz. Dabei setzen sich Technologien wie XML-Encryption bzw. XML-Signaturen durch. XML-Encryption Spezifikation beschreibt eine Reihe von Möglichkeiten zur Ver- und Entschlüsselung von XML-Dokumenten, dagegen definiert die XML-Signatur Spezifikation eine XML-Schreibweise für digitale Signaturen [18].

4.1.2 Einsatz von SSL/TLS in Web Services

Ein SOAP basierter Web Service bietet seine Funktionalität über das Web via XML-Nachrichten an, die mittels SOAP übermittelt werden. Eine häufige Kombination ist SOAP über HTTP und TCP, daher ist eine Verwendung von SSL/TLS möglich [4]. (Abb.7)

SSL/TLS bietet auf der Transportebene Vertraulichkeit und Authentifizierung an. Für den Nachrichtenaustausch zwischen genau zwei Partnern stellt SSL/TLS einen guten Lösungsansatz wegen der schnellen Umlauf und Struktur der Nachrichten dar. Ein bekanntes Beispiel dafür ist die Verschlüsselung der Nachrichten bei Banken-Transaktionen.

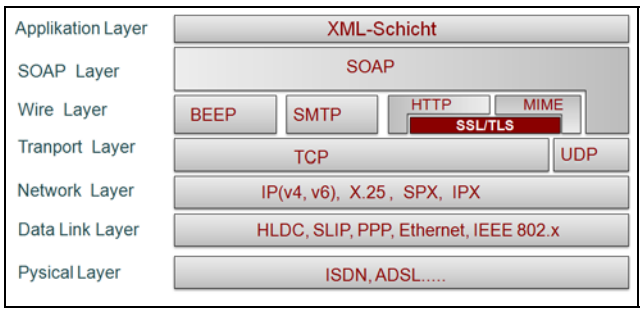


Abbildung 7: Einsatz von SSL/TLS in Web Services

Allerdings sehen die Experten, dass dieses Protokoll für eine umfassende Sicherung von Web Services nicht ausreichend ist. Es treten daher folgende Probleme bei dem Einsatz von SSL/TLS zur Sicherung der Web Dienste auf:

Kosten: Die Verwaltung der Umgebung und die Anschaffung der Technologie für jeden Service Konsumenten und jede Web-Service Kombination ist schwer und kostspielig.

Leistung (Performance): Das Anschaffen und der Aufbau von SSL Verbindungen für jede einzelne Nachricht kann Performanz-Problemen verursachen.
Eine große Anzahl von Teilnehmern bedeutet auch einen großen Aufwand bei der Schlüsselerstellung und Verwaltung.

Sicherheit: Es besteht ein Sicherheitsrisiko bei der Übermittlung von ungesicherten Web Services. Es ist auch keine Certificate Revokation List (CRL) für jeden Nutzer vorhanden, mit der die Ungültigkeit von verschiedenen Zertifikate festgestellt werden können. Die Vertraulichkeit der Daten ist lediglich zwischen zwei Knoten und nicht etwa *von Beginn bis Ende der Übermittlungskette* gewährleistet. Die folgende Abbildung illustriert diesen Kontext:

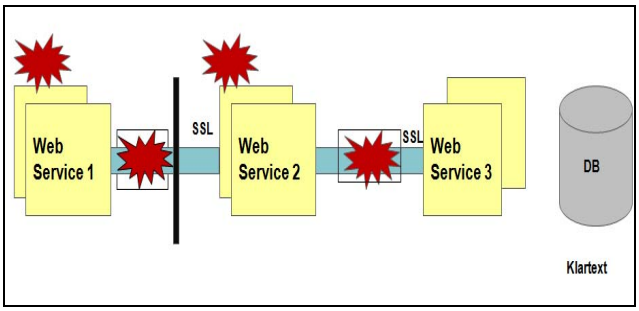


Abbildung 8: Web Service Übermittlungskette

5. Zusammenfassung

Diese Arbeit ist eine grundlegende Einführung in die Kryptographische Verfahren SSL/TLS und ihre Einsatz in Web Services. SSL/TLS stellt eine der verbreitesten Technologien zur Sicherung der Komunikationskanälen. Dies ist auf die Effizienz und die gute Strukturierung der enthaltenen kryptographischen Komponenten zurückzuführen.

Der Einsatz von Web Services ist ein moderner Trend in der Internet Welt. Allerdings treten diese Web Dienste häufiger in isolierten Umgebungen wie das Intranet. Der Grund dafür liegt an der immer noch umstrittenen Rolle von Sicherheitsverfahren (z.B SSL) zur Erfüllung der strikten Sicherheitsanforderungen .

6. Literatur

[1] M.Bichler-TUM Vorlesungsskript: Internetbasierte Gescäftssysteme: http://ibis.in.tum.de/teaching/ws08_09/index.htm

[2] E. Cerami- Web Services Essentials

[3] S. Garfinkel, G. Spafford – Web Security, Privacy & Commerce

[4] P.Kumar –The pros and cons of securing Web Services with SSL

[5] K.Manhart- Sicherheit bei Web Services http://www.tecchannel.de/webtechnik/soa/479383/sicherheit_bei_web_services/

[6] E. Rescorla - SSL and TLS Design and Building Secure Systems: *HTTP over SSL*, Addison-Wesley,2001 ISBN 0-201-61598-3 pp.291-307

[7] J.Schlichter-TUM Vorlesungsskript: Verteilte Anwendungen

[8] SSL Concepts: Hitsory of SSL http://publib.boulder.ibm.com/iseries/v5r2/ic2924/index.htm?info/rzain/rzainhistory.htm

[9] Josh Benaloh, Butler Lampson, Daniel Simon, Terence Spies, Bennet Yee, Microsoft Corp. The Private Communication Technology(PCT) Protocol. Internet Draft

[10] Nau Okamoto, Shigetomo Kimura , Yoshihiko Ebihara,*"An Introduction of Compression Algorithms into SSL/TLS and Proposal of Compression Algorithms Specialized for Application"*, Advanced Information Networking and Applications, 2003 , 17th International Conference.

[11] Transport Secure Layer. Wikipedia. Available at: http://de.wikipedia.org/wiki/Transport_Layer_Security

[12] Repges Markus , *"Einfüfrung in SSL"* Available at: http://www.repges.net/SSL/ssl.html

[13] Diffie, W. and Hellman, M.E.(1976) *"New Directions in Cryptograpy"*, IEEE Transactions on Information Theory (22:6), pp.644-54

[14] Remote Procedure Call. Wikipedia. Available at : http://de.wikipedia.org/wiki/Remote_Procedure_Call

[15] Web Services Description Language (WSDL) Version 2.0 Part1: Core Language http://www.w3.org/TR/wsdl20/

[16] UDDI Version 2 Specifications, OASIS- Committee Specifications http://uddi.org/pubs/ProgrammersAPI-V2.04-Published-20020719.pdf

[17] SOAP Version 1.2, W3C Recommendation (Second Edition) 27 April 2007: http://www.w3.org/TR/soap/

[18] OASIS Web Services Security (WSS) TC Available at: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss

[19] XML Signature Syntax and Processing (Second Edition) W3C Recommendation 10 June 2008. Available at : http://www.w3.org/TR/xmldsig-core/

Identity Management mit OpenID

Innovative Internet-Technologien und Mobilkommunikation WS2008/2009

Steffen Märkl
Lehrstuhl Netzarchitekturen und Netzdienste
Institut für Informatik
Technische Universität München
Email: maerkl@in.tum.de

***Kurzfassung*— Durch den Wandel des Internets in den vergangen Jahren muss sich ein Benutzer zunehmend mehr Logins und Passwörter für die verschiedensten Dienste merken. Zusätzlich müssen immer wieder die gleichen personenbezogenen Informationen zur Registrierung angeben werden, sei es nun bei Amazon, Ebay oder sonstigen Portalen. Bei dieser Fülle von Anmeldedaten und Logins kann der Anwender schnell den Überblick verlieren. Hier setzt Identity Management an. Der Benutzer soll die Möglichkeit haben, sich mit einem einzigen Account bei einem sog. Identity Provider, dem er seine Daten anvertraut, bei allen gewünschten Websites und -applikationen anzumelden. Dabei werden ausschließlich die Daten übermittelt, die er dem jeweiligen Anbieter zur Verfügung stellen möchte. OpenID ist eines der bekanntesten Identity Management Systeme, welches mittels Open Source Ansatz und einer unkomplizierten und einfach zu implementierenden Technik immer mehr Anbieter und auch Benutzer zählt.**

***Schlüsselworte*— Identity Management, Single Sign-On (SSO), OpenID, Identity Provider**

I. Einleitung

In der heutigen Welt, in der webbasierte Dienste immer mehr an Bedeutung gewinnen, ist jeder Nutzer im Internet täglich mit zahlreichen Authorisierungsmechanismen konfrontiert und muss eine Vielzahl von verschiedenen Identitäten verwalten. Benutzerkennung und Passwort werden für viele Tätigkeiten im Internet benötigt, z.B. um Emails online zu verwalten, Bücher in einem Online-Versandhandel zu kaufen, oder Beiträge in einem Blog oder Forum zu erstellen. Zusätzlich zu diesen regelmäßig genutzen Identitäten müssen sich Internetnutzer oft auch für Dienste anmelden, die sie eher selten in Anspruch nehmen. [1] Hier gilt es den Überblick über sämtliche Logins und Passwörter zu behalten und diese im Sinne einer möglichst hohen Sicherheit regelmäßig zu erneuern, sowie nach bestimmten Richtlinien und Vorgaben zu erstellen. Viele Anwender verwenden aufgrund dessen beim Großteil der von Ihnen genutzen Webdienste ein und dieselben Zugangsdaten. Dadurch haben Angreifer oft ein leichtes Spiel an vertrauliche Informationen zu gelangen oder sich der Identitäten ihrer Opfer zu bemächtigen.
In den letzten Jahren sind sog. Identity Management Systeme immer populärer geworden und erfreuen sich immer größerer Beliebtheit. Diese Systeme ermöglichen es dem Benutzer sich seiten- und dienstübergreifend mit einer virtuell einmaligen Identität Zugang zu verschiedensten Internet-Plattformen und -Diensten zu verschaffen, ohne sich dabei mehrere Zugangsdaten merken zu müssen. Der nachfolgende Artikel befasst sich in erster Linie mit der generellen Frage nach den Vor- und Nachteilen von Identity Management im World Wide Web, dessen Verwendungszweck sowie der genauen Funktionsweise, aber auch mit dem zukünftigen Potenzial. Exemplarisch wird hier einer der momentan bekanntesten Vertreter von Identity Management Systemen, OpenID, betrachtet.

II. Identity Managment

„Als Identity Management (IdM) wird der zielgerichtete und bewusste Umgang mit Identität, Anonymität und Pseudoanonymintät bezeichnet.“ [2] Mit der Einleitung des Internets in das sogenannte Zeitalter des Web 2.0 hat auch die Fragestellung nach der bewussten und unbewussten Preisgabe von privaten Daten und Informationen einen bisher unbekannten Grad an Komplexität erreicht. Sowohl erfahrene, aber vor allem unerfahrene Nutzer verlieren schnell den Überblick über ihre Identitäten, mit denen sie sich im Internet bewegen. Hier soll IdM Abhilfe schaffen. Ziel ist eine konsistente, authentische und dauerhafte Bereitstellung von personenbezogenen Daten und die damit einhergehende Redundanzfreiheit sowie „eine hinreichende Sicherheit über die Identität der Online-Kommunikationspartner zu bekommen“ [2], ohne gleichzeitig unnötig viele personenbezogene Details austauschen zu müssen. Die manuelle Anmeldung durch den Benutzer soll dabei so selten wie möglich durchgeführt werden.
Mittlerweile gibt es eine Vielzahl an IdM Systemen, welche alle unterschiedliche Eigenschaften und Funktionsumfänge besitzen. Einige von ihnen sind sehr umfangreich und decken zusätzliche Attribute, wie z.B. Zugriffsrechte mit ab, andere wiederum sind etwas schlanker, um das System möglichst einfach und kompatibel zu halten.

III. Single Sign-On

Befasst man sich mit IdM, so ist oft von dem Begriff des Single Sign-On (SSO) die Rede. Das Ziel des SSO ist es, „dass ein Benutzer nach einer einmaligen Authentifizierung“ [3], durch Kennworteingabe o.ä., „auf alle Rechner und Dienste, für die er berechtigt ist, zugreifen kann, ohne sich jedes Mal neu anmelden zu müssen.“ [3] Nach der initialen Identifikation durch den Benutzer übernimmt das SSO-System von nun an die Aufgabe, den Anwender an diversen Diensten und Systemen zu identifizieren. Hierbei darf das SSO System

dem ursprünglichen Authentifizierungsverfahren, in puncto Sicherheit in nichts nachstehen. Bei IdM Systemen handelt es sich folglich um, zum Teil erweiterte, SSO Systeme, in unterschiedlichem Umfang.

A. Vorteile

Die Vorteile von SSO Systemen:

- Die einmalige Authentifizierung zu Beginn spart viel Zeit die der Anwender normalerweise für das Suchen bzw. Eingeben der Anmeldeinformationen verschwendet. Dieser Zeitfaktor fällt mit steigender Zahl der benutzen Websites und Dienste zusehends mehr ins Gewicht. Oft ist es für die Anwender, sollten sie unterwegs sein, gar nicht möglich stets alle benötigten Login-Daten für alle, evtl. auch spontan genutzen, Dienste mit sich zu führen, geschweige denn sich zu merken.
- Durch die Tatsache, dass sich der Benutzer nur noch eines anstatt einer Vielzahl von Passwörtern merken muss, kann dieses dafür, durch Erhöhung der Komplexität, umso sicherer gewählt werden.
- Das Passwort muss nur noch einmal zum SSO-Server übertragen werden und bietet somit weit weniger Angriffmöglichkeiten als die multiple Authentifizierung auf mehreren Websites. Durch diesen einzigen Angriffspunkt und das einmalige Senden von UserID und Passwort werden vor allem *Phishing-Attacken* [3] erschwert und die Administration von Sicherungsmechanismen, wie z.B. SSL, erleichtert.
- Es gibt SSO- und IdM-Systeme, die mit *SmartCards* oder bzw. und anderen kryptographischen *Token* gekoppelt werden, wodurch ein sicheres Authentifizierungsverfahren ermöglicht wird. Voraussetzung ist natürlich, dass man die SmartCard oder das Token nicht verliert.
- Die Benutzerdaten sind nur noch in einem Benutzerkonto bzw. den damit verbundenen Identitäten hinterlegt. Somit ergibt sich durch die zentrale Datenhaltung eine Verbesserung der Konsistenz und eine erleichterte Administration.
- Der Benutzer an sich wird sensibilisiert nicht mehr überall seine Benutzerdaten zu hinterlassen. Durch den Gebrauch des SSO- bzw. IdM-Systems wissen deren Anwender genau auf welcher Website sie ihr, in anderen Fällen womöglich mehrfach genutztes, Passwort eingeben dürfen.

B. Nachteile

Ebeneso ist es auch wichtig sich der Nachteile, die SSO Systeme mit sich bringen, bewusst zu werden:

- Der bereits bei den Vorteilen (III. A.) genannte Punkt der zentralen Datenhaltung und der Einfachheit eine UserID und ein Passwort für alle Websites und Dienste zu verwenden, entpuppt sich bei genauerer Betrachtung als ein „Single Point of Attack“. Hat ein Angreifer erst einmal die Identität eines Benutzers angenommen, so hat er von diesem Moment Zugriff auf all dessen Dienste. Dieser Punkt ist selbstverständlich nur insofern ein Nachteil, als dass der Benutzer jeden seiner Dienste ansonsten mit einem separaten, sicheren Passwort absichern würde und die Authentifizierung am SSO-System nicht mit der erwähnten SmartCard- oder Token-Unterstützung stattfindet.
- Die Entscheidung für eine Authentifizierung der Benutzer durch einen bestimmten SSO- bzw. IdM-Service birgt auch Risiken für die Betreiber der Dienste. Sollten potenzielle Benutzer dem verwendeten System und dessen Sicherheit kein Vertrauen schenken, so könnten diese als Kunden verloren gehen.
- Der bereits mehrfach erwähnte, zentrale Authentifizierungs- und Authorisierungsdienst ist stets abhängig von der „Verfügbarkeit des Single Sign-On Systems“ [3]. Hieraus ergibt sich ein sog. „Single Point of Failure“. Sollte der SSO-Dienst einmal nicht funktionsbereit sein, sei es durch einen Hardwaredefekt, Wartungsarbeiten o.ä., so würde auch der Zugriff auf sämtliche Systeme in dieser Zeit verwehrt bleiben.

IV. OpenID

A. Einführung

OpenID und das zugrundeliegende Protokoll wurden „ursprünglich im Jahr 2005 von Brad Fitzpatrick, dem Chief Architect der Firma Six Apart Ltd.“ [4] und „Gründer von LiveJournal“ [5], entwickelt. Bei dem IdM-System OpenID handelt es sich um ein offenes, dezentral angelegtes [6] Open Source System für digitale Benutzeridentitäten. Es bringt die bereits im vorherigen Abschnitt über SSO-Systeme erläuterten Vor- und Nachteile mit sich und ersetzt klassischen Anmeldeprozess, durch Eingabe von Benutzernamen und Passwort pro verwendetem Dienst.
Der Unterschied zu anderen großen IdM-Systemen wie Shibboleth [7] oder dem Liberty Alliance Project [8] ist, dass es sich bei OpenID um ein wesentlich schlankeres, weniger komplexes, doch dadurch auch funktionsärmeres System handelt, welches zudem einen anderen Ansatz verfolgt.
Unterstützt eine Seite OpenID, so muss hier zur Anmeldung lediglich eine OpenID-Identität in Form einer URL, z.B. benutzer.provider.tld, angegeben werden, welche man bei der Registrierung bei einem OpenID Provider bekommt. Der Authentisierungsprozess wird beim Provider, und nicht mehr beim Betreiber der besuchten Website, durchgeführt. Abbildung 1. zeigt die Weboberfläche mit Anmeldemaske eines der bekanntesten Vertreter, *myOpenID.com*.

Der Benutzer hat die Möglichkeit sich den OpenID Provider auszusuchen, der seinen Ansprüchen, vor allem in puncto Sicherheit und Vertrauen, am nächsten kommt. Besonderheit ist hierbei, dass im Prinzip jeder, bedingt durch die dezentrale und komplett freie Gestaltung, einen OpenID Server betreiben kann und somit zum OpenID Provider für wiederum andere Benutzer werden kann Eine Registrierung oder Zustimmung irgendeiner Organisation ist dafür nicht notwendig. Der Ansatz von OpenID wird auch als *user-centric* [6] bezeichnet. Vertreter dieses Ansatzes stellen den Zugang bereit und lassen Benutzern die Kontrolle darüber wie ihre Identität online verwaltet und verwendet wird. Die Identity Provider, auf denen

Abbildung 1. Startseite des OpenID Providers myOpenID.com

die Benutzerdaten hinterlegt sind, sind dezentralisiert und, im Gegensatz zu den sog. *institution-centric* IdM Systemen, wie Shibboleth, gehören sie keiner Organisation an. Der Unterschied ist also, dass bei Shibboleth eine Institution versichert, dass der Benutzer der ist der er vorgibt zu sein, während dies bei OpenID dem Benutzer überlassen wird. [9] Hierdurch kann sich jeder Benutzer im Prinzip vor kommerziellem *Data Mining* durch Organisationen schützen, indem er einfach seinen eigenen Provider betreibt. In diesem Fall verlassen seine privaten Daten die eigenen vier Wände nicht ohne seine ausdrückliche Zustimmung. [10]

Nach der bisherigen Betrachtung stellt sich die Frage inwiefern sich OpenID von einem normalen SSO-System unterscheidet. Die großen Unterschiede sind zum einen die bereits erwähnte dezentrale Gestaltung mit einer großen Menge an Identity Providern und der damit verbundenen Entscheidungsfreiheit wo man seine Daten hinterlegt, zum anderen aber auch die Möglichkeit beim Provider mehrere Angaben, so genannte Attribute, zur Identität zu hinterlegen. Diese Option wird auch als *Attribute Exchange (AX)* bezeichnet. So kann man z.B. seinen vollständigen Namen, seine Email-Adresse, seine bevorzugte Sprache oder seine Geburtsdaten speichern. Diese Daten können wiederum als Identitäts-Profile gespeichert werden. Von nun an wählt der Anwender bei jeder Anfrage des Webservers (bzw. der Webapplikation), auf dem er sich mittels OpenID einloggen möchte, aus, welches Profil der Provider an diesen übermitteln darf. [11] Der Anwender hat somit stets die volle Kontrolle über seine personenbezogenen Daten und weiß zu jeder Zeit welchem Anbieter er welche Informationen mitteilen möchte. Laut OpenID.net befindet sich das Projekt immer noch in der Einführungsphase, erfreut sich jedoch, bedingt durch die Akzeptanz und Verbreitung durch große Organisationen wie Yahoo, „AOL, Microsoft, Sun, Novell, etc.“ [6], immer größerer Popularität. Es wird geschätzt, dass es mittlerweile über 368 Millionen (Stand: Januar 2008) OpenID-URIs gibt, mit knappen Zehntausend Seiten die den Login mittels OpenID unterstützen. [5]

B. Entitäten

Besucht ein Benutzer eine Website und loggt sich mit einer OpenID dort ein, so findet eine Kommunikation zwischen dem Webbrowser und dem Webserver statt. Der Webserver wiederum kommuniziert mit dem Identity Provider, der Benutzer mit dem Identity Provider und später wieder der Benutzer mit dem Webserver. Doch das ist nur ein grober Überblick. Der genaue Protokollablauf wird im Unterpunkt „Funktionsweise“ geschildert. Zuvor gilt es noch einige Begrifflichkeiten zu klären. Abbildung 2. soll hierbei einen Überblick über die Entitäten, die an einem OpenID Anmeldevorgang beteiligt sind, geben und deren Verbindungen untereinander veranschaulichen.

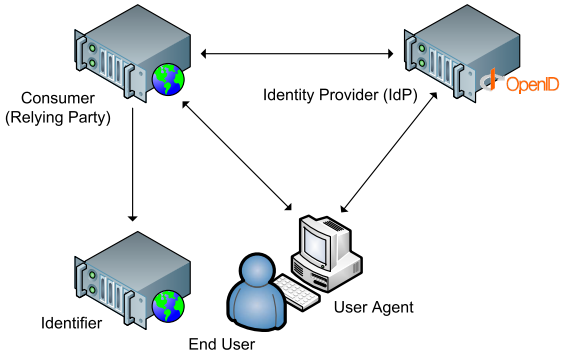


Abbildung 2. Komponenten (Entitäten) die bei einem OpenID-Anmeldevorgang beteiligt sind.

- End User:
  Hierbei handelt es sich um den Benutzer, der sich mit seiner OpenID bei einer Website anmelden möchte.
- Consumer:
  Der Consumer ist die Website bzw. der Webserver an dem sich der Benutzer, mittels OpenID, anmelden will. Damit ist er der eigentliche Konsument der vom Identity Provider gelieferten Daten bzw. des gesamten Dienstes. Folglich sind alle Websites, die eine Anmeldung mittels OpenID gestatten, Consumer. In der OpenID Spezifikation 2.0 [6] wird er auch als Relying Party bezeichnet, da er den vom Identity Provider gelieferten Informationen vertraut.
- Identifier:
  Der Identifier ist eine Website die der End User als OpenID angeben kann. Handelt es sich bei dem Server nicht gleichzeitig um einen Identity Provider, so befindet sich hier ein Verweis auf diesen. Der Consumer erfährt also von dem Identifier an welchen Identitiy Provider er sich bezüglich der Authentifizierung wenden kann.
- Identity Provider (IdP):
  Beim IdP handelt es sich um den Server wo die eigentliche Authentifizierung abläuft. Außerdem sind auf ihm die Benutzerdaten hinterlegt. Die OpenID URI verweist entweder direkt auf diesen IdP oder der Identifier. Während des Authentifizierungsvorgangs tauscht der Consumer Informationen mit dem auch OpenID Provider (OP) ge-

nannten Server aus um eine ID auf ihre Richtigkeit hin zu prüfen.
- User Agent:
  Der User Agent ist der Internet Browser mit dem der End User, durch Tastatur- und Mauseingaben, direkt kommuniziert.

C. Freiheit bei der Wahl der OpenID

Da sich OpenID, anders als diverse andere IdM Systeme, auf Webanwendungen konzentriert, liegt es auf der Hand, dass die Benutzer sich nicht durch klassische Benutzernamen, sondern durch URIs (Uniform Ressource Identificators) bzw. URLs (Uniform Resource Locators) ausweisen. Zum Einsatz kann hier prinzipiell jede beliebige Webadresse kommen, solange ein Browser sie aufrufen darf. [11] Jeder Nutzer einer OpenID muss sich selbstverständlich zuerst bei einem Identity Provider, z.B. myopenid.com, pip.verisignlabs.com oder myid.net (vollständige Liste unter wiki.openid.net/OpenIDServers), anmelden um eine initiale ID zu bekommen. Danach kann er sich jedoch jede beliebige URL als OpenID nehmen die er besitzt. „In dieser Seite muss der Webentwickler lediglich einen *Tag* als Verweis auf den eigentlichen Dienstleister, in den Quellcode der Seite, einbauen." [11] Genaueres dazu im folgenden Abschnitt „Identifier und Identity Provider" (IV. D.). Eine Vielzahl von Benutzern besitzen sogar schon eine OpenID, ohne es zu wissen. Wer bereits einen Benutzeraccount bei einem der in Abbildung 3. aufgelisteten „Webanwendung wie Blogger oder einem Portal wie AOL oder Yahoo hat, darf die dort vorhandenen Authentifizierungsmechanismen nämlich auch über das OpenID-Protokoll nutzen." [11]

Abbildung 3. Identitätsprovider einiger Webanwendungen [6]

D. Identifier und Identity Provider

In der Regel wird sich ein Benutzer bei einem IdP, durch Anmeldung, eine OpenID holen und diese fortan zum Login auf diversen Websites verwenden. Dies ist allerdings nicht verpflichtend. Theoretisch kann jede beliebige URL als OpenID verwendet werden. Ausschlaggebend hierfür sind die Informationen die auf der als OpenID angegebenen Seite, der sog. *OpenID Identity URL Page* [12], vorhanden sind. Hier muss ein Tag im Quellcode der Website einen Verweis auf den eigentlichen IdP enthalten. (Siehe Abbildung 4.)
Durch den ersten Eintrag wird der IdP (hier www.myopenid.com) bekanntgegeben der benutzt werden soll um den Besitzer der OpenID (hier beispielname.myopenid.com) in der zweiten Zeile, zu verifizieren. Duch Einbauen dieser zwei Zeilen in die eigene Homepage oder den eigenen Blog, sind die Anwender also in der Lage ihre persönliche Adresse als OpenID zu verwenden und diese je nach Lust und Laune zu ändern. [11]

```
<link rel=\openid.server\
href=\www.myopenid.com\/>

<link rel=\openid.delegate\
href=\beispielname.myopdenid.com\/>
```

Abbildung 4. HTML Code für die OpenID Identity URL Page

E. Funktionsweise

In dem folgenden Abschnitt wird der genaue Ablauf der einzelnen, am Authentifizierungsvorgang beteiligten, Komponenten veranschaulichen, erklärt und vor allem auf die zwei alternativen Kommunikationsmodi zwischen Website oder Webserver und deren Unterschiede eingegangen. Die nachfolgenden Sequenzdiagramme und Erklärungen sind in Anlehnung an das „OpenID Book" [12] und die OpenID Specification 2.0 [13] entstanden.
Man unterscheidet zwischen zwei grundlegenden Modi, die je nach „Intelligenz" des Consumers bzw. dessen Konfiguration zum Einsatz kommen: dem „Smart mode" und dem „Dumb mode". Ein intelligenter Consumer (Smart Mode) hat anfangs etwas mehr Arbeit und dafür gegen Ende weniger, benötigt hierfür allerdings *lokales Caching* von Statusinformationen. Ein Consumer im Dumb mode hingegen ist komplett zustandslos und benötigt daher einen zusätzlichen *HTTP request*. [14]

1) Smart mode: In diesem Modus vereinbaren der Consumer und der IdP ein sog. *shared secret*. Dabei handelt es sich um einen zeitlich begrenzt gültigen, gemeinsamen Schlüssel, welcher gecached und verwendet wird um die künftigen Kommunikationsnachrichten zu authentifizieren. Dieses shared secret kann entweder im Klartext übertragen oder per Diffie-Hellman-Verfahren zum Austausch von Schlüsseln erzeugt werden. [14] Dadurch wird der HTTP Verkehr zwischen Consumer und IdP am Ende des Authentifikationsprozesses reduziert. Dieser Schritt ist zwar optional, aber nach OpenID Spezifikation 1.1 [14] empfohlen und auch üblich.
Der genaue Ablauf ist im Folgenden und zuerst für den ersten Login (Abbildung 5.) und danach für die darauffolgenden Logins (Abbildung 6.) beschrieben.

a) Erster Login:

1) Der Benutzer besucht die Website des Consumers auf der er sich mittels OpenID einloggen möchte. (z.B. livejournal.com)
2) Er bekommt die Seite vom Consumer geliefert.
3) Nun wählt er bei der Anmeldemaske die Option zur Authentifizierung mittels OpenID auf der Website aus, trägt seine OpenID URL (OpenID Identity) ein (z.B. beispielid.myopenid.com) und startet mit der Übermittlung an den Webserver den Anmeldevorgang.
   Die Hauptaufgabe des Webservers ist nun herauszufinden ob der Benutzer, der sich soeben mit einer ID anmelden will, wirklich die Identität besitzt die er im vorherigen Schritt vorgibt zu besitzen. Der Consumer extrahiert nun aus der OpenID URL die Second-Level-Domain (oder „OpenID Provider Endpoint URL“) des Identifiers und nimmt Kontakt zu diesem auf. Dabei muss es sich nicht unbedingt um den IdP handeln. (Schritt 3a) (Hierfür kann seit der Spezifikation 2.0 [13] auch Yadis [15] zum Einsatz kommen.)
4) Nachdem sich der Consumer die Site des Identifiers geholt hat, wird diese geparst um die Adresse des IdP zu ermitteln. Die Information hierfür ist im Code der HTML Webpage enthalten. Dieser gesamte Prozess wird auch als *Discovery* bezeichnet. Ist der Identifier nicht gleichzeitig der IdP so leitet der Consumer den User Agent, mittels *Browser redirect*, an den richtigen IdP weiter, um die Richtigkeit der vom Benutzer angegebenen Identität einzuholen. Dies geschieht mit der *HTTP GET* Methode. Optional kann der Consumer an diesem Punkt auch eine Verbindung mit dem IdP aufbauen und mit diesem ein shared secret austauschen, welches durch den Austausch von Public Keys, mit Hilfe des Diffie-Hellman-Verfahrens, auf beiden Seiten erstellt werden kann. Der IdP benutzt dieses shared secret um die nachfolgenden Nachrichten zu signieren und der Consumer um diese zu verifizieren. Hierdurch entfällt die Notwendigkeit von direkten Anfragen zur Verifizierung der Signierung nach jedem *Authentifizierungs-request* oder *-response*. [13] (Schritt 4a)
5) Vorausgesetzt der End User ist noch nicht beim IdP eingeloggt, so bekommt er nun dessen Anmeldemaske geschickt. Hier muss er nun das zugehörige Passwort zu seiner OpenID eingeben. Der Authentifizierungsprozess des Benutzers am IdP ist nicht Teil der Spezifikation und bleibt somit dem Provider selbst überlassen. Sollte der Benutzer bereits eingeloggt sein, so entfällt dieser Schritt.
6) Der IdP gibt nun eine Nachricht mit seiner Signatur, wiederum mittels Browser redirect, an den Consumer zurück. Diese gibt Auskunft über die erfolgreiche oder fehlgeschlagene Anmeldung. Zum Einsatz kommt wieder die HTTP GET Methode. Wichtig ist, dass es sich hierbei um eine indirekte Kommunikation zwischen IdP und Consumer handelt.
7) Mit dem shared secret, welches sich im Cache befindet, kann der Consumer nun die signierte Nachricht verifizieren und somit ermitteln, ob der Identity Provider die Nachricht wirklich signiert hat. Handelt es sich bei der Nachricht um eine Mitteilung über eine positive Authentifizierung und konnte der IdP als Absender verifiziert werden, so wird der End User auf der Website des Consumers eingeloggt.

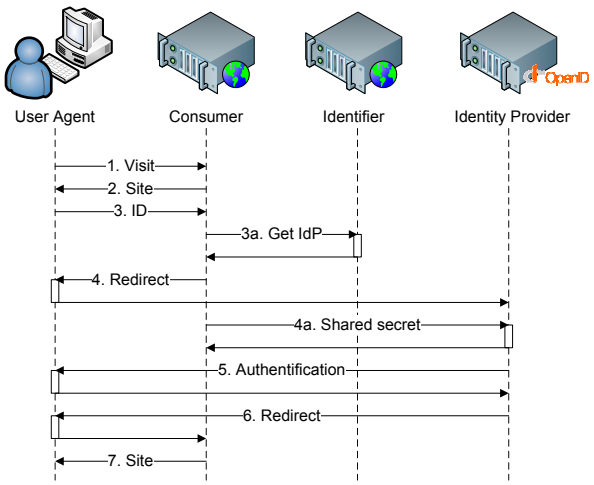


Abbildung 5. Kommunikationsablauf im Smart mode beim ersten Login.

b) Folgende Logins: Hat sich ein Benutzer zuvor schon bei seinem IdP eingeloggt und der Consumer hat das shared secret mit dem IdP bereits in seinem Cache, so wird die Kommunikation der Entitäten untereinander sehr simpel. Der End User kann sich beim Consumer einloggen ohne zuvor interaktiv mit dem IdP kommunizieren, d.h. ohne sich beim IdP nochmals anmelden, zu müssen. Er bekommt dann direkt, nach der Eingabe seiner OpenID URL beim Consumer, sofortigen Zugang zur Website. Hierzu können vom IdP verschiedene Techniken, wie aktive *Browser Sessions* oder *Cookies*, verwendet werden.

1) Der Benutzer besucht die Website des Consumers.
2) Er bekommt die Website mit Anmeldemaske zurück.
3) Der Benutzer gibt seine OpenID URL ein und schickt diese zurück an den Consumer. Dieser leitet daraus wieder die URL des Identifiers ab und nimmt Kontakt zu diesem auf. (Schritt 3a)
4) Nachdem er sich die Seite des Identifiers geholt hat und diese geparst hat, erfolgt die Weiterleitung des User Agents, mittels Browser redirect, an den IdP zur Überprüfung der angegebenen Identität.
5) Der IdP gibt die Information über die erfolgreiche oder fehlgeschlagene Authentifizierung der OpenID mit seiner Signatur, via Browser redirect, an den Consumer zurück.
6) Im Anschluss daran verifiziert der Consumer die Information mittels des im Zwischenspeicher liegenden shared secret.

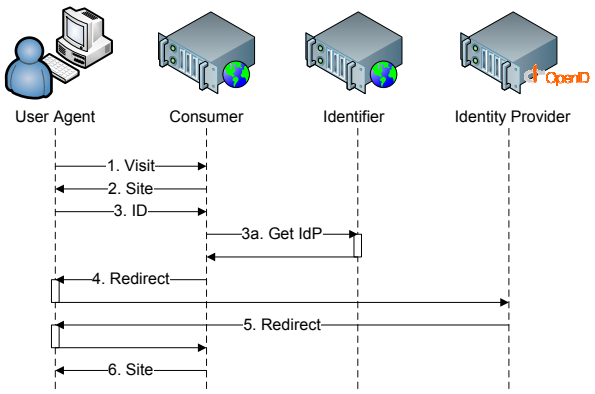


Abbildung 6. Kommunikationsablauf im Smart mode mit bereits bestehendem shared secret.

2) Dumb mode: Dieser Modus kann verwendet werden, wenn der Consumer außerstande ist shared secrets zu erstellen oder diese in einem Cache zu speichern. Daher wird er auch als *Stateless mode* bezeichnet. [13]
Der Consumer muss, anders als beim Smart mode, einen extra Schritt durchführen um den Benutzer zu authentifizieren. Es gibt demnach einen HTTP request und folglich eine response message mehr zwischen Consumer und IdP. Der genaue Ablauf ist im Folgenden beschrieben.

1) Der End User besucht die Website des Consumers.
2) Der End User bekommt die Website mit der Anmeldemaske zurück.
3) Hier hat der User die Möglichkeit seine OpenID URL einzugeben und diese per Submit zu übermitteln. Der Consumer extrahiert die Identifier URL und kontaktiert diesen. (Schritt 3a)
4) Nachdem sich der Consumer die Seite des Identifiers geholt hat, wird diese geparst und der Standort bzw. die IP oder URL des IdP ermittelt. Nach diesem Vorgang leitet der Consumer den User Agent, mittels Browser redirect, zum IdP weiter um die benötigten Informationen über die Gültigkeit der angegebenen OpenID zu bekommen.
5) Nun bekommt der End User die Anmeldemaske des IdPs und muss sich bei diesem eingeloggen.
6) Der Identity Provider schickt dem Consumer nun, wieder via Browser redirect, eine Nachricht über eine erfolgreiche oder fehlgeschlagene Authentifizierung.
7) Sollte die Nachricht erfolgreich sein, so stellt der Consumer nochmals eine direkte Verbindung mit dem IdP, vorzugsweise über SSL, her. Er fragt dann die Authentifizierungsinformationen direkt beim IdP an und vergleicht diese, inklusive der Rückgabe URL, mit der umgeleiteten Information, die er über den User Agent bekommen hat. Mit diesem Verfahren der doppelten Überprüfung soll sichergestellt werden, dass der User Agent (bzw. der End User oder ein potenzieller Angreifer) keine Möglichkeit zum Betrug hat.
8) Stimmen die beiden Informationen aus den vorhergehenden Schritten überein, so wird der End User erfolgreich auf der Website des Consumers eingeloggt.

Der *Dumb mode* sollte ausschließlich verwendet werden, wenn dies der Consumer verlangt bzw. ihm die nötigen Fähigkeiten für den Smart mode fehlen. Je nach Implementierung des OpenID Providers kann dieser Modus ohne Caching verwundbar gegenüber sog. *Replay-Attacken* sein und sollte deshalb, wenn möglich, vermieden werden. [16]

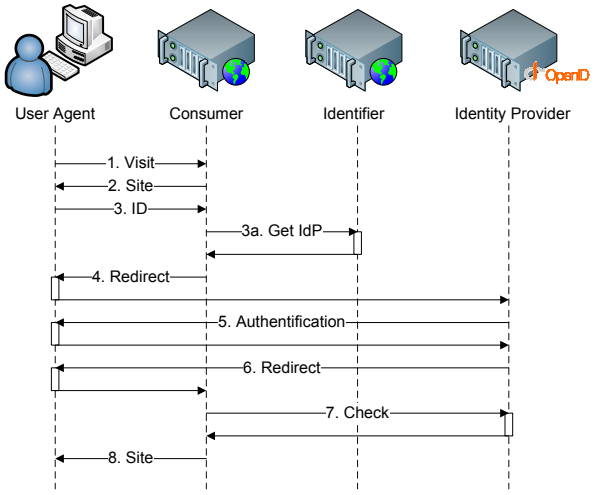


Abbildung 7. Kommunikationsablauf im Dumb mode.

War der Authentifizierungsvorgang erfolgreich, so ist der Benutzer mit seiner OpenID Identität auf der Website angemeldet und kann die gewünschten Dienste nutzen.

F. Implementierung

Wie bereits anfangs erwähnt, steht es jedem interessierten und technisch versierten Anwender frei, seinen eigenen Identity Provider aufzusetzen. Entsprechende Open Source Pakete liegen sowohl für die Server- als auch die Consumer-Variante in nahezu allen gängigen Programmiersprachen vor (z.B. C#, C++, Java, Perl, Python, PHP). Damit steht einer Umsetzung auf einem Großteil der Server nichts mehr im Wege, da diese meist PHP, Perl oder Python als cgi-Scripts unterstützen und fast jeder Client heutzutage Java ausführen kann. [10] Zur Implementierung gibt es z.B. in PHP das komplett fertige Paket „phpMyID“. Die darin enthaltenen PHP-Dateien müssen im Verzeichnis der Homepage abgelegt werden. Wurden diese angepasst, muss der Besitzer der Website die URL zu der PHP-Datei nur noch in die beiden bereits im obigen Abschnitt „Identifier und Identity Provider“ (IV. D.) gezeigten Zeilen im Quellcode der Startseite einzutragen. Danach ist der eigene IdP einsatzfähig. Die Dateien inklusive Installationsanleitung und die genauen Voraussetzungen befinden sich direkt auf der Homepage von phpMyID [17].

G. Sicherheit

Geht es um Authentifizierungsmechnismen so stellt sich auch immer die Frage nach deren Sicherheit. Wenn jeder als

Provider agieren darf, wie sicher kann das Verfahren dann sein? Prinzipiell kann gesagt werden, dass das Verfahren nur so sicher wie die hostende Website bzw. deren Quellcode selbst ist. Damit liegt es in den Händen des Betreibers. Lässt sich die Seite aufgrund unsicherer Skripte o.ä. manipulieren oder fremde Codesegmente importieren, so kann auch hier keine Garantie über die wirkliche Authentizität des sich anmeldenden Benutzers gegeben werden. Denn ist es möglich die Website zu manipulieren, so kann man auch die gesamte Kommunikation zu einem anderen IdP umleiten oder einen solchen betreiben. [11] Die Technologie von OpenID ist demnach gegenüber *Phishing-Attacken* anfällig. Der Grund hierfür ist die Tatsache, dass eine Weiterleitung auf die Seite des OpenID Providers nötig ist. „Als Betreiber einer Seite, welche OpenID zur Anmeldung benutzt, kann man auf einfache Weise eine Weiterleitung auf eine Seite erstellen, welche der Providerseite gleicht, jedoch als Proxy dient und Benutzername und Passwort an den Betreiber weiterleitet. Die OpenID Provider können dies umgehen, indem sie z.B. *Cookies* setzen, die ein individuelles Bild zeigen oder indem sie ein clientseitiges *TLS-Zertifikat* zur Authentifizierung nutzen. Insbesondere letzteres wird von vielen Providern unterstützt.“ [5] Im Internet finden sich Seiten auf denen sich ein solches *Phishing* in Form einer Demo mitverfolgen lässt.
Kann ein Angreifer eine Identität fälschen oder sich gegebenenfalls sogar einer fremden bemächigen, sprich lässt sich die Kommunikation zwischen dem Consumer und dem IdP abhören und die dabei erhaltenen Informationen speichern, so könnte ein Angreifer, versuchen eine der vom Identity Provider an den Consumer gesendeten Bestätigungen über eine erfolgreiche Authentifizierung zu imitieren. Auch das Abfangen einer Nachricht, um diese später als Bestätigung zur Anmeldung mit einem falschen Namen zu verwenden, oder evtl. kritische Benutzerdaten aus der Kommunikation zwischen Consumer und Identity Provider zu gewinnen, wären mögliche Szenarien. Hiergegen ist OpenID jedoch gut gerüstet. Wie bereits zuvor erwähnt „nutzt OpenID TLS, um die Verbindung abzusichern und fügt jeder Abfrage zusätzlich eine Challenge bei“ [11], wodurch jede Antwort jeweils nur einmal Gültigkeit besitzt und nicht wieder verwendet werden kann. [11] Letztendlich sind diese Sicherheitsbedenken zwar zu beachten, jedoch ist nicht OpenID das Problem, sondern eher wie es eingesetzt wird. Ein Customer wird standardmäßig allen IdPs vertrauen mit denen er Kontakt aufnimmt. Dies ist für die meisten Anwendungen die OpenID als Authentifizierungsmethode benutzen durchaus akzeptabel, für Anwendungen mit höheren Sicherheitsanforderungen jedoch nicht. Hierfür lässt sich der Consumer so konfigurieren, dass er nur einigen wenigen, großen und gut abgesicherten IdPs, wie z.B. Verisign, vertraut.

H. Bewertung

Grundsätzlich bleibt zu sagen, dass OpenID mit seinem Ansatz und seiner Funktionsweise in die richtige Richtung geht, da es, bedingt durch seine Funktionen, dem Anwender das Leben erheblich erleichtern kann. Im Endeffekt sind IdM Systeme eine gute Sache, die den Anwendern auch viel ihrer Privatsphäre und vor allem die Kontrolle über die eigenen Daten zurück geben und ihnen einen gewissen Komfort im Umgang mit ihrer Identität im Internet zur Verfügung stellen. Dabei hat der Consumer zwei Möglichkeiten. Entweder er verzichtet auf ein gewisses Maß an Sicherheit, vertraut jedem IdP und gewährt dem Nutzer die größtmögliche Freiheit bei der Wahl seines IdP, oder er legt erhöhten Wert auf Sicherheit, vertraut nur ausgewählten IdP und schränkt den Anwender dadurch wiederum in seiner Freiheit an. Denkbar wäre in diesem Zusammenhang, um eine maximale Sicherheit bei den IdP zu erlangen, dass die Erstellung eines Accounts bei den IdPs, ähnlich einer Eröffnung eines Kontos bei einer Online-Bank, ausschließlich durch erfolgreiche Verifizierung mittels PostIdent möglich ist. In diesem Fall wäre es möglich, dass die IdPs für die Validität der persönlichen Daten Ihrer Kunden bürgen. Es hängt also immer von der Art der Anwendung ab für die OpenID als Authentifizierungsmechanismus dienen soll, inwiefern der eigentliche Grundgedanke von Fitzpatrick umgesetzt werden kann.
Fest steht jedoch leider, dass die Vorteile von OpenID erst richtig zum tragen kommen, sobald eine flächendeckende Lösung gefunden ist. Die Zusammenarbeit vieler unterschiedlicher Unternehmen und besonders technologischen Wegbereitern wie Sun, IBM, Intel uvm. an der Weiterentwicklung der Interoperabilität und die Bemühungen andere Techniken in eigene Lösungen zu integrieren, ist definitiv ein Anfang. Solange dem Benutzer jedoch keine einheitliche Lösung präsentiert werden kann und er sich lediglich zusätzlich zu seinen bisher zu merkenden Logins und Passwörtern auch noch seine OpenID und deren Passwort merken muss, steht OpenID bzw. IdM noch ein weiter Weg bevor. Denn eines steht fest, „am Ende muss der Anwender selbst den Überblick behalten - auch ein Identity Management System kann ihn dabei nur unterstützen.“ [11]

V. Zusammenfassung und Ausblick

Identity Management bringt, wie beschrieben, viele Vor- aber auch einige Nachteile mit sich. So braucht sich der Anwender, bedingt durch die SSO Funktionalität, nicht wie zuvor für jede Website und Webapplikation einen Benutzernamen mit zugehörigem Passwort merken, sondern hat durch die einmalige Authentifizierung beim IdP ein deutlich erhöhtes Maß an Komfort. Zusätzlich hat er durch die Verwendung von Attributen, die er selbst verwalten kann, die Möglichkeit eine genaue Kontrolle darüber zu haben, welchem Anbieter er welche personenbezogenen Daten zur Verfügung stellen möchte und kann dies zentral verwalten.
Betrachtet man die in puncto Sicherheit bereits erwähnten potenziellen Schwachstellen, so sind diese nicht auf die Arbeitsweise oder Technik des OpenID-Protokolls zurückzuführen. Grund sind mangelnde Absicherung oder die falsche Konfiguration durch die Betreiber der Consumer-Websites, aber auch die Unaufmerksamkeit der Benutzer. [10] „Für Anwendungen und Dienste die nur eine vergleichweise wenig komplexe Identitfizierung erfordern, wie beispielsweise Blogs u.ä., erfreut

sich OpenID zwar steigender Beliebtheit. In elektronische Geschäftsprozesse jedoch hat OpenID bisher keinen Einzug gefunden.“ [10]

Das größte Handicap von OpenID, ist die zwar wachsende aber dennoch recht geringe Verbreitung unter den Anbietern und die damit einhergehende noch mäßige Popularität unter den Benutzern. Hinzu kommt ein Problem, welches auf IdM Systeme im Allgemeinen zu beziehen ist, nämlich die mangelnde Interoperabilität der verschiedenen Lösungen untereinander, was auch gerade im Hinblick auf die teils sehr großen Unterscheide im Leistungsspektrum der verschiedenen Lösungen zurückzuführen ist. Wenngleich OpenID immer mehr Unterstützung findet und die Zahl der Anwendungen, die OpenID nutzen, steigt, müssen dennoch „einige richtig große Anwendungen erst noch zeigen, ob sie allen operativen und konzeptionellen Anforderungen an Vertraulichkeit und auch Verfügbarkeit gewachsen sind.“ [11] So gibt es zeitweise immer noch Ausfälle bei OpenID Providern, während derer die Endnutzer ihre virtuellen Identitäten nicht nutzen können.

Die Zukunft wird zeigen ob die unterschiedlichen Ansätze, die von den verschiedenen Vertretern aus der Industrie unterstützt werden, sich dauerhaft etablieren können. Ob OpenID sich nach und nach bei allen Betreibern von Websites mit Authentifizierungsmechanismen durchsetzt und vielleicht auch Einzug in die Unternehmenswelt halten wird, bleibt abzuwarten.

LITERATUR

[1] A. Myllyniemi, “Identity management systems - a comparison of current solutions,” Helsinki University of Technology, Tech. Rep., 2006. [Online]. Available: http://www.tml.tkk.fi/Publications/C/22/papers/Myllyniemi_final.pdf

[2] Wikipedia, “Identitätsmanagement,” 2008. [Online]. Available: http://de.wikipedia.org/wiki/Identitätsmanagement

[3] ——, “Single sign-on.” [Online]. Available: http://de.wikipedia.org/wiki/Single_Sign-On

[4] D. Recordon and D. Reed, “Openid 2.0: A platform for user-centric identity management,” ACM - Association for Computing Machinery, Tech. Rep., 2006. [Online]. Available: http://portal.acm.org/citation.cfm?id=1179532

[5] Wikipedia, “Openid.” [Online]. Available: http://de.wikipedia.org/wiki/OpenID

[6] OpenID. [Online]. Available: http://openid.net

[7] Shibboleth. [Online]. Available: http://shibboleth.internet2.edu/

[8] L. Alliance. [Online]. Available: http://www.projectliberty.org/

[9] V. Smith, “Openid vs. shibboleth: power to the people.” [Online]. Available: http://vsmith.info/OpenID

[10] C. J. Ruwe, “Openid - management/authentifikation digitaler identitäten im web,” Helmut Schmidt Universität - Universität der Bundeswehr Hamburg, Tech. Rep., 2007. [Online]. Available: http://academics.cruwe.de/files/openid.pdf

[11] N. Magnus, “Identitätsverwaltung im web mit open id - offene identität,” *Linux-Magazin*, vol. 07, pp. 50–53, 2008.

[12] R. U. Rehmann, *The OpenID Book - A comprehensive guide to OpenID protocol and running OpenID enabled websites*. Conformatix Technologies Inc., 2008. [Online]. Available: http://openidbook.com/

[13] OpenID.net, “Openid authentication 2.0 - final,” 2007. [Online]. Available: http://openid.net/specs/openid-authentication-2_0.html

[14] D. Recordon and B. Fitzpatrick, “Openid authentication 1.1,” 2006. [Online]. Available: http://openid.net/specs/openid-authentication-1_1.html

[15] Wikipedia, “Yadis.” [Online]. Available: http://de.wikipedia.org/wiki/Yadis

[16] O. Wiki. [Online]. Available: http://wiki.openid.net

[17] phpMyID. [Online]. Available: http://siege.org/projects/phpMyID/

Identity Federation und Web Services

Karim Djelassi
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
karim@djelassi.com

ABSTRACT
Diese Ausarbeitung gibt einen Überblick über Identity Federation und Web Services und erklärt was sich hinter diesen Begriffen jeweils verbirgt. Im Anschluss werden aktuelle Identity Federation Projekte aufgezählt und klassifiziert. Die Identity Federation Projekte OASIS und Liberty Alliance werden genauer behandelt und im Detail vorgestellt. Zur Veranschaulichung wird ein Anwendungsbeispiel für Identity Federation präsentiert.

SCHLÜSSELWÖRTER
Identity federation, web service, identity federation management, digital identity, federated identity, authentication, information system, Liberty Alliance, Shibboleth, OpenID, OASIS, SAML, WSDL, SOAP

1. EINLEITUNG
Die Welt, in der wir leben, wird von einer steigenden Anzahl an Informationssystemen und Informationsservices unterstützt und ermöglicht. Dieses als „align and enable“ bekannte Prinzip beschreibt das Zusammenspiel von Informationstechnologie und Gesellschaft im Informationszeitalter [1]. Viele Menschen interagieren regelmäßig mit einer Vielzahl von unterschiedlichen Informationssystemen. Dabei kann es sich beispielsweise um Webmail-, Fotoalben-, Shopping- oder Blogging-Portale handeln. Ebenso machen webbasierte Business Applikationen einen großen Teil der regelmäßig verwendeten Informationssysteme aus.

So gut wie jede Interaktion mit einem Informationssystem erfordert eine Authentifizierung des Benutzers. Um sich gegenüber einem Informationssystem authentifizieren zu können, muss sich der Benutzer mit einem Benutzerkonto am System anmelden. Die Authentifizierung geschieht in der Regel mittels Eingabe von Benutzername und Passwort. Da der Benutzer vom System über sein Benutzerkonto identifiziert wird, spricht man auch von einer *Digital Identity* [2].

Da die meisten Menschen mit mehreren Informationssystemen interagieren, haben sie entsprechend auch mehrere Digital Identities. Aus Sicherheitsgründen sollten sich Digital Identities eines Benutzers im Bezug auf Benutzername und Passwort unterscheiden. Dennoch enthalten sie ähnliche persönliche Informationen, wie z.B. die Email-Adresse oder evtl. die Anschrift des Benutzers. Das Resultat ist eine schnell unüberschaubare Anzahl von Digital Identities, für die ein Benutzer unterschiedliche Anmeldeinformationen bereit halten muss, obwohl sie fast dieselben persönlichen Informationen enthalten.

Die aggregierte Anzahl an Digital Identities aller Benutzer in allen Informationssystemen ist dementsprechend enorm. Der Betreiber eines Informationssystems verwaltet nur die Digital Identities der Benutzer seines Systems. Sobald jedoch Informationssysteme in der Lage sein müssen sich mit anderen Informationssystemen auszutauschen müssen Digital Identities abgeglichen und synchronisiert werden können. Aufgrund der Vielfältigkeit der heute existierenden Informationssystem-Landschaft wird für jede Verbindung zweier Informationssysteme eine individuell maßgeschneiderte Softwarelösung benötigt. Diese Softwarelösungen sind in aller Regel sowohl zeit- als auch kostenintensiv [3].

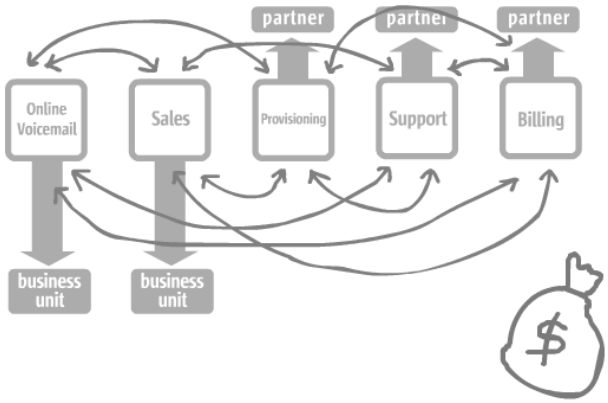


Abb. 1: Veranschaulichung der Komplexität von Informationssystem-Verbindungen [3]

Im Folgenden wird erläutert inwiefern Identity Federation und Web Services dazu beitragen können die mit Abb.1 symbolisch veranschaulichte Komplexität von Informationssystem-Verbindungen zu reduzieren.

1.1 Identity Federation
Identity Federation, auch bekannt unter der Formulierung *Federated Identity Management*, ist ein generisches Konzept, das die Angleichung und Vereinigung von Digital Identities und den in ihnen enthaltenen persönlichen Informationen beschreibt. Praktisch soll ein Benutzer nicht mehr eine Digital Identity pro Informationssystem besitzen, sondern nur noch auf wenige System-übergreifende Digital Identities angewiesen sein. Da Digital Identities durch den Prozess der Identity Federation vereinigt werden, spricht man hier auch von *Federated Identities*. Um Identity Federation global und einheitlich zu ermöglichen werden offene Standards und Spezifikationen entwickelt und benutzt [4].

Zur Umsetzung dieses Konzepts müssen sich Informationssystem-Betreiber im Bezug auf die Authentizität ihrer individuell gesammelten Digital Identities gegenseitig vertrauen. Gleichzeitig verständigt man sich darauf, die einmalige Validierung eines Benutzers mit deiner Digital Identity unterschiedlich zu handhaben. Die Informationssystem-Betreiber gehen ein Verhältnis des gegenseitigen Vertrauens ein und vereinigen ihre Digital Identities zu Federated Identities [3]. So würden sich z.B. Universitäten darauf verständigen ihre Studenten auf unterschiedliche Art und Weise zu immatrikulieren und gleichzeitig die Authentizität eines an einer vertrauten Universität immatrikulierten Studenten akzeptieren.

Damit stellt das Konzept der Identity Federation eine vielversprechende Lösung für eine breit gefächerte Menge an Problemen dar. So können durch den richtigen Einsatz von Identity Federation Lösungen unter anderem Probleme mit der

Skalierbarkeit von Informationssystem-Verbindungen eliminiert und Produktionskosten für entsprechende Software reduziert werden. Ebenso lässt sich durch den Einsatz solcher Lösungen der Datenschutz für Benutzer erhöhen und die Benutzerfreundlichkeit umfassend verbessern [3].

1.2 Web Services

Die Entwicklung von Computern begann vor ungefähr 60 Jahren im zweiten Weltkrieg. Die damaligen Computer wurden ausschließlich für militärische Zwecke entworfen und gebaut. Eine der charakteristischsten Eigenschaften dieser Computer war, dass sie die Berechnung der ihnen gestellten Aufgaben alleine bewerkstelligen mussten. Die Evolution der isoliert arbeitenden Computer setzte sich zunächst fort und wurde letztendlich in den Siebzigerjahren durch die Entwicklung der Personal Computer (PC) beendet. Die für die damalige Zeit unglaublich leistungsfähigen Computer wurden verstärkt an immer mehr Arbeitsplätzen eingesetzt. Durch die Ausstattung vieler Mitarbeiter einer Organisation mit einem PC wurden Daten verstärkt lokal abgespeichert und Informationsstrukturen wandelten sich zu tendenziell dezentraleren Schemata [5].

An diesem Punkt der Entwicklung bekamen Netzwerke eine entscheidende Bedeutung für den Informationsaustausch – die Entwicklung der ersten verteilten Systeme begann. Diese waren innerhalb kürzester Zeit in der Lage Funktionalität für die Computer eines Netzwerkes bereitzustellen. Die Netzwerke beschränken sich jedoch in der Regel auf eine Organisation. Dies änderte sich jedoch schlagartig mit der Einführung und der rapide ansteigenden Nutzung des Internets [5].

Verteilte Systeme bekamen durch diese Entwicklung noch eine viel bedeutendere Rolle zugesprochen – eine Entwicklung, die bis heute anhält und weiterhin die Bedeutsamkeit von verteilten Systemen steigen lässt. Ebenso stiegen jedoch die Anforderungen an verteilte Systeme: Netzwerkstrukturen wuchsen, Informationsmengen nahmen zu und Informationsstrukturen wurden komplexer. Durch die Verwicklung von immer mehr Organisationen und die globale Verteilung von Informationen waren eine steigende Inkohärenz und eine daraus resultierende Inkonsistenz der Informationen unvermeidlich [5].

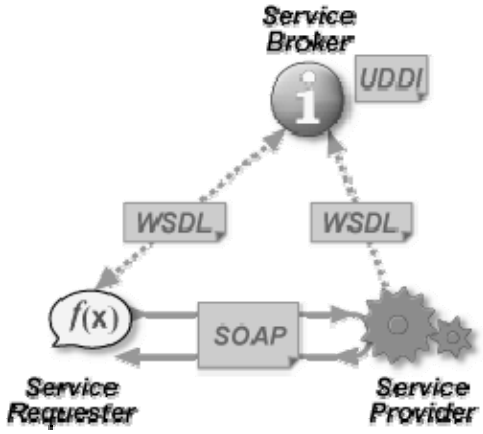


Abb. 2: Web Services Protokoll Architektur [6]

Insbesondere die konstant ansteigende Anzahl von Organisationen mit denen Informationsaustausch betrieben wurde, sorgte für eine ebenso konstant ansteigende Kostenkurve für Informationsverarbeitung und eine Vielzahl von Softwarelösungen, die aufgrund mangelnder Übertragbarkeit kaum wiederverwendbar waren [3]. Dieser Umstand machte die Entwicklung von standardisierten und somit kosteneffizienten Wegen des Informationsaustauschs dringend notwendig: *Web Services* wurden eingeführt. Diese Technologie ermöglicht es verteilten Systemen miteinander zu kommunizieren und vollautomatisch Informationen auszutauschen.

Aktuell existieren zwei maßgebliche Web Service Plattformen: J2EE von Sun Microsystems und .NET von Microsoft. Beide Plattformen beherbergen mehrere Web Service Frameworks. Obwohl die unterschiedlichen Frameworks auf unterschiedlichen Plattformen beruhen besteht dennoch Kompatibilität durch die Verwendung und Einhaltung von gemeinsamen und offenen Protokollen.

Wie in Abb.2 dargestellt handelt es sich bei den essentiellen Protokollen um die *Web Service Description Language (WSDL)* und das *Simple Object Access Protocol (SOAP)*. Wie der Name vermuten lässt wird WSDL für die Beschreibung von Web Services verwendet. Mit Hilfe einer WSDL Beschreibung ist es einem Benutzer oder einem Computer möglich festzustellen wo ein spezieller Web Service zu finden ist und in welcher Art und Weise die Interaktion mit diesem erfolgen muss. Im Wiederspruch zu seinem Namen handelt es sich bei SOAP nicht mehr um ein einfaches Protokoll. Mit der Zeit wurde es zu einem komplexen und sehr vielseitigen Protokoll, welches zum Versenden und Empfangen von Web Service Nachrichten verwendet wird. Die an einen Web Service versendeten Nachrichten enthalten in der Regel Meta-Informationen, welche die vom Benutzer benötigten Informationen beschreiben. Die im Gegenzug von einem Web Service versendeten und vom Benutzer empfangenen Nachrichten enthalten in der Regel die angefragten Informationen. Beide Protokolle, WSDL und SOAP, basieren auf XML [6].

Mit Bezug auf Identity Federation stellen Web Services die nötigen Plattformen und Frameworks bereit, in deren Umgebung die Entwicklung und der Einsatz von Identity Federation Systemen ermöglicht werden. Es ist von entscheidender Bedeutung, dass alle verwendeten Protokolle frei verfügbar sind und offenen Standards genügen damit uneingeschränkte Kompatibilität zu jeder Zeit garantiert werden kann.

2. PROJEKTE

Mittlerweile existiert eine beträchtliche Anzahl an Identity Federation Projekten, deren Entwicklungsstände sich teilweise der Markreife nähern. Einige wenige Projekte und eine Auswahl von am jeweiligen Projekt beteiligten Unternehmen haben bereits damit begonnen Identity Federation in Form von Pilotprojekten aktiv einzusetzen. Nahezu marktreife Identity Federation Projekte zeichnen sich unter anderem dadurch aus, dass neben Standards und Spezifikationen auch fertig implementierte Software erhältlich ist. Diese Basissoftware implementiert den Teil der Standards und Spezifikationen, der für alle Verwender identisch ist. So können sie von Unternehmen bei minimalem Zeit- und Kostenaufwand aufgegriffen, erweitert und an die eigenen Bedürfnisse angepasst werden [7].

Zu den am mit weitesten fortgeschrittenen Organisationen zählen unter anderem *Liberty Alliance*, *Internet2* und die *Organization for the Advancement of Structured Information Standards (OASIS)* mit ihren Projektgruppen *Liberty Federation*, *Shibboleth* und dem *Security Services Technical Committee (SSTC)*. Speziell diese drei Projektgruppen konnten die Entwicklung ihrer Identity Federation Projekte durch enge Kooperation schnell voran treiben und bilden mittlerweile die Speerspitze auf dem Gebiet der Identity Federation [8].

Bereits kurz nach dem Start von Shibboleth im Jahr 2000, dicht gefolgt von Liberty Federation und SSTC im Jahr 2001, begann die Kooperation zwischen den Projektgruppen. Wie in Abb. 3 erkennbar fand der erste Austausch zwischen Liberty Federation und SSTC nach knapp 22 Monaten Projektarbeit im November 2002 mit dem Release des *SAML V1.0* Standard statt. Die Veröffentlichung dieses Standards nahm Einfluss auf die in der

frühen *Phase 1* befindliche Liberty Federation. Im Januar 2003 konnte die erste Liberty Federation Spezifikation veröffentlicht werden: *ID-FF V1.1*. Diese wurde dicht gefolgt von den im Juli bzw. August 2003 veröffentlichten ersten Shibboleth Spezifikationen: *Shib 1.0 bzw. 1.1*. Beide Spezifikationen basierten auf dem SAML V1.0 Standard.

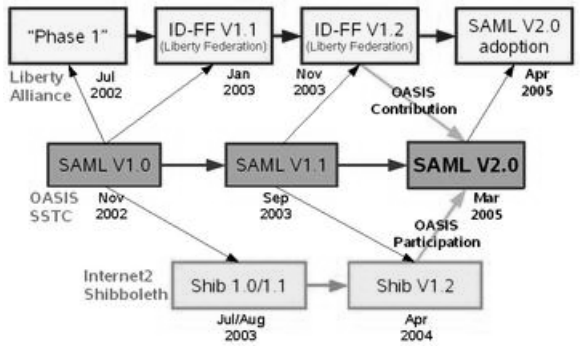


Abb. 3: Kooperation zwischen Liberty Alliance, Shibboleth und OASIS [15]

Basierend auf der im September 2003 folgenden SAML V1.1 Spezifikation wurden die Spezifikationen *ID-FF V1.2* und *Shib 1.2* entwickelt. Die elementaren Aspekte beider Spezifikationen konnten in den im März 2005 veröffentlichten *SAML V2.0* eingebracht werden. Die aktuelle Version der Shibboleth Spezifikation ist die im März 2008 veröffentlichte *Shib V2.0*, welche [9]. Für Liberty Federation stellt die ID-FF V1.2 Spezifikation seit 2003 weiterhin den aktuellen Stand dar. Dieser wurde allerdings um eine ganze Reihe von zusätzlichen Spezifikationen ergänzt, wodurch die Abdeckung verschiedenster Bedürfnisse unterschiedlicher Branchen im Bezug auf Identity Federation gewährleistet wird [10].

SAML V2.0 stellt bis zum heutigen Tag die unumstrittene Basis für Identity Federation Projekte und Spezifikationen dar und wurde von einer Vielzahl von Identity Federation Projektgruppen aufgegriffen. Welche Rolle SAML V2.0 spielt und in welchem Zusammenhang es mit Liberty Federation steht im wird im Folgenden genauer erläutert.

3. OASIS

OASIS ist ein gemeinnütziges Konsortium, das die Entwicklung und Verbreitung von offenen Standards für die globale Informationsgesellschaft voran treibt. Das Konsortium wurde im Jahr 1993 von einer kleinen Anzahl von Unternehmen gegründet, die es sich zum Ziel gesetzt hatten die Kompatibilität zwischen ihren Produkten zu gewährleisten. Mittlerweile besteht das Konsortium aus über 5.000 Mitgliedern, die über 600 Unternehmen aus unterschiedlichen Ländern der ganzen Welt repräsentieren [11].

OASIS fordert aktiv dazu auf sich an der Mitgestaltung von offenen Standards zu beteiligen. Mitglied werden kann jeder, vom Selbständigen über kleine Unternehmen bis hin zum internationalen Konzern. Der jährliche Mitgliedsbeitrag liegt je nach Art der Mitgliedschaft und Unternehmensform des Mitglieds bei 300 bis 50.000 Dollar [12].

Strukturell gliedert sich OASIS in das *Board of Directors*, das *Tech Advisory Board* und diverse *Technical Committees*. Die Boards werden im Zwei-Jahres Intervall mit Mitgliedern besetzt, welche wiederum von Mitgliedern gewählt wurden. Die Teilnahme an beliebigen Technical Committees steht jedem Mitglied frei [11].

3.1 SSTC

Das *Security Services Technical Committee (SSTC)* ist eines der angesprochenen Technical Committees. Es wurde im November 2000 ins Leben gerufen, worauf hin im Januar 2001 das erste Treffen der SSTC-Mitglieder stattfand und die Aufgabe des Komitees definiert wurde. Aufgabe des SSTC ist es ein auf XML basierendes Framework, welches die Erstellung und den Austausch von Authentifizierungs- und Autorisations-Informationen ermöglicht, zu definieren, zu erweitern und zu pflegen. Zentrales Objekt dieser Bemühungen ist seit jeher die Entwicklung des SAML Standards. [13]

3.2 SAML

Die *Security Assertion Markup Language (SAML)*, welche aktuell in der Version 2.0 vorliegt, ist dazu bestimmt die Erstellung und den Austausch von Authentifizierungs- und Autorisations-Informationen zu ermöglichen.

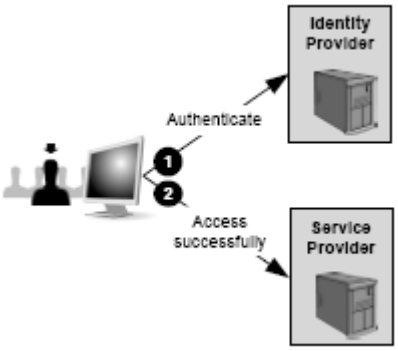


Abb. 4: Single Sign-On Authentifizierungsprozess, ausgelöst vom Identity Provider [16]

In Anspielung auf das bekannte Sprichwort „think globally, act locally“ wird dieses im Sinne der Identity Federation genau umgekehrt zu „think locally, act globally“ [14]. Dieses lässt sich durch das für den Endbenutzer wichtigste Feature der Identity Federation, den Single Sign-On, näher erläutern. Wie in Abb.4 ersichtlich authentifiziert sich der Benutzer bei seinem Identity Provider, dem Anbieter, der die Digital Identity des Benutzers verwaltet, dem der Benutzer nahe steht und Vertrauen schenkt – „think locally“. Mithilfe dieser Authentifizierung ist der Benutzer in der Lage auf Service Provider zuzugreifen. Bei Service Providern handelt es sich um Anbieter eines speziellen Angebots, jedoch muss für den Zugriff zunächst die Authentizität des Benutzers gewährleistet werden. Mittels der vom Identity Provider bestätigten Authentizität des Benutzers kann dieser wiederum auf die Angebote verschiedenster Service Provider weltweit zugreifen – „think globally“.

Eine weitere Art der Realisierung des Single Sign-On wird in Abb.5 dargestellt. Im Falle des Nichtvorhandenseins der Authentifizierung des Benutzers wird dieser an seinen Identity Provider weitergeleitet. Dort muss sich der Benutzer zunächst authentifizieren und kann im Anschluss zum Angebot des Service Providers zurückkehren [16].

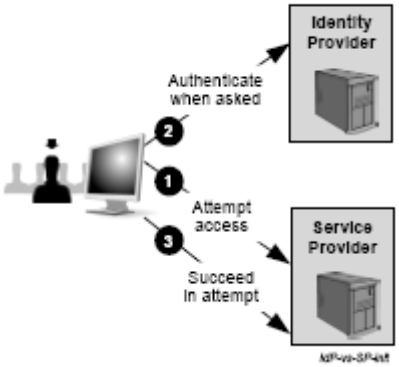


Abb. 5: Single Sign-On Authentifizierungsprozess, ausgelöst vom Service Provider [16]

Zu den Stärken von SAML zählt vor allem die starke Anpassbarkeit bei gleichzeitiger Wahrung der Kompatibilität zum eigenen Standard. Auf diese Weise ist es SAML möglich den unterschiedlichsten Anforderungen von Unternehmen zu Rechnung zu tragen und somit als unbestrittener Basisstandard für Identity Federation zu gelten. Ebenfalls Resultat des sehr allgemein gehaltenen Standards ist die Plattformunabhängigkeit, was wiederum zur weiteren Verbreitung und Adaption von SAML beiträgt. Aus Sicht eines Unternehmens betrachtet bietet SAML sehr großes Potential für Zeit- und Kostenersparnisse – Kosten die sonst durch die Entwicklung von komplett individuell erstellter Software verursacht worden wären. Aus Sicht des Endbenutzers ist die stark erhöhte Benutzerfreundlichkeit durch den Single Sign-On direkt spürbar. Die auf den ersten Blick nicht ganz offensichtliche Verbesserung von Sicherheit und Privatsphäre für den Endbenutzer durch insgesamt weniger und gleichzeitig leichter kontrollierbare Federated Identities zählt ebenfalls zu den zentralen Stärken von SAML [14].

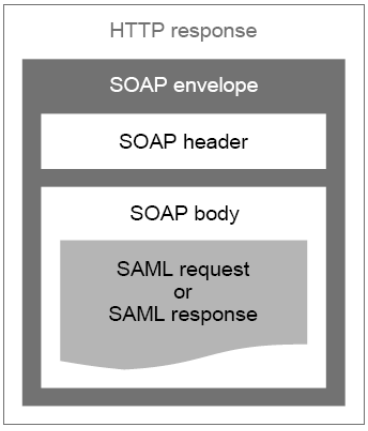


Abb. 6: Strukturelle Einbettung einer SAML Nachricht in SOAP und http [16]

Ebenso gliedert sich SAML in die bestehende Landschaft aus Standards und Spezifikationen nahtlos ein. Abb.6 zeigt eine in eine SOAP Nachricht eingebettete SAML Nachricht. SOAP wird im Zusammenhang mit Web Services verwendet. Es bietet sich an SAML und Identity Federation insgesamt in Verbindung mit Web Services zu nutzen. Sowohl Identity Federation mit SAML als auch Web Services mit SOAP sind stark durch Standards und Spezifikationen geprägte Technologien. Auf diese Weise können vollständig automatisierte Prozesse über System- und Unternehmensgrenzen hinweg auf einer gemeinsamen Basis existieren und funktionieren [16].

4. LIBERTY ALLIANCE

Die Liberty Alliance Projektgruppe arbeitet an der Entwicklung von Identity Federation Spezifikationen, Protokollen und Applikationen. Die leitende Vision der Liberty Alliance ist eine auf offenen Standards basierende und vernetzte Welt zu ermöglichen in der Kunden, Bürger, Unternehmen und Regierungen Onlinetransaktionen leichter als bisher ausführen können wobei die Privatsphäre und die Sicherheit von persönlichen Informationen dauerhaft geschützt bleibt [17]. Diese Vision soll dadurch erreicht werden, dass alle erarbeiteten Spezifikationen, Protokolle und Applikationen frei verfügbar gemacht werden und von jedem, der Liberty Federation für verwenden möchte, aufgegriffen und adaptiert werden können.

4.1 Projektmitglieder

Die Liberty Alliance wurde im Jahr 2001 von 30 führenden Organisationen aus den Bereichen IT, Finanzwesen, Telekommunikation, Medien, Produktion, Regierung und Bildung gegründet. Die Organisationen entstammen unterschiedlichen Teilen dieses Planeten und bieten so eine gute Repräsentation verschiedener Kulturen. Insbesondere die umfassende Repräsentation unterschiedlicher Kulturen im Projekt ist ein wichtiger Aspekt bei Entwicklung von global anwendbaren Standards für Identity Federation. Die aktuelle Anzahl der Liberty Alliance Mitglieder beläuft sich mittlerweile auf 150 bedeutende Organisationen [18].

Strukturell sind die Mitglieder in mehrere Stufen unterteilt, welche jeweils einen anderen Grad der Beteiligung aber auch des Einflusses repräsentieren. An der Spitze der Hierarchie stehen die Organisationen, die dem Management Board angehören. Beispiele für diese Mitgliedschaftsstufe sind AOL, Intel, Sun und Oracle. Dennoch sind auch bei Mitgliedschaftsstufen mit weniger Beteiligung und Einfluss bedeutende Unternehmen wie z.B. IBM, HP, Nokia, Adobe und Paypal zu finden [19].

4.2 Liberty Federation

Die Gründung der Liberty Alliance fand zunächst ausschließlich vor dem Hintergrund der gemeinsamen Entwicklung von Identity Federation Spezifikationen statt. Das erste und zunächst einzige Projekt war somit die Liberty Federation.

Mit dem Fortschritt der Liberty Federation ergaben sich Ansatzpunkte für Erweiterungen und weitere Aspekte die näherer Betrachtung bedurften. Es kam zur Gründung diverser Strategic Initiatives innerhalb der Liberty Alliance, welche alle auf der Liberty Federation basierten [20].

Ein weiteres Projekt, das sich umfassend auf alle Strategic Initiatives bezieht, ist die Liberty Interoperable. Im Rahmen dieses Projekts werden Testszenarien für Liberty Federation entwickelt und angewendet. Aufgabe ist es Unternehmen, die einen Test ihrer Liberty Federation Lösung wünschen zu prüfen um die Kompatibilität mit den Standards und Spezifikationen zu garantieren. Bei einem erfolgreich absolvierten Test erhält das Unternehmen ein Zertifikat, das die Standard-Konformität dokumentiert. Sollte ein Unternehmen mit seiner Liberty Federation Lösung den Test nicht bestehen werden detailierte Hinweise und Testergebnisse bereitgestellt um Schwächen und Fehler der Implementierung zu beseitigen [21].

Mit diesem Testprogramm ist die Liberty Alliance bisher alleiniger Vorreiter und macht einen wichtigen Schritt in Richtung Vertrauensbildung zwischen Unternehmen. Auf diese Weise können Unternehmen, die den Test erfolgreich absolviert haben und das Zertifikat vorweisen können, davon ausgehen dass zumindest auf Ebene der unterschiedlichen Implementierungen

und im Bezug auf deren Kompatibilität die Wahrscheinlichkeit für Probleme sehr gering ist.

4.3 Liberty Web Services

Bei den Liberty Web Services handelt es sich um eine Strategic Initiative der Liberty Alliance. Basierend auf der ID-FF V1.2 Spezifikation der Liberty Federation und dem SAML V2.0 Standard existieren die Spezifikationen *Identity Web Services Framework ID-WSF V2.0* und *Identity Service Interface Specifications ID-SIS V1.0* im Liberty Web Services Projekt. Zusammen ermöglichen beide Technologien die Entwicklung von Identity-sensitiven Web Services [22].

Sinn und Zweck von Identity-sensitiven Web Services ist die Ermöglichung völlig neuer und stark automatisierter Anwendungen im Web. Durch die Liberty Web Services werden Wege eröffnet, die die Vergabe von Zugriffsrechten auf persönliche Informationen eines Benutzers ermöglichen. Ziel ist es den automatischen Informationsaustausch zwischen Service Providern zu realisieren um so das Service Angebot für den Benutzer zu bereichern. Die Herausforderung ist zunächst die Auffindung von Informationen mittels eines *Discovery Service (DS)* überhaupt erst zu ermöglichen und im Anschluss einen standardisierten Informationsaustausch zu gewährleisten [23].

In diesem Zusammenhang wird der Informationen zur Verfügung stellende Service Provider als *Web Service Provider (WSP)* bezeichnet. Der Informationen konsumierende Service Provider wird entsprechend als *Web Service Consumer bezeichnet (WSC)*. Abb.7 zeigt in diesem Zusammenhang die abstrahierte Liberty Federation im Vergleich mit den detailierten Liberty Web Services. *Authentication Service (AS)*, *Single Sign-On Service (SSOS)*, *Identity Mapping Service (IMS)* und Discovery Service (DS) stellen die detailieren Funktionalitäten des abstrahierten *Identity Providers (IdP)* dar [23].

Ein beispielhaftes Szenario ist die vollautomatische Buchung eines Bahntickets. Statt einer klassischen Online-Buchung durch den Benutzer, für die eine manuelle Auswertung von Terminkalender, möglichen Zugverbindungen, entsprechenden Preisen und eventuellen Sonderangeboten nötig wäre, soll die Buchung vollständig vom Buchungssystem erledigt werden. Um das Zeitfenster für die Zugfahrt feststellen zu können benötigt das Buchungssystem Zugriff auf den Terminkalender des Benutzers. Mittels eines Discovery Service kann das Buchungssystem feststellen welcher Service Provider den Terminkalender des Benutzers verwaltet. Sofern der Benutzer den eingeschränkten Zugriff auf den Terminkalender durch das Buchungssystem gestattet hat kann das Buchungssystem die für die Buchung des Bahntickets relevanten Informationen vollautomatisch abrufen. Entsprechend wird das Buchungssystem dazu in die Lage versetzt ein Ticket für eine in den Terminkalender passende Zugverbindung zu buchen.

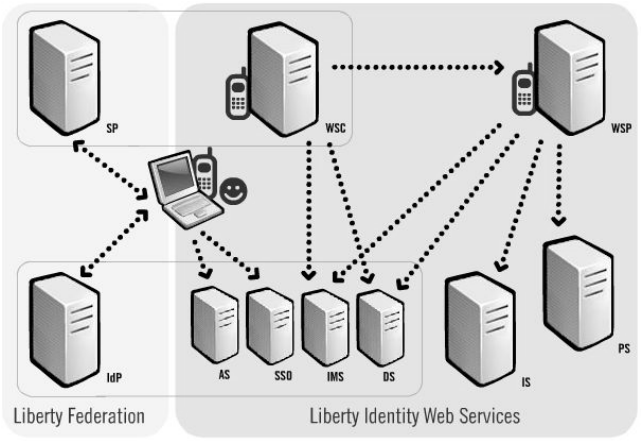


Abb. 7: Liberty Federation und Liberty Web Services im Vergleich [24]

Die Bestrebungen seitens Liberty Alliance gehen in die Richtung immer mehr Ressourcenarten zu standardisieren. Genau so wie sich der Terminkalender eines Benutzers für einen anderen Service Provider als nützlich erweisen kann sind ähnliche Applikationen mit Fotoalben, Sozialen Beziehungen, Favoriten, Blogg-Beträgen etc. denkbar. Liberty Web Services ermöglichen Applikationen, die heute noch unmöglich sind.

5. ANWENDUNGSBEISPIEL

Zur Veranschaulichung des Einsatzes von Identity Federation wird die Netzausweis Implementierung der Deutschen Telekom als Anwendungsbeispiel herangezogen. Das Netzausweis Projekt basiert auf den Standards SAML und Liberty Federation. Es erreichte im Jahr 2005 die Marktreife und wurde sofort eingeführt. In das Projekt involviert waren zunächst die drei Sparten des Deutsche Telekom Konzerns: T-Home, T-Mobile und T-Systems. Ziel war die Vereinigung der Benutzerkonten der Kunden. Dieser Schritt war sowohl für die Deutsche Telekom selbst, jedoch auch für den Kunden aufgrund bereits in dieser Ausarbeitung genannten Beweggründen von Vorteil [25].

Obwohl es sich bei den Sparten der Deutschen Telekom um Teile desselben Konzerns handelt waren aufgrund der jeweiligen Größe die Unterschiede im Bezug auf die Informationssystem-Landschaft enorm. Somit stand die Deutsche Telekom vor derselben Problemstellung wie mehrere unabhängige Unternehmen, die auf Informationsaustausch angewiesen sind. Im Zuge der erfolgreichen Markteinführung konnten weitere Unternehmen für den Netzausweis gewonnen und an das Identity Federation Netzwerk der Deutschen Telekom angebunden werden. Angetrieben durch die steigende Anzahl an Partnerunternehmen wurde das Netzausweis Projekt verlängert und um Multi-Protokoll Funktionalität erweitert. Neben den bereits implementierten Standards SAML und Liberty Federation wurden auch die Protokolle von OpenID und Microsoft Cardspace integriert [25].

Für die erfolgreiche Implementierung und Einführung des Netzausweises wurde die Deutsche Telekom mit dem IDDY Award 2006 ausgezeichnet. Inspiriert durch die Multi-Protokoll Erweiterung des Netzausweises wurde die Deutschen Telekom ein weiteres Mal ausgezeichnet, dieses Mal mit dem IDDY Award 2008 [25]. Der Netzausweis ist somit der reale Beweis dafür, dass Liberty Federation Lösungen umsetzbar sind und sich bereits in der Praxis bewährt haben.

6. ZUSAMMENFASSUNG UND AUSBLICK

Identity Federation gehört zu den wegweisenden Technologien des 21. Jahrhunderts. Mit der stetig steigenden Anzahl an Service Providern im Internet und der ebenso rasant steigenden Anzahl an

Benutzern dieser Services wird Identity Federation ein absolut notwendiges Mittel darstellen um die entstehende Komplexität zu bewältigen und kontrollierbar zu machen.

Bereits heute sind Bemühungen die Benutzerfreundlichkeit von Service Angeboten zu erhöhen auf der Tagesordnung. Nicht zuletzt kann die immer tiefergreifende und umfassendere Vernetzung von Service Angeboten durch Single Sign-Ons oder zumindest den Einsatz von gemeinsamen genutzten Digital Identity Datenbanken für verwandte Service Angebote die Komplexität für den Endbenutzer stark reduzieren. Als aktuelle Beispiele sind Windows Live ID von Microsoft, Apple ID von Apple oder myTUM von der TU München zu nennen, welche bereits innerhalb dieser großen Organisationen für eine bemerkenswerte Komplexitätsreduzierung sorgen.

Der nächste Schritt wird sein über Unternehmensgrenzen hinweg die unzähligen Digital Identities in wenige Federated Identities zusammenzuführen und zu bündeln. Aktuelle Ansätze wie die von OASIS oder Liberty Alliance stehen stellvertretend für diese Bewegung und haben darüber hinaus große Aussichten auf Erfolg. Nicht zuletzt lassen sich diese guten Erfolgsaussichten durch die vielversprechenden bis sehr erfolgreichen Pilotprojekte und die Geschlossenheit der maßgeblich an der Entwicklung beteiligten Unternehmen begründen.

Man hat offensichtlich aus Fehlern der Vergangenheit gelernt. Noch heute bekommt man die redundante Entwicklung verschiedener HTML Standards von diversen Unternehmen zu spüren. Anstatt dieselbe Technologie an verschiedenen Stellen redundant und getrennt voneinander weiterzuentwickeln war es beim Großprojekt Identity Federation offenbar möglich sich mit dem Großteil der maßgeblichen Unternehmen und somit auch mit der direkten Konkurrenz zusammenzufinden und an einer gemeinsamen Lösung zu arbeiten. Selbst getrennt voneinander operierende Projektgruppen waren und sind in der Lage aktiv Informationen auszutauschen und sicherzustellen, dass die Entwicklungen nicht in völlig unterschiedliche Richtungen laufen und stattdessen auf derselben Basis aufbauen.

Gemessen an den vielversprechenden Pilotprojekten der aktuellen Tage und den ersten im Business-2-Business Sektor produktiv genutzten Identity Federation Lösungen, speziell im Bereich Outsourcing und Supply Chain Management, kann die vollständige Marktreife nicht mehr allzu lange auf sich warten lassen. In dem Moment der Markreife wird Federated Identity auch Einzug erhalten bei Service Providern deren Angebote sich an private Endbenutzer richten. Letztendlich möchte der Großteil der privaten Endbenutzer von IT und Service Angeboten möglichst unkompliziert benutzen. Es wird immer mehr darauf ankommen Technologien intuitiv zugänglich und bedienbar für jedermann zu machen. Die enorme Steigerung der Benutzerfreundlichkeit durch Identity Federation ist nicht die Lösung aller Probleme, dennoch wird sie die IT ein Stück näher an die reale Welt heran führen und besser in den Alltag der Menschen integrieren.

7. ABBILDUNGEN

8. QUELLEN

[1] Krcmar, H., Informationsmanagement, Springer

[2] „Identity and Access Management", Allan Milgate, http://identityaccessman.blogspot.com/, 29.1.2009

[3] „Federated Identity Management", Sun Microsystems, http://www.sun.com/software/media/flash/demo_federation/index.html, 29.1.2009

[4] „Federated Identity", Wikipedia Foundation, http://en.wikipedia.org/wiki/Federated_identity, 29.1.2009

[5] Eberhart, A. and Fischer, S., Web Services: Grundlagen und praktische Umsetzung in J2EE und .NET, Hanser

[6] „Web Service", Wikipedia Foundation, http://en.wikipedia.org/wiki/Web_service, 29.1.2009

[7] „Identity Federation Multi-Protocol Solutions", Symlabs, http://symlabs.com/solutions/identity-federation, 29.1.2009

[8] „Whats Federated Identity Management?", David F. Carr, http://www.eweek.com/c/a/Channel/Whats-Federated-Identity-Management/, 29.1.2009

[9] „Shibboleth 2 Available", Internet2, http://shibboleth.internet2.edu/shib-v2.0.html, 29.1.2009

[10] „Liberty Alliance Complete Specifications ZIP Package", Liberty Alliance, http://www.projectliberty.org/resource_center/specifications/liberty_alliance_complete_specifications_zip_package_21_november_2008, 29.1.2009

[11] „OASIS – Who we are", OASIS, http://www.oasis-open.org/who/, 29.1.2009

[12] „Categories and Dues", OASIS, http://www.oasis-open.org/join/categories.php, 29.1.2009

[13] „OASIS Security Services TC", OASIS, http://www.oasis-open.org/committees/security/charter.php, 8.1.2009

[14] „SAML V2.0 Executive Overview", OASIS, http://www.oasis-open.org/committees/download.php/13525/sstc-saml-exec-overview-2.0-cd-01-2col.pdf, 8.1.2009

[15] „Liberty Federation Strategic Initiative ", Liberty Alliance, http://www.projectliberty.org/liberty/strategic_initiatives/federation, 29.1.2009

[16] „ SAML V2.0 Technical Overview ", OASIS, http://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf, 8.1.2009

[17] „About", Liberty Alliance, http://www.projectliberty.org/liberty/about, 29.1.2009

[18] „History", Liberty Alliance, http://www.projectliberty.org/liberty/about/history, 29.1.2009

[19] „Current Members", Liberty Alliance, http://www.projectliberty.org/liberty/membership/current_members, 29.1.2009

[20] „Strategic Initiatives“, Liberty Alliance, http://www.projectliberty.org/liberty/strategic_initiatives, 29.1.2009

[21] „Liberty Interoperable“, Liberty Alliance, http://www.projectliberty.org/liberty/liberty_interoperable, 29.1.2009

[22] „Liberty Web Services“, Liberty Alliance, http://www.projectliberty.org/liberty/strategic_initiatives/web_services, 29.1.2009

[23] „Liberty Alliance Web Services Framework: A Technical Overview“, Liberty Alliance, http://www.projectliberty.org/liberty/content/download/4120/27687/file/idwsf-intro-v1.0.pdf, 29.1.2009

[24] „Liberty Alliance“, Wikipedia Foundation, http://en.wikipedia.org/wiki/Liberty_alliance, 29.1.2009

[25] „Case Study: Deutsche Telekom lowers implementation barriers for online IP based services“, Liberty Alliance, http://www.projectliberty.org/liberty/content/download/4421/29639/file/Deutsche%20FINAL9.08.pdf, 8.1.2009

Geeignete Repräsentation von Wissen & Regeln in einem vernetzten System

Philipp Dirding
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
philipp.dirding@mytum.de

Kurzfassung
In diesem Paper soll ein Überblick über die Organisation und den Einsatz von Wissen und Schlussfolgerungen auf diesem Wissen gegeben werden. Zuerst wird ein Einblick in ein Einsatzgebiet gegeben. Hier beschränke ich mich auf das Autonomic Computing, das mit dem von IBM entwickelten Self-Management Konzept erklärt wird.

Anschließend folgt die Definition der eigentlichen Wissensrepräsentation durch Ontologien. Hier soll das theoretische Modell kurz erläutert werden und danach in der Praxis verwendete Konzepte des Modells vorgestellt werden.

Abschließend soll das eAutomation-Konzept von IBM als Referenzmodell vorgestellt werden, als Beispiel für eine Umsetzung von Wissensrepräsentation in der Praxis.

Keywords (Schlüsselworte)
Wissen, Regel, Inferenzregel, Ontologie, Autonomic Computing, Self-Management, eAutomation

1. EINFÜHRUNG
In heutigen Systeme werden sowohl die Hardware als auch die Software immer komplexer[2]. Besonders ERP-Systeme (Enterprise Resource Planning-Systeme) wie SAP ERP oder Oracle E-Business Suite sind nur mit erheblichen Aufwand zu installieren, konfigurieren und betreiben. Aufwand heißt in diesem Zusammenhang, dass hohe Kosten durch qualifizierte IT-Experten entstehen und der Betrieb viel Zeit verschlingt.

Aber nicht nur einzelne Anwendungen sind komplex, sondern auch die Integration in eine Umgebung mit verschiedenen unterschiedlichen Systemen[2]. Gerade diese Integration wird in Zeiten des Pervasive Computing, in denen mehr und mehr Informationstechnologien im Alltag eingesetzt werden und zusammenarbeiten müssen, immer wichtiger und umfasst immer mehr Systeme[2].

Diese Komplexität der einzelnen Systeme und der Systemlandschaften ist in der Vergangenheit immer größer geworden und wird auch in Zukunft weiter wachsen.

Um den Problemen, wie hoher Installations-, Konfigurations- und Administrationsaufwand, bei diesen komplexen Systemen zu begegnen, wurde, besonders angetrieben von IBM, das Konzept des Autonomic Computing entwickelt[8].

Autonomic Computing hat sich zum Ziel gesetzt, die Administratoren von Detailfragen und Routineaufgaben zu befreien und ihre Arbeit auf die Gestaltung von groben Vorgaben für das System zu beschränken. Mit diesen Vorgaben sollen sich die Systeme selbst organisieren und an ihre Umgebung anpassen, damit sie ihre Aufgaben möglichst gut erfüllen.

Autonomic Computing wird in [2] verglichen mit dem Nervensystem verglichen, das Herzfrequenz und Körpertemperatur regelt, ohne dass das Gehirn bewusst in diese Prozesse eingreifen muss.

1.1 SELF-CHOP
Autonomic Computing umfasst mehrere Teilbereiche. IBM teilt sie in vier verschiedene Aufgaben auf: Self-configuration, Self-healing, Self-optimization und Self-protection. Abgekürzt werden diese vier Punkte nach ihren Anfangsbuchstaben als Self-CHOP[1].

1.1.1 Self-configuration
Unter Self-configuration versteht man die autonome Installation, Konfiguration und Integration in die Umgebung, großer komplexer Systeme. Diese Prozesse können, von IT-Experten durchgeführt, in einer heterogenen Hard- und Softwarelandschaft sehr zeitaufwendig und kostspielig sein. Autonome Systeme erledigen die Aufgaben selber nach Zielvorgaben auf einer höheren Ebene (high level policies). Diese Zielvorgaben beschreiben Wunschzustände, aber nicht wie diese erreicht werden[2].

In Self-configuration-Umgebung müssen Systeme nur noch in diese eingebracht werden und danach erkennt das System die Umgebung eigenständig und installiert und konfiguriert sich entsprechend der anderen Systeme in der Umgebung und ebenso erkennen die anderen Systeme die neue Komponente und reagieren auf diese entsprechend um die Zielvorgaben zu erfüllen[2].

1.1.2 Self-healing
Systeme erfüllen das Konzept Self-healing wenn sie eigenes Fehlverhalten autonom erkennen, verfolgen und die Ursache bestimmen. Als Fehlverhalten gelten hier Bugs oder Ausfälle in Hard- und Software. Die Diagnose kann über verschiedene Arten von Detektoren geschehen. So können Sensoren gewisse Parameter überwachen oder Logdateien analysiert werden und mit dem Sollstatus des Systems verglichen werden[2].

Dann soll die gestellte Diagnose mit Fehlerbehebungsroutinen verglichen werden, zum Beispiel Softwarepatches und diese dann angewendet und nochmals getestet werden[2]. Alternativ wird das Problem an die Administratoren weitergegeben.

Self-healing-Funktionen übernehmen damit die Arbeit der großen Abteilungen, die für die Identifikation, Verfolgung und Ursachenfindung von Fehlern zuständig sind und für die Arbeit oft Wochen brauchen und nicht immer zu Lösungen kommen[2].

1.1.3 Self-optimization
In komplexen Systemen gibt es sehr viele verschiedene Stellschrauben für Performanz[2]. Menschen können diese erstens nicht alle überblicken und zweitens ihre Auswirkungen oft nicht

abschätzen, insbesondere im Zusammenhang mit anderen integrierten Komponenten.

Unter Self-optimization versteht man die Verlagerung der Aufgabe das System möglichst performant einzustellen auf das System selber. Durch kontinuierliche Überwachung und Experimentieren mit den Möglichkeiten die Performanz zu erhöhen, lernt das System immer genauer die Entscheidungen zu treffen, die zu einer erhöhten Performanz und Effizienz führen.

1.1.4 Self-protection

Unter self-protection-Systemen versteht Systeme, die sich selbst vor Angriffen von außerhalb des Systems und vor Verkettungen von Fehlern, die nicht durch self-healing-Funktionen behoben werden konnten. Diese Angriffe und Fehler werden nicht nur bei akutem Auftreten erkennt und beseitigt werden, sondern das System soll diese durch Analyse verschiedener Sensoren antizipieren können und sich dagegen schützen[2].

2. WISSENSREPRÄSENTATIONEN

Autonomic Computing-Systeme handeln somit zu einem gewissen Grad, wie der Name schon sagt, autonom nach Zielvorgaben. Es liegen nicht wie bei automatischen Systemen genaue Handlungsanweisungen vor, nach denen das System agiert, sondern Entscheidungen muss das System eigenständig treffen. Diese Entscheidungen treffen Systeme auf der Basis einer Wissensgrundlage. Regeln, die aus den Zielvorgaben auf höherer Ebene generiert werden, werden auf dieses Wissen angewendet. Dieses Anwenden der Regeln auf das Wissen ermöglicht den autonomen Systemen Schluss zu folgern (to reason). Das Ergebnis dieser Schlussfolgerungen ist wiederum neues Wissen oder Handlungsanweisungen.

Welches Wissen das genau sein kann, hängt von den einzelnen Modellen und Implementierungen ab, aber das Wissen muss im System irgendwie repräsentiert werden. Eine Möglichkeit Wissen zu repräsentieren sind Ontologien.

2.1 Ontologie

Ontologien sind formale Spezifikationen, für die Strukturierung von Wissen. Zwei Arten von Regeln bestimmen diese Struktur. Integritäts und Inferenzregeln[9].

Die Integritätsregeln, sind Regeln die den Aufbau des Wissens bestimmen. Sie bestimmen zum Beispiel welche Beziehungen zwischen Wissensentitäten erlaubt sind oder wie einzelne Wissensentitäten aussehen können.

Die Inferenzregeln ermöglichen das Schlussfolgern auf der Basis des vorhandenen Wissens. Aus dem vorhandenen Wissen lassen sich Schlüsse ziehen und somit neues Wissen generieren. Ist zum Beispiel bereits das Wissen vorhanden, dass sich zwei Server A und B im selben Raum befinden, und dazu bekannt wird, dass B und C im selben Raum stehen, lässt sich schlussfolgern, dass A und C auch im selben Raum stehen.

Da sowohl Mensch als auch Maschine diese Regeln kennen, nach denen das Wissen aufgebaut ist, ermöglichen Ontologien die Kommunikation zwischen Mensch und Maschine.

Ontologien sind immer auf einen bestimmten Einsatzzweck spezialisiert, der eine bestimmte Wissenstruktur erfordert und daher gibt es viele verschiedene Ontologien, die sich mehr oder minder stark unterscheiden[4]. Es wäre unpraktisch, wenn nicht sogar unmöglich alles vorhandene Wissen in einer Ontologie unterbringen zu wollen. Nun gibt es mehrere Wege, das Wissen strukturiert sein kann, d.h. die Integritätsregeln aussehen. Im Folgenden sollen einige gebräuchliche Konzepte vorgestellt werden.

2.2 WISSENSTRUKTUREN

2.2.1 Taxonomie

Einer der einfachsten Ansätze Wissen zu strukturieren ist eine Taxonomie. Eine Taxonomie teilt Gegenstände, genannt Entitäten, hierarchisch in Kategorien ein.

Erfunden wurde die Taxonomie von dem Biologen Carl von Linné zur Einteilung aller Lebensformen in Familien, Gattungen, Arten[4].

Taxonomien bilden Über- und Unterordnungsbeziehungen zwischen den Entitäten ab und können somit auch Vererbung darstellen.

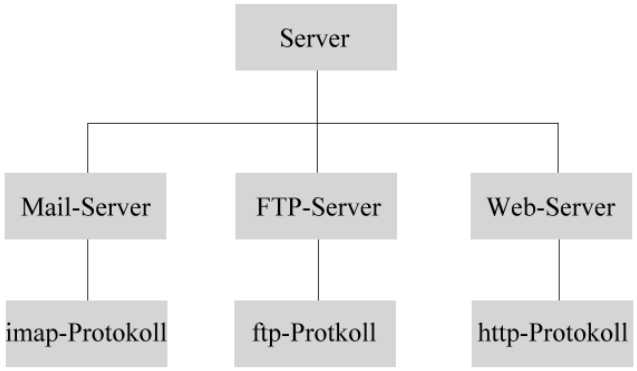


Abbildung 1. Eine einfache Taxonomie.

Beziehungen zwischen Kategorien einer Ebene sind nicht vorgesehen.

2.2.2 Thesaurus

Ein Thesaurus strukturiert ebenfalls Wissen ist ein Schlagwortverzeichnis oder kontrolliertes Vokabular, dessen Begriffe untereinander in Beziehung stehen.

Zu einem Begriff werden zum Beispiel Ober- und Unterbegriffe, Synonyme, Homonyme oder Äquivalenzbeziehungen gespeichert. Und jeder dieser Begriffe hat selber wieder Ober- und Unterbegriffe usw. So lassen sich Themenbereiche genau beschreiben und repräsentieren.

Für Thesauri gibt es verschiedene Normen, die zum Beispiel die verschiedenen Arten der Relationen vorgeben.

LAN

-Synonyme
  Local-Area Networks
  Ethernet
  Inhouse-Netz
  Local Area Network
  Lokales Netz
-Oberbegriffe
  Computernetz
-Verwandte Begriffe
  Intranet
  Unternehmensnetzwerk
-Zuordnung
  N.10.07.01 Informationstechnik und Systemarchitektur
  B.09.02 IS-Entwicklung und -Betrieb

Abbildung 2. Thesaurus LAN[11]

2.2.3 Semantisches Netz

Ein semantisches Netz ist ein graphisches Modell, in dem Begriffe untereinander in Beziehung gesetzt werden. Begriffe werden durch Knoten dargestellt, während die Kanten zwischen den Knoten die inhaltlichen Beziehungen der Begriffe untereinander repräsentieren.

Es gibt verschiedene semantische Netze, die unterschiedliche Beziehungstypen erlauben. Die vorher vorgestellten Wissensstrukturen Taxonomie und Thesaurus sind semantische Netze mit einer begrenzten Anzahl von Relationen[7].

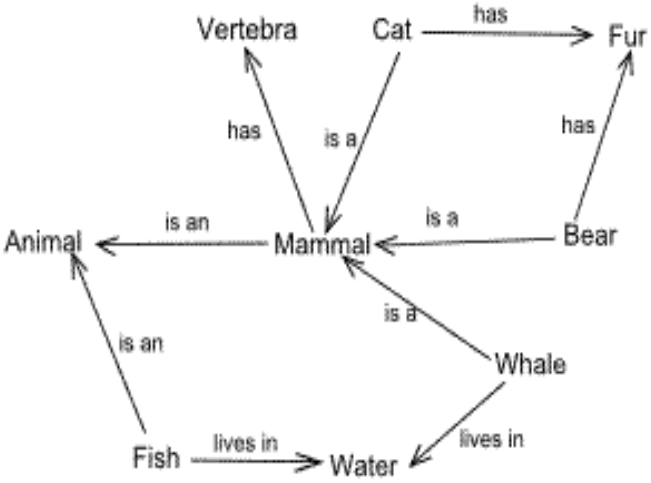


Abbildung 3. Beispiel eines semantischen Netzes[7]

2.2.4 Frames

Frames orientieren sich an der Idee, dass der Mensch in gewissen Schemen, also Abstraktionen von der Realität, denkt und sollen den Rahmen für diese Schemen bilden[4].

Das Konzept erinnert stark an das Objekt-orientierte Klassen-Modell, allerdings fehlen im die Methoden. Außerdem ist es eng verwandt mit dem semantischen Netz.

Jede Wissensentität ist ein Frame und hat einen Namen und besitzt Attribute, die Slots genannt werden. Diese Slots können sowohl eine Eigenschaften des jeweiligen Frames sein, als auch eine Beziehung zu einem anderen Frame darstellen[4].

Ausgehend von der menschlichen Idee eines Büros kann man ein Frame konstruieren:

Der Name des Frames ist „Büro“. In einem Büro erwartet man einen Schreibtisch, einen Bürostuhl und Fenster. Diese drei Dinge sind Slots und stellen Beziehungen zu jeweils einem anderen Frame dar. Der Schreibtisch ist wieder ein Frame, das Slots (z.B. Material) hat. Ein Slot der eine Eigenschaft des Frames „Büro“ ist, könnte zum Beispiel die Größe in Quadratmetern sein.

So bildet sich ein „Netz“ aus Gegenstände oder Wissensentitäten in Form von Frames.

2.2.5 Prädikatenlogik

Die Aussagenlogik ist die Grundlage für die Prädikatenlogik. Sie besteht aus Prädikaten und Termen. So kann der Satz „Die Sonne scheint“ durch ein Prädikat und einen Term dargestellt werden:

scheint(Sonne)[5]

Prädikate stellen die Namen von Eigenschaften, Relationen oder Klassen dar, während Terme Objekte des Diskursbereichs darstellen[6].

So lassen sich Relationen zwischen verschiedenen Termen bilden, wenn sie zum Beispiel ein Prädikat teilen.

Es ergibt sich allerdings ein Problem bei der Prädikatenlogik. Die Folgerungen geschehen ohne inhaltliche Wertung und somit gibt es einen Satz „erlaubter“ Lösungen, aber man keine dieser Lösungen kann „empfohlen“ werden[4].

Prädikatenlogik ist jedoch einfach in Software zu implementieren, da mit Prolog eine logische Programmiersprache existiert die auf diesem Modell basiert.

3. REFERENZMODELL: eAutomation

Da vor allem das Self-Management von Systemen stark von ihnen vorangetrieben wird, hat IBM ein Referenzmodell für dieses Konzept entwickelt, genannt eAutomation. Dieses Referenzmodell ist zum Beispiel in die IBM Tivoli System Automation für z/OS eingeflossen.

3.1 Correlation engine

eAutomation ist eine correlation engine mit der sich drei Analysetypen durchführen lassen[3]:

- Data filtering
- Thresholding
- Sequencing

Data filtering ist zum Beispiel die Unterdrückung multipler Fehlermeldungen, die ein und dieselbe Ursache haben. Produziert zum Beispiel ein Netzwerkswitch Fehler, so sind bestimmte Server nicht mehr erreichbar. Hier werden vom Netzwerkswitch, von den Servern, die nicht mehr erreichbar sind, und von den Komponenten, die auf den Server zugreifen wollen Fehlerberichte geschickt, obwohl der Fehler mit der Benachrichtigung vom Switch umfassend beschrieben ist[3].

Thresholding beschreibt Grenzwerte. Über Thresholding kann zum Beispiel beschrieben werden, dass wenn eine Aktion x Mal fehlschlägt, ein ernsthafter Fehler aufgetreten ist[3].

Sequencing alarmiert zum Beispiel, wenn eine Sicherheitstür mehr als eine bestimmte Zeit offen steht[3].

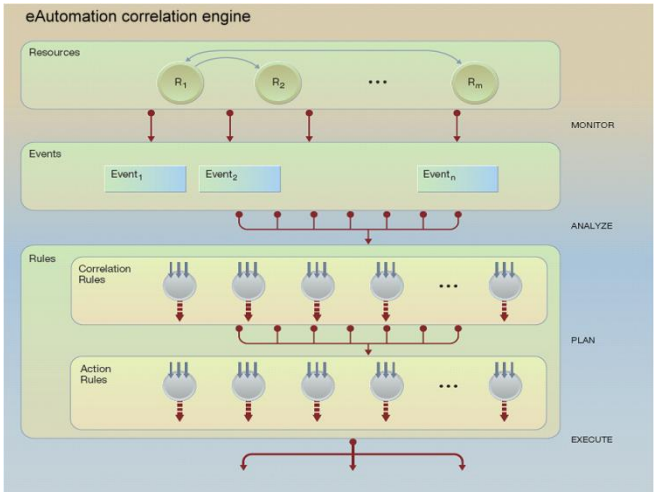


Abbildung 4. Die eAutomation correlation engine[3]

Die correlation engine in der Lage erstens auf Probleme bzw. Fehler zu reagieren und zweitens kann sie Probleme und Fehler antizipieren um sie zu vermeiden.

3.2 MAPE-Modell

Das MAPE-Modell beschreibt die Gesamtarchitektur des Self-Management-Systems und bricht dieses in vier einzelne Teile:

- Monitor
- Analyze
- Plan
- Execute

Diese vier Teile sind im Grunde Teil eines jeden Managementprozesses. Zuerst werden vorhandene Daten gesammelt (Monitor). Diese gesammelten Daten werden analysiert (Analyze). Aus den gewonnen Erkenntnissen werden Aktionen vorbereitet (Plan) und ausgeführt (Execute).

Da alle vier Schritte auf demselben Problembereich arbeiten, muss sichergestellt werden, dass das geteilte Wissen von allen verstanden wird.

Diese Integration wird durch eine Aufteilung der correlation engine auf drei Ebenen[3]:

- Resource layer
- Event layer
- Rule layer

Diese Aufteilung korreliert mit dem MAPE-Modell wie man in Abbildung 4 entnehmen kann.

Resourcen können quasi alles sein, aber sie stellen immer etwas dar, was vom Self-management-System verwaltet werden soll. Das kann ein Server, ein bestimmtes Programm oder auch ein Service sein.

Events sind nichts anderes als Statusänderungen von Ressourcen (z.B. Ausfall eines Webservers) und durch Weitergabe dienen sie als Benachrichtigungen.

Diese Benachrichtigungen werden vom rule layer aufgenommen.

3.3 Ontologie

Das eAutomation-Modell besitzt eine eigene Ontologie, die auf der KOANA-Ontologie der Uni Karlsruhe basiert, welche wiederum auf semantischen Netzen basiert.

3.3.1 Resourcen

Das Kernelement der eAutomation-Ontologie ist die Ressource, da diese das eigentliche zu managende Objekt darstellt. Das Modell, wie es in der Ontologie realisiert wurde ist in Figure-3 dargestellt.

Jede Ressource hat einen eindeutigen Namen. Der Status der einer Ressource und damit zwei weitere sehr wichtige Attribute sind Current_Operational_State und Desired_Operational_State[3]. Ersteres beschreibt den aktuellen Zustand der Ressource. Jede Änderung dieses Attributs ist ein Event. Zweiteres Attribut spezifiziert den gewünschten Zustand der Ressource. Werte für diese Attribute sind nicht frei wählbar sondern vordefiniert. So kann für einen Server zum Beispiel gelten, dass er für die beiden genannten Attribute „online“ und „offline“ annehmen kann.

Ressourcen untereinander können zwei Zusammenhänge haben. Es gibt start/stop-Beziehungen und location-Relationen. Eine

Start/stop-Relation ist zum Beispiel *startAfter*, die angibt, dass für den Start einer Ressource die in startAfter enthaltene, den Current_Operational_Status „online“ haben muss. Location-Relationen sind zum Beispiel *Collocated* und *AntiCollocated*. *Collocated* gibt an, dass eine Ressource nur da in Betrieb sein darf wo die entsprechend andere vorhanden ist. *AntiCollocated* beschreibt genau das Gegenteil.

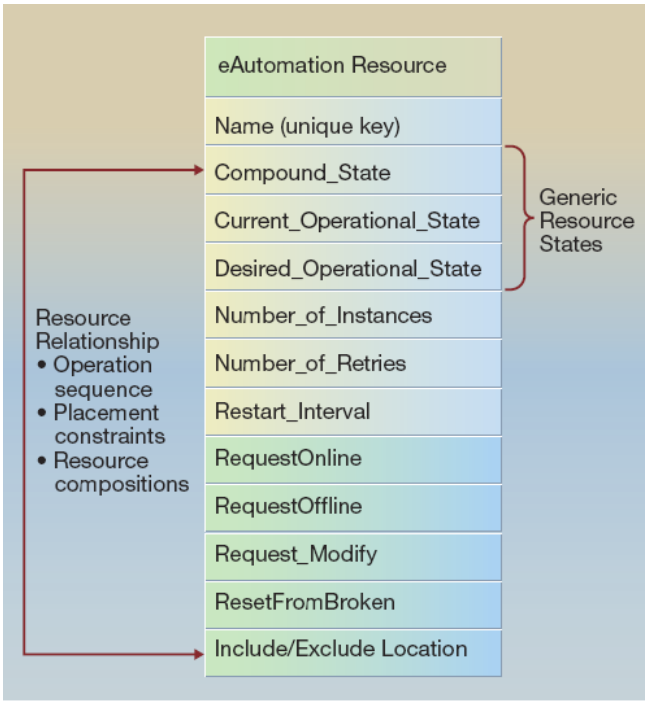


Abbildung 5. Modell einer Ressource[3]

3.3.2 Regeln

Die eAutomation correlation engine unterstützt zwei Arten von Regelsätzen: correlation und action rules.

Correlation rules können zum Beispiel Zusammenhänge zwischen Events erkennen und diese Zusammenfassen und mit einer sinnvollen Meldung versehen (data filtering). Sie erleichtern dem Administrator die Problemsuche oder das Verständnis des Problems.

Action rules dagegen sind Regeln, die basierend auf Events aktiv im System Veränderungen erzeugen können. Das nimmt dem Administrator tatsächliche Arbeit ab.

4. Literatur

[1] Miller, B. (2005, September). The autonomic computing edge: Can you chop up autonomic computing. http://www.ibm.com/developerworks/library/ac-edge4/index.html (16.11.2008)

[2] Kephart, J. O. and D. M. Chess (2003). The vision of autonomic computing. Computer 36 (1), 41-50.

[3] Stojanovic, L., J. Schneider, A. Maedche, S. Libischer, R. Studer, T. Lumpp, A. Abecker, G. Breiter, and J. Dinger (2004). The role of ontologies in autonomic computing systems. IBM Systems Journal 43 (3).

[4] Davis, R., H. Shrobe, and P. Szolovits (1993). What is a knowledge representation? AI Magazine 14 (1), 17-33.

[5] o.V. Prädikatenlogik. http://www.ifi.uzh.ch/req/courses/logische_programmierung/ws03/documents/Praedikatenlogik.pdf (12.11.2008)

[6] François Bry. (2004). Wie können Daten aus dem Web automatisch gefunden werden? Aspekte eines aktuellen Informatikvorhabens

http://www.pms.ifi.lmu.de/mitarbeiter/bry/tag-der-mathematik-2004/tag-der-mathematik-2004.html (13.11.2008)

[7] http://en.wikipedia.org/wiki/Semantic_net (18.11.2008)

[8] http://de.wikipedia.org/wiki/Autonomic_Computing (28.01.2009)

[9] http://de.wikipedia.org/wiki/Ontologie_(Informatik) (28.01.2009)

[10] http://de.wikipedia.org/wiki/Wissensrepr%C3%A4sentation(28.01.2009)

[11] http://www.genios.de/thesaurus/subthesaurus/deskriptor/desk 2827.html

Mechanismen zur automatischen Konfiguration von Netzwerkkomponenten und Services

Andreas Maier
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
maierand@in.tum.de

ABSTRACT (Kurzfassung)
In diesem Ausarbeitung werden die theoretischen Grundlagen zur automatischen Konfiguration von Netzwerkkomponenten und Services beschrieben: allgemeine mit maschinellem Lernen lösbare Problemfelder und die durch Knowledge Plane zu lösenden Aufgaben.

Keywords (Schlüsselworte)
Knowledge Plane, Maschinelles Lernen, Klassifikation und Regression, Acting & Planning, Interpretieren und Verstehen, Rapid Configuration of Network, Anomaly detection

1. INTRODUCTION
In diesem Vortrag möchte ich die neuen Methoden und Mechanismen zur automatischen Konfiguration von Netzwerkkomponenten und Services betrachten, ein neues vor kurzem entwickeltes Paradigma der Netzwerkverwaltung - Knowledge Plane.

2. Lernen in Computersystemen
Knowledge Plane – ist ein verteiltes und dezentralisiertes Konstrukt für die Sammlung und Verarbeitung von Informationen in einem Netwerk. Die Informationen werden von unterschiedlichsten Schichten gesammelt von Anwendungsschicht bis zum Physical layer. Eigenes Verhalten wird erforscht und aufgetretene Probleme analysiert mit dem Ziel, beispielsweise die Betriebseigenschaften so zu steuern, dass größere Ausfallsicherheit und bessere Arbeitsleistung oder erhöhte Sicherheit erreicht wird. Dazu benötigt die Knowledge Plane das Konzept maschinellen Lernens.

Maschinelles Lernen ist ein Oberbegriff für die „künstliche“ Generierung von Wissen aus Erfahrung: Ein künstliches System lernt aus Beispielen und kann nach Beendigung der Lernphase verallgemeinern. Das heißt, es lernt nicht einfach die Beispiele auswendig, sondern es „erkennt“ Gesetzmäßigkeiten in den Lerndaten. So kann das System auch unbekannte Daten beurteilen.

Anderseits kann auch das Lernen in Computersystemen als Performance Task betrachtet werden. Das Bild 1. veranschaulicht dieses Modell. Das Rechenssystem versucht mit Hilfe von Lernregel die Umgebung ins Wissensmodell umzuwandeln. Die entstandenen Kenntnisse werden für die Verbesserung von Performance verwendet. Optional kann man die vorhandenen Kenntnisse dazu benutzen um den Lernprozess zu beeinflussen oder zu verbessern. Das Performance wiederum beeinflusst die Umgebung. Dabei unterscheidet man zwischen Online und Offline Lernen. Beim Offline Lernen wird das vorhandene Wissen nur einmal ins Performance transformiert dagegen wiederholt sich bei Online Lernen der Vorgang als endlose Schleife.

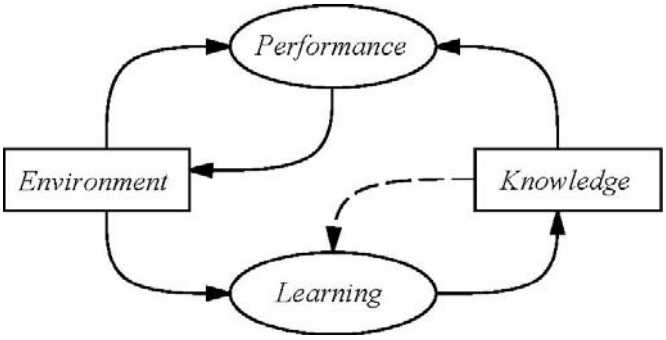


Bild 1. Lernen in Computersystemen als Performance Task.[2]

3. Mit maschinellem Lernen lösbare Problemfelder
In dem maschinellen Lernen werden die Lernprobleme in drei Hauptkategorien unterteilt: Klassifikation und Regression auf Messdaten, Acting & Planning, Interpretieren und Verstehen.

Formulation	Performance Task
Classification & Regression	predict y given x predict rest of x given part of x predict $P(x)$ given x
Acting & Planning	iteratively choose action a in state s choose actions $\langle a_1, \ldots, a_n \rangle$ to achieve goal g find setting s to optimize objective $J(s)$
Interpretation & Understanding	parse data stream into tree structure of objects or events

Bild 2. Übersicht der mit maschinellem Lernen lösbaren Problemfeldern.[2]

3.1 Klassifikation und Regression auf Messdaten
Bei Klassifikation wird zu jeder Beobachtung ein Label (Klasse) gegeben. Z.B. „Server antwortet nicht“, „Server ist überlastet“, „Keine Route zum Server“ bei der Untersuchung der Verbindungsabbrüche.

Bei Regression versucht man den unbekannten Wert anhand der vorhandenen Beobachtungen vorherzusagen, z.B. die Zeit, die man braucht um Verbindung zum Server aufzubauen.

Bei manchen Situationen ist es unmöglich alle benötigte Messungen durchzuführen, z.B. wegen allgemeiner Aufwendigkeit oder unvollständiger bzw. verrauschter Muster. In dem Fall sollen die Daten vervollständigt werden. Hier spricht man von Mustervervollständigung(pattern completion, flexible prediction).

Wir betrachten noch die Methoden, die dafür benutzt werden um maschinell zu lernen. Dabei unterscheidet man zwischen betreutem Lernen ("supervised learning"), unbetreutem Lernen ("unsupervised learning") und halb-betreutem Lernen (semi-supervised learning).

Beim betreuten Lernen (supervised learning) gibt ein Mensch gleich ein „Lehrer“ die Werte der Zielfunktion für alle Trainingsbeispiele an. Es gibt Mehrzahl an Paradigmen für betreutes Lernen wie decision tree and rule induction[1] [2], neural network methods[3], nearest neighbor approaches[4], und probabilistic methods[5]. Die Paradigmen unterscheiden sich in Algorithmen der Wissensbeschaffung und Repräsentation des vorhandenen Wissens.

Die zweite Klasse der Wissensbeschaffung ist unbetreutes Lernen (unsupervised learning) - es gibt keine vorkategorisierten Beispiele. Hier gibt es auch mehrere Paradigmen, die man grob in zwei Kategorien unterteilen kann Clustering[6] [7] und density estimation[8]. In Falle vom Clustering ermittelt man Gruppen (Clustern) von Objekten, deren Eigenschaften oder Eigenschaftsausprägungen bestimmte Ähnlichkeiten (bzw. Unähnlichkeiten) aufweisen. Beim density estimation versucht man eine Dichtefunktion zu erstellen, die die vorhandenen Trainingswerte abdecken wird und dann auf neue Werte eingesetzt werden kann.

Die dritte Klasse der Wissensbeschaffung ist halb-betreutes Lernen (semi-supervised learning). Das ist wie der Name schon sagt eine Mischung von zwei schon beschriebenen Klassen. D.h. nur ein Teil der Trainingsbeispiele wird von Menschen klassifiziert. Der Rest sollte vom Rechensystem selbst bearbeitet werden.

3.2 Acting & Planning

Das nächste Aufgabefeld, das wir betrachten ist acting & planning, also das Anwenden von Wissen für Aktions- oder Planauswahl. In einfachster Form wird eine Aktion direkt ausgewählt ohne vorherige Aktionen in Betracht zu ziehen. In diesem Fall kann die Klassifikation direkt mit Aktionsauswahl verbunden werden, d.h. jeder ermittelten Klasse wird eine Aktion zugeordnet. Genauso kann die Aktionsauswahl mit der Regression verknüpft werden um die Werte vorherzusagen oder die Brauchbarkeit von Aktionen zu ermitteln. Aktionen können einzeln ausgewählt werden oder in Form von macro-operators zusammengesetzt um als Pakete ausgeführt zu werden.

Genauso wie mit Klassifikation und Regression gibt es auch bei Acting und Planning die speziellen Methoden, die dafür benutzt werden um maschinell zu lernen: Learning apprentice oder adaptive interface, programming by demonstration, behavioral cloning, unterstütztes Lernen (reinforcement learning), learning from problem solving and mental search, empirical optimization, z.B surface methodology.

[1] Quinlan, 1993

[2] Clark & Niblett, 1988

[3] Rumelhart, 1986

[4] Aha, 1991

[5] Buntine, 1996

[6] Fisher, 1987

[7] Cheeseman, 1988

[8] Priebe & Marchette, 1993

Learning apprentice[9] oder adaptive interface[10], d.h. das System lernt durch Beobachtung der Aktionen von Benutzern. Das System gibt dem Benutzer Empfehlungen, die er akzeptieren oder auch andere Entscheidungen treffen kann. Dadurch wandeln wir das Problem in das Problem des betreuten Lernens (supervised learning) und können auch die Methoden, die dafür entwickelt wurden, anwenden; sprich decision tree, rule induction,, neural network methods, nearest neighbor approaches, und probabilistic methods.

Das verwandte Paradigma ist programming by demonstration[11], wo dagegen versucht wird ein Set von macro-operatoren zu erstellen.

Ein weiteres verwandtes Modell behavioral cloning[12]. Hier lernt das System durch Beobachtung der Aktionen von Benutzern ohne Benutzer direkt zu befragen.

Außerdem existiert noch unterstütztes Lernen (reinforcement learning): Der Agent bekommt zwar Feedback von der Umwelt zu seiner Kategorisierung, erfährt aber nicht explizit, was die richtige Kategorisierung gewesen wäre, da es eventuell mehrere Schritte notwendig sind um den gewünschten Zustand zu erreichen.

Zum Beispiel in Falle vom dynamischen Netzwerkrouting: das System wird versuchen mehrere Routen aufzubauen, jeder Route schließt mehrere Entscheidungsschritte ein bis sie komplett aufgebaut ist. Jedoch werden die Entscheidungskriterien wie Routenmetrics erst am Ende von Routenaufbau bereit stehen.

Die eng mit unterstütztem Lernen verbundene Methode ist learning from problem solving and mental search[13]. In diesem Fall werden die Änderungen zuerst auf einem Modell erprobt.

Unser Routingbeispiel in diesem Fall würde folgendermaßen aussehen: das Lernsystem würde das gewünschte Verhalten von Netzwerk modellieren, bevor es dieses Verhalten praktisch umsetzen würde.

Und die letzte Methode der Wissensbeschaffung ist empirical optimization. In diesem Fall existiert kein Modell vom Testsystem. Es gibt Anzahl von Parametern, deren Auswirkung auf das Gesamtsystem unbekannt ist. Durch das Verändern von einzelnen Parametern oder Parameterbündeln wird versucht die Auswirkung auf das Gesamtsystem zu ermitteln. Ein Beispiel davon ist surface methodology[14].

3.3 Interpretieren und Verstehen

Das dritte Aufgabenfeld in maschinellem Lernen ist Learning for Interpretation and Understanding. Anstatt wie bei Klassifikation und Regression ein Vorhersagemodell zu erstellen, versucht man hier ein Modell zu erstellen, das eine Erklärung mit Hilfe von tieferen Strukturen möglich macht. Z.B. könnte man ein anomales Transferverhalten in Netwerk dadurch erklären, dass es auf einem der Server eine größere Datei veröffentlicht wurde, auf die die Nachfrage hoch ist.

Es gibt Mehrzahl von Lernaufgaben und damit verbundene Erklärungsmodelle.

[9] Mitchell, 1985

[10] Langley, 1999

[11] Cypher, 1993

[12] Sammut, 1992

[13] Sleeman, 1982

[14] Myers & Montgomery, 1995

Der erste Ansatz ist, dass jede Traininginstanz mit damit verbundenes Erklärungsmodell kommt. Das erlaubt eine effektive Erklärungsmodell zu erstellen, ist aber leider mit hohem Erstellungsaufwand verbunden.

Der zweiter Ansatz ist, dass jede Trainingsinstanz ohne damit verbundenes Erklärungsmodell kommt, besitzt aber Backgroundwissen, woraus ein Erklärungsmodell erstellt werden kann.

Und der letzte Ansatz ist, dass jede Trainingsinstanz ohne damit verbundenes Erklärungsmodell kommt, aber auch ohne Backgroundwissen, woraus ein Erklärungsmodell erstellt werden kann. Das Lernsystem erstellt das Erklärungsmodell, indem es die Regelmäßigkeiten in Daten sucht.

4. Die durch Knowledge Plane zu lösenden Aufgaben

Mit Knowledge Plane können mehrere Aufgabenarten gelöst werden. Das sind einige davon: Anomaly Detection and Fault Diagnosis, Responding to intruders and Worms, Rapid configuration of Network

4.1 Network Configuration and Optimization

4.1.1 Das Spektrum von Konfigurationsaufgaben

Tabele 1. Das Spektrum von Konfigurationsaufgaben[2]

Problem	Global parameters	Local parameters	Topology	Components
Parameter Selection	X			
Compatible Parameter Configuration	X	X		
Topological Configuration	X	X	X	
Component Selection and Configuration	X	X	X	X

Es gibt mehrere Aufgaben für die Herstellung und Konfiguration der Netzwerke. Die einfachste davon ist das parameter selection, wo die mehreren Parameter optimiert werden müssen.

Die nächste Aufgabe ist compatible parameter selection. Hier werden die Komponente des Systems nach festen Regeln miteinander verbunden. Die Effektivität des Systems wird dadurch beeinflusst, dass die einzelnen Parameter kompatibel sein müssen um mit einander kommunizieren zu können. Zum Beispiel müssen die IP-Adressen und Subnetmaske bei der Konfiguration eines Netzwerkes bestimmten Topologieregeln folgen um miteinander kommunizieren zu können. Gesamte Systemsleistung kann ziemlich komplex von lokalen Parametern abhängen.

Die dritte Aufgabe beinhält topological configuration. Das System besteht aus einzelnen Komponenten, aber die Verbindungstopologie muss noch ermittelt werden. Zum Beispiel gibt es Mehrzahl an Arbeitsstationen, Gateways, Dateiserver, Drücker, Backupgeräten. Die Aufgabe ist das Netzwerk so zu konfigurieren, damit die Gesamtleistung maximal wäre. Natürlich muss jede einzelne Topologie mit compatible parameter selection optimiert werden.

Die vierte Aufgabe ist component selection and configuration. Am Anfang besteht die Konfiguration aus einem Katalog mit möglichen Komponenten und deren Preisen. Die benötigten Komponente und ihre Anzahl müssen ermittelt werden und danach muss natürlich topological configuration gelöst werden.

4.1.2 Reconfiguration process

Bis jetzt haben wir nur das Problem der Auswahl der richtigen Konfiguration betrachtet. Dennoch existiert das Problem wie die Konfiguration effektiv implementiert werden kann. Zum Beispiel während der Installierung eines Netzwerks werden normalerweise Gateways und Routers dann Datei und Druckserver und danach erst die Arbeitsstation installiert. Die Ursache dafür ist dass man den Konfigurations- und Test- Aufwand minimieren will, der zum Beispiel zur Rekonfiguration nötig wäre. Automatic configuration tools z.B. DHCP können Arbeitsstationen konfigurieren, nachdem der Server installiert worden ist.

4.1.3 Existing AI/ML Work Configuration

4.1.3.1 Parameter Selection

Wie schon früher erwähnt wurde ist Parameter Selection reines Optimierungsproblem, wenn ein Systemmodell bekannt ist. Wenn Systemmodell nicht vorhanden ist dann können die statistischen Methoden angewendet werden (empirische Optimierung).

4.1.3.2 Compatible Parameter Configuration

Das allgemein bekannte Modell der AI ist so genanntes constraint satisfaction problem (CSP). CSP wird als Graph dargestellt. Um das Problem effektiv zu lösen wird eine Gruppe von Algorithmen entwickelt[15]. Ein anderer Einsatz ist CSP als Erfüllbarkeitsproblem der Aussagenlogik darzustellen. Eine andere Möglichkeit wäre das CSP durch die randomisierten Algorithmen zu lösen, z.B. WalkSAT[16].

Das standarte CSP hat eine fixe Graphstruktur, die aber durch zusätzliche Graphen oder Beschränkungen erweitert werden kann. Auf diesem Gebiet können Methoden constraint logic programming (CLP)[17] und dafür entwickelte Programmiersprachen angewendet werden.

4.1.3.3 Topological Configuration

Es gibt zwei Hauptansätze für die topological configuration: refinement und repair.

Refinement Methoden starten mit einem einzigen „box“, das das ganze zu konfigurierende System darstellt. Dieses „box“ hat formale Spezifikation von gewünschtem Verhalten. Die Refinement-Regeln analysieren diese Spezifikation und ersetzen dieses „box“ mit zwei oder mehreren anderen „boxen“ mit entsprechenden Verbindungen. Zum Beispiel das kleine Office-Netzwerk sollte zuerst als ein „box“ dargestellt werden, das eine bestimmte Anzahl an Arbeitsstationen, Dateiserver, Drücker mit DSL- Leitung verbindet. Die Refinement- Regel soll ersetzen dieses „box“ mit dem lokalen Netzwerk und Router/NAT box. Die andere Refinement-Regel soll dieses Netzwerk als ein drahtloses Access-point und eine Anzahl an Netzwerkarten definieren. (Alternativ als Ethernet-switch eine Anzahl von Ethernetkarten und Verbindungskabeln). Es existieren die

[15] Kumar, 1992

[16] Selman, 1993

[17] Jaffar & Maher, 1994

Forschungen, die maschinelles Lernen auf dem Gebiet anwenden[18].

Der auf dem repair basierende Ansatz geht von einer Startkonfiguration aus, die die gewünschte Spezifikation nicht erfüllt und dann wird es versucht diese Konfiguration zu reparieren, bis sie gewünschte Bedingungen erfüllt. Zum Beispiel soll die Startkonfiguration alle Drücker, Computer und andere Geräte mit einem einzigen Switch verbinden, was viel zu teuer und gross sein könnte. Die repair-Regel ersetzt diesen einzigen Switch mit mehreren kleineren und billigeren Switchs. Es existieren mehrere auf repair basierende Algorithmen[19] [20] [21].

4.1.3.4 Component Selection and Configuration

Die bereits beschriebenen auf dem refinement und repair basierenden Methoden können weiter erweitert werden um component selection und configuration zu behandeln.

4.1.3.5 Changing Operating Conditions

Bis jetzt haben wir das Problem der Konfigurationsoptimierung unter den konstanten operativen Bedingungen betrachtet. Dennoch kann es passieren, das die optimale Konfiguration an die wechselnden Bedingungen anpassen sollte, wie es bei großen Netzwerkumgebungen üblich ist. Bis jetzt existieren leider keine Erforschungen auf dem Gebiet.

4.2 Anomaly Detection

Bei der Anomaly Detection wird ermittelt, ob etwas Untypisches oder Unerwünschtes im Netzwerkverhalten vorhanden ist.

Dabei gibt es zwei generelle Einsätze für Anomaly Detection: One-Class Learning und Density Estimation.

One-Class Learning – Classifier erstellt eine kompakte Beschreibung, die den gewünschten prozentuellen Anteil (z.B. 95%) von „normal“ Netwerktraffic deckt, der Rest wird als anomal betrachtet.

Dichteschätzung (Density Estimation) – das beobachtete System wird als die Sammlung von Werten modelliert. Der Zustand mit geringer Wahrscheinlichkeit wird als anomal betrachtet.

Bei der Anomaly Detection sollen das „level of analysis“ und Kontrollvariablen ausgewählt werden, außerdem sollen die Daten von Sensoren interpretiert und aufsummiert werden. Da die Anomalie auf einem Ebene nicht erkennbar sein könnte, müssen eventuell die Daten von mehreren Ebenen noch aufsummiert werden. Z.B. Worm kann auf einem Computer nicht erkannt werden, dagegen wenn wir Daten von mehreren Computer betrachten, kann untypisches Traffic erkannt werden.

Eine weitere Aufgabe ist die falsche und wiederholte Alarme auszufiltern, dafür muss eventuell Methoden des überwachten Lernens angewendet werden. Z.B. ein Teil der Anomalien kann unwichtig oder ungefährlich sein.

Fehlereingrenzung erfordert globales Wissen innerhalb der Knowledge Plane. Z.B. ein Netzwerkroute ist überlastet: die Netzwerküberlastung kann lokal z.B. an jedem Router festgestellt werden, dagegen kann die Ermittlung von der Höhe und den Grenzen von der Überlastung für die Gesamtroute erst global erfolgen.

Bei der nächsten Aktivität dem Diagnosis werden die Quellen des Problems vermutet. Also werden die Quellen von dem unerwünschten Verhalten gesucht. Dabei können sowohl die bekannten Probleme erkannt werden, die früher von Operator aufgelistet wurden, als auch die neuen Probleme charakterisiert werden, bei denen nur einige Teile bekannt sind. Normalerweise folgt „Diagnosis“ der Fehlereingrenzung, aber man kann auch die Fehler in System vermuten ohne genau Fehlerquelle zu kennen. Genau wie bei „Anomaly detection“ können auch hier die Methoden des überwachten Lernens angewendet werden.

Sowohl Fehlereingrenzung als auch Diagnosis erfordern aktive Messungen. Dabei muss das Balance zwischen Messkosten und Informationswert erhalten werden.

Fault Isolation und Diagnosis erfordern auch das Systemmodell, das diagnostiziert wird. Um so ein Modell zu erstellen können die Methoden aus dem maschinellen Lernen wie Learning for Interpretation and Understanding angewendet werden.

Nachdem das Problem diagnostiziert wurde, können die Methoden des überwachten Lernens angewendet werden, um den Netzwerkoperator zu unterstützen oder Repairstrategie zu teilen.

4.3 Abwehr von Angriffen und Worms

Die Aufgaben zum Schutz eines Netzwerks vor Angriffen und Worms kann man so aufteilen wie sie normalerweise von Netzwerkmanager durchgeführt werden: Prävention (Prevention Tasks), Erkennung (Detection Tasks), Erwiderung und Wiederherstellung (Response and Recovery Tasks).

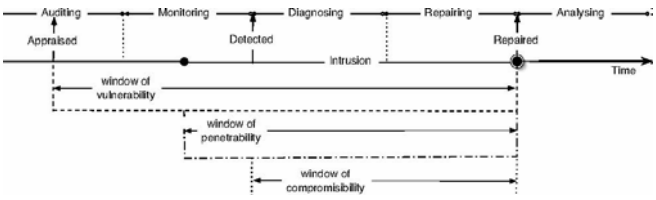


Bild 3. Abwehr von Angriffen und Worms[2]

Das Ziel des Netzwerksmanagers, dass die drei zeitlichen Abschnitte (window of vulnerability, penetrability, compromisibility) zu einem zeitlichen Punkt konvergieren. Die Aufgaben für die Abwehr von Angriffen unterscheiden sich unwesentlich von dem Wiederherstellen nach einem kritischen Fehler.

4.3.1 Prävention (Prevention Tasks)

Netzwerkadministrator strebt the window of vulnerability zu minimieren (die Zeitspanne zwischen dem Zeitpunkt, als eine Schwachstelle bekannt geworden ist und dem Anwenden von einem Patch oder einer neuen Konfiguration). Die Grundstrategie für dieses Ziel ist den Gefährdungsgrad zu minimieren (z. Beispiel Ausschalten von unnötigen Diensten) und ständige Beobachtung von neu aufgedeckten Schwachstellen um sie möglichst früh in eigenem System festzustellen.

Dafür werden Scan tools wie Nessus, Satan oder Oval benutzt. Da es immer neue Sicherheitsrisiken und Softwarefehler entdeckt werden, sollte das benutzte Tool immer aktuell gehalten werden. Wenn die Schwachstelle entdeckt wurde und kein Patch dafür

[18] Mitchell, 1985

[19] Zweben, 1994

[20] Zhang and Dietterich, 1995

[21] Boyan and Moore, 2000

existieret, musste Entscheidung getroffen werden, ob der betroffene Service ausgeschaltet werden kann, dabei wird das Risiko und die Bedienungsqualität beachtet.

Letztendlich überwacht der Netzwerkadministrator das System, um die Verhaltensmuster vor und nach dem Eindringen vergleichen zu können.

4.3.2 Erkennung (Detection Tasks)

Die Aufgabe des Netzwerksmanagements ist window of penetrability (die Zeitspanne zwischen dem Zeitpunkt, in dem das Computersystem aufgebrochen und dem Moment, wenn das Computersystem vollständig repariert worden ist) möglichst klein zu halten. Die korrekte Diagnostik erlaubt dem Netzwerkmanager entsprechend zu reagieren. Dabei soll das Gleichgewicht zwischen der Qualität und Schnelligkeit erhalten werden.

Je früher das Eindringen erkannt wird, desto mehr Chance gibt es unautorisiertes Benutzen oder Missbrauch des Systems zu verhindern. Deswegen versucht der Netzwerkadministrator das System zu überwachen, indem er Protokolle, Alarmsignale beobachtet. Jeder, der einmal ein Netzwerk administriert hat, weiß, dass man mit Informationen und Fehlalarmen überschwemmt wird. Aus diesem Grund werden die Netzwerkgeräte so konfiguriert, dass die Anzahl von Falschalarmen akzeptabel gehalten wird, was folglich das Risiko von Nichterkennen von Eindringung erhöht.

4.3.3 Erwiderung und Wiederherstellung (Response and Recovery Tasks)

Als die Diagnostik das Eindringen entdeckt hat, muss der Netzwerkmanager entsprechend handeln. Die Aufgabe des Netzwerksmanagements ist window of compromisibility (die Zeitspanne zwischen dem Zeitpunkt, in dem das Eindringen entdeckt worden ist und dem Moment, wenn das Computersystem entsprechend reagiert hat) zu reduzieren, indem er die automatic intrusion response systems einsetzt. Leider sind diese Systeme heutzutage nicht in der Lage sogar bei manuell gesteuerten Schutzmaßnahmen zu unterstützen. Aus diesem Grund erstellen die Netzwerkmanagers entsprechende Prozeduren für jede einzelne Angriffsart.

Eine Antwort auf einen Angriff kann aus dem Schließen von Benutzerprozessen über Blockieren des Benutzerskontos, Blockieren der IP- Adresse bis zum Ausschalten des Netzwerks bestehen. Da bei Wiederherstellungsmaßnahmen (damage recovery bzw. repairing) die Funktionalität des Systems erhalten werden muss, sind solche Maßnahmen schwer zu automatisieren. Die Reaktion sollte den Einfluss auf das Gesamtsystem minimieren (zum Beispiel nicht alle Netzwertsports schließen wenn das Blockieren von einem einzigen IP ausreichend ist).

Nachdem der Angriff abgewehrt worden ist, sollte Angriffsursachen und –folgen mit vorhandenen Protokollen analysiert und dokumentiert werden.

5. Fazit

In diesem Vortrag haben wir die Methoden und Aufgaben des maschinellen Lernen, sowie auch theoretische und bereits existierende Algorithmen betrachtet. Danach haben wir einige, aber natürlich nicht alle, durch die Knowledge Plane zu lösenden Aufgaben diskutiert wie Rapid Configuration of Network, Anomaly detection, Responding to intruders and Worms.

Obwohl auf dem Gebiet der Automatischen Konfiguration viel erforscht wird und mehrere Algorithmen z.B. aus maschinellem Lernen angewendet werden, befindet sich das Problem noch in der Frühphase der Entwicklung. Für die Erstellung eines Autonomes Netzwerk- oder Computersystems werden noch weitere Forschungen sowohl konzeptueller als auch experimenteller Art benötigt.

6. Literatur

[1] Pat Langley, John E. Laird, Seth Rogers - Cognitive Architectures: Research Issues and Challenges

[2] Tom Dietterich, Pat Langley - Machine Learning for Cognitive Networks: Technology Assessment and Research Challenges

Interaktion in intelligenten Gebäuden

Seminar Innovative Internettechnologien und Mobilkommunikation WS2008

Sebastian Klepper
Lehrstuhl Netzarchitekturen und Netzdienste
Fakultät für Informatik
Technische Universität München
Email: sebastian.klepper@mytum.de

***Zusammenfassung* — Die Bedeutung intelligenter Systeme bei der Unterstützung alltäglicher Aufgaben nimmt rapide zu. Dabei tritt das System nicht als greifbare Maschine in Erscheinung, sondern wird zunehmend in die Umgebung des Benutzers eingebettet, zum Beispiel in Fahrzeuge, Kleidung oder Gebäude.**
Diese Arbeit beschäftigt sich mit der Interaktion zwischen intelligenten Systemen, welche in Gebäude integriert sind, und ihren Benutzern. Zunächst werden wichtige Aspekte erörtert, die in dieser Hinsicht bei der Entwicklung solcher Systeme zu berücksichtigen sind. Im Anschluss daran werden Herausforderungen betrachtet, die bei ihrer Umsetzung auftreten. Abschließend wird eine Zusammenfassung der erarbeiteten Prinzipien und ein Ausblick über die weitere Entwicklung in diesem Bereich gegeben.

***Schlüsselworte* — Benutzerschnittstelle, Mensch-Maschine-Interaktion, Smart Home, Home Automation, Ubiquitous Computing, Künstliche Intelligenz**

1. Einleitung

Sowohl im häuslichen als auch im betrieblichen Umfeld werden immer häufiger automatisierte Systeme eingesetzt, um Menschen den Alltag angenehmer zu gestalten und Arbeit abzunehmen.
Im häuslichen Umfeld können beispielsweise sogenannte „Home Automation“ (HA) Systeme die Steuerung von Heizung, Beleuchtung und Gartenbewässerung übernehmen und so gleichzeitig Ressourcen sparen.
Im betrieblichen Umfeld können intelligente Systeme bei Meetings assistieren, multimediale Kommunikation erleichtern und durch Benutzeridentifizierung betriebliche Zugriffs- und Zugangsrechte kontrollieren.

Automatisierung bedeutet jedoch nicht, dass diese Systeme autonom Entscheidungen treffen und Handlungen durchführen. Dies wäre aber gerade wünschenswert, denn ein automatisiertes System mit hohem Betriebsaufwand (z. B. konstante Überwachung des Systemverhaltens) ersetzt die Arbeit, die es dem Menschen abnehme soll, lediglich durch eine neue Aufgabe. Es gibt also außer der Automatisierung noch andere Schlüsselfaktoren, durch die intelligente Systeme ihren eigentlichen Zweck erfüllen können.

Eines der zentralen Probleme ist die Interaktion mit dem Menschen: Die in dieser Arbeit behandelten Systeme arbeiten nicht vom Menschen abgeschottet, sondern wirken durch ihr Verhalten auf dessen Umfeld ein. Somit befinden sich beide Parteien zwangsläufig in einer ständigen Interaktion.

Um das Systemverhalten kontrollierbar und berechenbar zu gestalten, müssen bei der Entwicklung der Systeme diverse Aspekte beachtet werden. Einige wichtige dieser Aspekte sollen im Folgenden betrachtet werden, gefolgt von den Schwierigkeiten ihrer Umsetzung und möglichen Lösungen.

2. Aspekte bei der Entwicklung intelligenter Systeme

Bei der Interaktion zwischen Mensch und Maschine sollen Interessenskonflikte, Missverständnisse und Fehlverhalten vermieden werden. Der Erfolg einer Interaktion zwischen Mensch und Maschine hängt demnach im Wesentlichen vom gegenseitigen Verständnis ab, dieses wiederum von der Qualität der Kommunikation, die während der Interaktion stattfindet.

Die Interaktion muss von einem Informationsaustausch begleitet werden: Das System benötigt Information darüber, was der Benutzer in einer spezifischen Situation erwartet, und es muss Einwände oder Bestätigungen des Benutzers entgegennehmen können. Der Benutzer hingegen muss über Handlungen des Systems informiert werden und die vorangegangenen Entscheidungen nachvollziehen können. Zudem müssen ihm die Grenzen des Systems und seine eigenen Handlungsmöglichkeiten aufgezeigt werden. [2] Warum das alles nötig ist, wird im Laufe dieses Kapitels deutlich werden.

Zunächst soll die allgemeine Funktionsweise eines automatisierten Systems betrachtet werden:

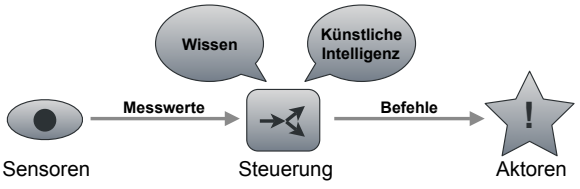


Abbildung 1. Funktionsweise eines automatisierten Systems.

Abbildung 1 zeigt den Informationsfluss innerhalb eines solchen Systems. Umgebungsparameter werden von Sensoren erfasst und an eine Steuerungseinheit übergeben. Diese wertet die Daten aus und versucht daraus den aktuellen Zustand der Umgebung abzuleiten. Dieser Zustand wird auf Handlungsbedarf überprüft dann gegebenenfalls eine Strategie entwickelt,

wie Anpassungen vorgenommen werden können. Befehle an sogenannte Aktoren sorgen für die Durchführung der Strategie. Aktoren sind durch das System ansprechbare Geräte, die Einfluss auf die Umwelt nehmen können, z. B. Motoren inkl. Steuerungseinheit.

Zur Veranschaulichung in den nächsten Abschnitten soll ein Beispiel dienen: Ein Home Automation System ist für das Raumklima in einem Gebäude verantwortlich. Zur Erfassung von Umgebungsdaten stehen sowohl innen als auch außen Sensoren für Lufttemperatur und -feuchtigkeit zur Verfügung. Die Anpassung des Raumklimas erfolgt dann nicht durch eine Klimaanlage, sondern schlicht durch öffnen und schließen von Fenstern. [2]

2.1 Notwendigkeit von Transparenz

Wie man sieht, kann sich das System nur in starker Abhängigkeit von Sensordaten ein Bild von seiner Umgebung machen. Zur Analyse und Entscheidungsfindung wird zwar noch anderes Wissen hinzugezogen, z. B. festgelegte Entscheidungsregeln oder Erfahrungswerte, doch die Sensordaten bilden die Grundlage für einen erkannten Zustand, einen daraus identifizierten Handlungsbedarf, dementsprechend getroffene Entscheidungen und letztendlich für Handlungen des Systems.

Diese Abhängigkeit stellt eine große Schwachstelle dar, denn durch defekte Sensoren können Messfehler auftreten oder Daten gänzlich fehlen. Außerdem unterliegen Sensoren immer einer gewissen Messungenauigkeit und manche Umgebungsparameter werden nicht berücksichtigt, evtl. weil sie gar nicht gemessen werden können. So kann ein unvollständiges oder inkorrektes Bild der Umgebung entstehen und das System trifft falsche Entscheidungen, obwohl es in seinen Augen richtig handelt. [2]

Im Beispielsystem könnte das bedeuten, dass Fenster zum Lüften geöffnet werden, obwohl draußen ein Unwetter herrscht. Das beschriebene System erfasst nämlich weder Windstärke und -richtung, noch ist es in der Lage, Regen zu erkennen, weil hierfür keine entsprechenden Sensoren vorhanden sind. Aber auch wenn diese nachgerüstet werden, kann es z. B. vorkommen, dass die Fenster nicht geöffnet werden sollen, weil sich Allergiker in den Räumen aufhalten und draußen starker Pollenflug herrscht.

Solche, teilweise unvorhergesehen auftretenden Situationen erkennen zu können würde einen enormen technischen Aufwand bedeuten. Neben einer immer umfangreicheren Konstellation von Sensoren müssten auch die Bestandteile der Steuerungseinheit (Wissensdatenbank, Algorithmen, etc.) umfangreicher werden, um Defizite bei der Datenerfassung durch künstliche Intelligenz auszugleichen.

a) Entscheidungen des Systems:
Das System wird immer komplexer und damit für den Benutzer eine undurchschaubare „Black Box“. Er kann einerseits nicht nachvollziehen, warum das System so handelt, wie es handelt, und verliert die Kontrolle über Teile seines Alltags. Gleichzeitig steigen die Erwartungen an das System mit unerklärlichen, aber vermeintlich grenzenlosen Fähigkeiten. Da dies nicht der Realität entspricht, werden diese falschen Erwartungen früher oder später enttäuscht, wenn das System an seine Grenzen stößt. Andererseits hat der Benutzer auch keine Möglichkeit, Fehlentscheidungen des Systems zu korrigieren oder generell zu verhindern, wenn er nicht weiß, wie es zu diesen Entscheidungen gekommen ist. [2]

Beides hat Frustration zur Folge, gefolgt von unnötig hohem Aufwand um den Fehler zu finden und zu beseitigen. Die Konfiguration eines Systems gestaltet sich um einiges aufwändiger, wenn man die Auswirkung von getroffenen Einstellungen nicht im Verhalten des Systems wiederfinden kann.

b) Handlungen des Systems:
Solange der Benutzer die Entscheidungen des Systems nicht nachvollziehen kann, wird für ihn auch die Handlungsweise unerklärlich bleiben. Aber auch hier ist Transparenz erforderlich, denn der Benutzer muss Handlungen des Systems erwarten können, d. h. in bestimmten Situationen mit ihnen rechnen. Tritt das System unerwartet oder unangemessen in Aktion, wird das den Benutzer unter Umständen stören oder sogar verwirren und seine eigenen Aktivitäten unterbrechen, was natürlich nicht wünschenswert ist.
Zudem ist es dem Benutzer, wenn er nicht überrascht sondern ausreichend informiert wird, besser möglich, die Situation selbst zu erfassen und den Handlungen des Systems (ggf. stillschweigend) zuzustimmen oder einzugreifen.

2.2 Interaktion in Form von Unterstützung

Es wird deutlich, dass eine alternative Form der Interaktion von Mensch und Maschine wünschenswert wäre. Bei dieser Form sollte das System den Menschen unterstützen statt ihn bzw. seine Umgebung selbsttätig zu kontrollieren. Dafür ist Technologie erforderlich, die folgende Eigenschaften besitzt:

Sie ist auf subtile Art und Weise „allgegenwärtig“ (Ubiquitous Computing).

Sie versorgt den Benutzer zur richtigen Zeit am richtigen Ort mit wichtigen Informationen.

Sie teilt Entscheidungen nachvollziehbar mit und gestaltet Handlungen für den Benutzer kontrollierbar.

Sie handelt somit nicht bevormundend, sondern hilft dem Benutzer Vorgänge zu verstehen und daraus zu lernen.

Nebenbei sei bemerkt, dass durch diese Herangehensweise ein als „Graceful Degradation“ bekanntes Phänomen ermöglicht wird. Dabei reagiert ein System auf Fehler oder andere unerwartete Ereignisse nicht mit einem Ausfall, sondern reduziert in angemessener Weise Teile seiner Funktionalität oder deren Qualität.

Es gibt nun zwei denkbare Formen eines solchen unterstützenden Systems: Es kann dem Benutzer Vorschläge machen statt von selbst zu handeln oder sogar versuchen, dem Benutzer durch diese Handlungsweise etwas beizubringen. [2]

Generell sollte dem Benutzer aber auch die Möglichkeit geboten werden, selbst zu bestimmten, in welchem Umfang er an Entscheidungen beteiligt und über Handlungen in Kenntnis gesetzt werden möchte. Schließlich soll durch das System Arbeit wegfallen und nicht durch neue Aufgaben ersetzt oder sogar vermehrt werden.

a) Vorschlagendes System:
Ein vorschlagendes System arbeitet anfangs wie das zuvor beschriebene System. Es versucht, aus eingehenden Daten den momentanen Zustand der Umwelt abzuleiten und Handlungsbedarf zu identifizieren, anschließend entwickelt es eine oder mehrere Strategien zur Anpassung.

Es führt die seiner Meinung nach nötigen Maßnahmen aber nicht eigenständig durch, sondern tritt mit Vorschlägen an den Benutzer heran, welcher dann eine der Situation angemessene Strategie auswählt.

In der Praxis könnte das so aussehen: Der Benutzer wird drüber informiert, dass es sinnvoll sein könnte zu lüften. Fenster, die das System gerne öffnen würde, werden durch Beleuchtung o. Ä. visuell hervorgehoben und der Benutzer darauf hingewiesen. Der Benutzer entscheidet nun, ob und welche Fenster geöffnet werden. Die Arbeit des Fensteröffnens kann nun von ihm selbst durchgeführt, oder automatisch erledigt werden.

Man beachte, dass hier nicht nur von einer Strategie die Rede ist, sondern von mehreren. Da der Benutzer im Gegensatz zum System nicht den beschriebenen Beschränkungen unterliegt, kann er deutlich mehr Kontekt berücksichtigen.
Im Bezug auf das Beispielsystem bedeutet das: Der Benutzer kann zwar keine quantitative Messung der Klimafaktoren (Temperatur, Luftfeuchtigkeit) vornehmen, er kann sie aber immerhin qualitativ einschätzen. Zudem kann er weitere Faktoren wie Wetterlage oder Pollenflug berücksichtigen.
Das System muss sich somit nicht mehr auf eine Strategie festlegen, sondern kann dem Benutzer mehrere Alternativen anbieten und ihm die Entscheidung überlassen. Dadurch wird das Risiko von Fehlentscheidungen drastisch verringert.

b) Lehrendes System:
Wenn das System dazu in der Lage ist, dem Benutzer eine bestimmte Handlungsweise näher zu bringen, ohne ihn zu überfordern, dann ergibt sich unter Umständen ein Lerneffekt. Wie viel und wie leicht der Benutzer lernen kann, hängt davon ab, wie viel Aufschluss ihm das System über Motivation, Details und Konsequenzen der durchgeführten Maßnahmen gewährt. Gerade im Moment der Handlung oder direkt danach lässt sich dieser Lerneffekt am besten nutzen: Statt passiv zu beobachten, muss der Benutzer selbst denken und evtl. sogar selbst handeln, dabei kann er Zusammenhänge erkennen, seine Umgebung verstehen und lernen, sie zu kontrollieren.

Zudem wird einer unnötigen Abhängigkeit von maschineller Unterstützung vorgebeugt. Diese würde sonst immer weiter zunehmen – eine Problematik, die heute schon in manchen Bereichen des täglichen Lebens unübersehbar geworden ist. Ein automatisch handelndes System lernt unter Umständen selbst aus den Konsequenzen seiner Handlungen, es verschließt sich aber vor dem Benutzer und gibt ihm nicht die Chance, auch etwas zu lernen. Auch Synergieeffekte, z. B. dadurch dass Benutzer und System neu erworbenes Wissen austauschen, werden nicht genutzt. [2]

2.3 Momente der Interaktion

Betrachtet man ein Szenario wie die Klimakontrolle im Beispiel genauer, erkennt man mehrere entscheidende Momente. Diese ermöglichen jeweils ein gewisses Maß an Benutzerinvolvierung bzw. erfordern sie sogar, wenn dem Prinzip der Unterstützung gefolgt werden soll.

Abbildung 2 zeigt den zeitlichen Zusammenhang dieser wichtigen Momente:

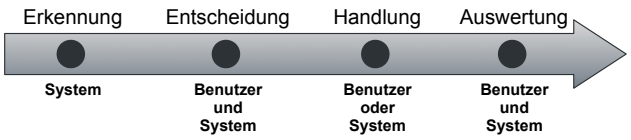


Abbildung 2. Zeitlicher Verlauf eines Szenarios mit Interaktion.

a) Erkennung:
Zu diesem Zeitpunkt wurde aufgrund von gesammelten Daten und bestehendem Wissen ein bestimmter Zustand festgestellt und eventuell auch schon ein Handlungsbedarf erkannt. Maßnahmen zur Anpassung müssen ausgewählt werden. Den Benutzer hier schon einzubeziehen hat wenig Sinn, denn er kann mit den gesammelten Daten wenig anfangen, die Auswertung muss demnach vom System vorgenommen werden.
Aber auch wenn bereits der Zustand der Umgebung identifiziert wurde, sollte der Benutzer weder die Erkennung von Handlungsbedarf noch die Auswahl von entsprechenden Maßnahmen übernehmen müssen. Gerade diese Arbeiten sollen den Menschen abgenommen werden, dementsprechend muss das System zu diesem Zeitpunkt noch keine Kommunikation zum Benutzer aufbauen, sondern kann sich auf die Zustandserkennung und Strategieentwicklung konzentrieren. [2]

Im Beispielszenario liegen hier zunächst die Sensordaten vor. Das System erkennt nun erstens, dass das Raumklima nicht optimal ist (Zustand mit Handlungsbedarf), und zweitens, dass es durch Lüften verbessert werden könnte (Strategie). Die Strategie wird nun im Detail entwickelt, d. h. es wird geprüft, welche Fenster für einen optimalen Luftaustausch geöffnet werden müssten – hierbei können sich mehrere Möglichkeiten ergeben.

b) Entscheidung:
Im Moment der Entscheidung kommt es darauf an, eine oder

mehrere Maßnahmen bzw. Strategien auszuwählen und damit deren Umsetzung einzuleiten. Für ein unterstützendes System ist es genau jetzt essentiell, dem Benutzer Einblick in das Geschehen zu gewähren und ihm die Alternativen aufzuzeigen, da er mehr Kontext berücksichtigen und somit evtl. bessere Entscheidungen treffen kann. [2]

Im Beispielszenario wird dem Benutzer nun mitgeteilt, dass es sinnvoll wäre zu lüften und welche Fenster geöffnet werden könnten. Der Benutzer kann sich nun gänzlich gegen das Lüften entscheiden, eine der vorgeschlagenen Möglichkeiten auswählen, oder eine Strategie anpassen, indem er z. B. einzelne Fenster ausschließt oder hinzufügt.

c) Handlung:
Die Strategie, um die Umgebungsbedingungen den Wünschen des Benutzers anzupassen, wurde nun festgelegt und muss möglichst bald durchgeführt werden, um einen optimalen Effekt zu erzielen. Denn wenn zwischen dem Moment der Erkennung und der Handlung zu viel Zeit vergeht, haben sich wichtige Parameter evtl. schon wieder verändert und die geplanten Maßnahmen ist nicht mehr angemessen oder sogar kontraproduktiv.
Gerade deshalb sollte die Involvierung des Benutzers wohlüberlegt stattfinden, denn dabei kann es zu den größten Verzögerungen können. Einerseits kann und möchte ein Mensch keine komplexen Gedankengänge in möglichst kurzer Zeit vollziehen. Erst recht nicht, wenn dazu Berechnungen nötig sind.
Andererseits kann es natürlich auch sein, dass der Benutzer andere Aktivitäten abschließen möchte, bevor er sich mit der Angelegenheit befasst. Vielleicht vergisst er in der Zwischenzeit sogar, dass er eine Entscheidung treffen soll – hier darf das System nicht blind auf eine Eingabe warten, sondern sollte den Benutzer z. B. an den Entscheidungsbedarf erinnern. Zuvor sollte es aber aus den bereits erwähnten Gründen überprüfen, ob überhaupt noch Handlungsbedarf besteht. [2]

Im Beispielszenario wurde nun bereits beschlossen, dass bestimmte Fenster geöffnet werden sollen. Der Benutzer kann nun das System diese Arbeit durchführen lassen. Wenn der Benutzer die Fenster aber selbst öffnet, muss das System diese Handlung erkennen oder sie muss ihm mitgeteilt werden, damit das Szenario erfolgreich abgeschlossen werden kann.

d) Auswertung:
Wenn die Handlung abgeschlossen ist, bedeutet das nämlich weder zwangsläufig das Ende des Szenarios, noch der Interaktion. Insbesondere der Zeitpunkt „Ende des Szenarios“ kann sich aus der Sicht von System und Benutzer unterscheiden.
Im Falle eines vorschlagenden Systems endet das Szenario für den Benutzer meist mit dem Abschluss der Handlung, da auch keine weitere Interaktion mit dem System stattfindet. Das System kann jedoch im Hintergrund die Auswirkungen der getroffenen Maßnahmen beobachten und analysieren, um den Erfolg der gewählten Strategie zu bewerten.
Im Falle eines lehrenden Systems wird der Benutzer allerdings auch in dieser Phase involviert und zumindest über festgestellte Konsequenzen, evtl. aber auch detailliert über Zusammenhänge einzelner Maßnahmen und der erzielten Veränderung informiert. So kann auch der Benutzer den Erfolg der getroffenen Entscheidung bewerten und für die Zukunft Schlüsse daraus ziehen. [2]

Im Beispielszenario beobachtet das System noch eine Weile den Lüftungsvorgang und schließt die Fenster nach gegebener Zeit wieder. (Falls dies nicht festgelegt wurde, könnte sich hier eine weitere Entscheidungssituation ergeben!) Der Benutzer wird zum Schluss über den Erfolg informiert, d. h. über die erreichte Verbesserung des Raumklimas. Wie bereits erwähnt ist ein gemeinsames Lernen mit dem System möglich, indem erworbenes Wissen ausgetauscht wird und zukünftig bessere Strategien ausgewählt werden können – System und Benutzer profitieren beide davon, wenn Lerneffekte genutzt werden.

2.4 Präsentation von Information

Offensichtlich ist es fast zu jedem Zeitpunkt wichtig, dass dem Benutzer verschiedene Informationen vermittelt werden. Dazu gehören Möglichkeiten, Grenzen und Status des Systems, sowie Auskunft über Kontrollmöglichkeiten. Dabei spielt es natürlich eine große Rolle, wie, wann und wo die Information präsentiert wird, denn man möchte diese Basis der Kommunikation so effizient und effektiv wie möglich gestalten.

a) Ort:
Wenn Information präsentiert werden soll, ist eine der wichtigsten Fragen die des am besten geeigneten Ortes. Es bieten sich auf Anhieb mehrere Möglichkeiten an, die je nach Situation und zu präsentierender Information verschiedene Vor- und Nachteile haben.

i) Überall:
Die einfachste, aber nicht gerade subtilste Methode wäre zunächst einmal, die Information „überall“ zu präsentieren. Die kann z. B. durch weiträumig hörbare Sprachausgabe geschehen, oder durch Anzeige auf möglichst vielen Displays an möglichst vielen Orten. Dadurch können mehrere Personen gleichzeitig mit Information versorgt werden, was aber auch ein Nachteil sein kann, etwa weil Unbetroffene in ihren Aktivitäten gestört werden.

ii) Betroffener Benutzer:
Eine gezieltere Vorgehensweise wäre es, die Information nur bei dem betroffenen Benutzer zu präsentieren. Das kann zum Beispiel durch Senden einer Meldung an tragbare Geräte wie PDAs oder Mobiltelefone geschehen. Diese Variante ist vor allem bei zeitnah zu präsentierenden Informationen zu bevorzugen, da man davon ausgehen kann, dass der Benutzer sie sofort wahrnimmt. Natürlich muss hier zuerst entschieden werden, für welche Benutzer die Information überhaupt relevant ist.

iii) Betroffenes Subjekt:
Möchte man hingegen den Betreff der Information betonen, so kann die Information auch bei dem „Subjekt“ der Angelegenheit, etwa auf dem Gegenstand selbst oder einem Display in der Nähe, präsentiert werden.

b) Zeitpunkt:
Genauso wichtig wie die Wahl des Ortes und offensichtlich teilweise eng damit verbunden ist die Wahl des Zeitpunktes, zu dem die Information präsentiert werden soll.

i) Sofort:
Auch hier gibt es eine vergleichsweise simple Methode, die Information wird sofort präsentiert, was durchaus sinnvoll sein kann, z. B. wenn Handlungsbedarf erkannt worden ist und eine Entscheidung ansteht. Für eine sofortige Präsentation muss natürlich sichergestellt sein, dass man den oder die betroffenen Benutzer auch zeitnah erreicht – wie man gesehen hat, sind dafür nicht alle Varianten bei der Ortswahl geeignet.

ii) Festgelegter Zeitpunkt:
Eine weitere Möglichkeit wäre es, den Benutzer selbst festlegen zu lassen, wann er mit (einer bestimmten Art von) Information versorgt werden möchte. Das kann zum Beispiel bei oft oder regelmäßig auftretenden Ereignissen sinnvoll sein, über die der Benutzer immer zur gleichen Zeit in Form einer Zusammenfassung in Kenntnis gesetzt werden möchte.

iii) Ermittelter Zeitpunkt:
Die wohl aufwändigste, aber auch interessanteste Variante ist es, das System den optimalen Zeitpunkt für die Übermittlung der Information an den Benutzer ermitteln zu lassen. Hierbei können eine Reihe von Faktoren berücksichtigt werden, es ist allerdings wieder ein möglichst vollständiges Abbild der Realität erforderlich, um einen geeigneten Zeitpunkt finden zu können.
Interessant ist diese Vorgehensweise zum Beispiel, wenn das System weiß, dass der Benutzer gerade mit etwas wichtigem beschäftigt ist – etwa weil er sich in einer Besprechung befindet. Das System kann nun diesen Umstand bei der Wahl des Zeitpunktes in Betracht ziehen, das Ende der Aktivität abwarten und ihm so die Information zum nächstmöglichen Zeitpunkt zukommen lassen, ohne ihn vorher gestört zu haben.

Ein weiteres Problem in diesem Zusammenhang ist der Umgang mit gleichzeitig auftretenden Ereignissen, denn hier können sich Ansprüche an das selbe Informationsmedium überschneiden. Dieses Problem kann gelöst werden, indem man die Informationen sequentiell ausgibt, wobei ihnen eine Priorität zugeordnet werden muss. Falls Informationen eine gewisse Gemeinsamkeit aufweisen, können sie aber auch zusammengefasst werden – auch innerhalb einer Sequenz von vielen, aber klassifizierbaren Informationen.

c) Medium:
Unabhängig davon, wo und wann man die Information präsentiert, hat man in einem modernen Umfeld verschiedene Medien zur Verfügung.

i) Auditiv:
Wenn Information möglichst ortsungebunden präsentiert werden soll, z. B. an mehreren Orten oder für mehrere Personen gleichzeitig, dann bietet sich eine Ausgabe von Sprache oder anderen akustischen Signalen (vor allem Signaltöne oder Melodien mit hohem Wiedererkennungswert) an. Bei zu umfangreichen bzw. komplexen Informationen überfordert diese Variante aber schnell und ist damit eher ungeeignet.

ii) Visuell:
Gerade bei umfangreichen, komplexen Informationen ist eine sichtbare Darstellung wohl die beste Variante, wie ein bekanntes Sprichwort bestätigt. Zusammenhänge können in visualisierter Form besser verstanden und Informationen in beliebiger Reihenfolge aufgenommen werden, was bei reiner Sprachausgabe nicht möglich ist.

iii) Audio-Visuell:
Man kann natürlich auch die Vorteile beider Varianten kombinieren, indem man dieselbe Information sowohl optisch als auch akustisch vermittelt. Man kann aber auch jeweils unterschiedliche Aspekte der Information präsentieren und z. B. durch eine weiträumig hörbare Meldung auf einen Sachverhalt aufmerksam machen und interessierten Personen dann an einem bestimmten Ort weitere Details und Kontrollmöglichkeiten bieten.

d) Detailgrad:
Zu guter Letzt hat man noch die Möglichkeit, das Informationsangebot unterschiedlich detailliert zu gestalten. Der richtige Detailgrad ist auf der einen Seite abhängig von der Situation und der Information, die vermittelt werden soll. Auf der anderen Seite beeinflusst aber auch die Wahl von Ort, Zeit und Medium den optimalen Detailgrad.

i) Ausführlich:
Eine ausführliche Präsentation kann erforderlich sein, wenn ein komplizierter Sachverhalt erklärt werden muss, z. B. weil der Benutzer eine diesbezügliche Entscheidung treffen soll. Stehen ihm nur wenige Informationen zur Verfügung, erhöht sich das Risiko einer Fehlentscheidung. Auch für die optimale Nutzung des Lerneffekts kann es dienlich sein, wenn man dem Benutzer detaillierte Informationen über festgestellte Auswirkungen oder neu gewonnene Erkenntnisse zukommen lässt.

ii) Zusammengefasst:
Eine Zusammenfassung hingegen bietet sich überall dort an, wo das gewählte Medium keine ausführliche Darstellung der Information zulässt. Es kann aber auch sinnvoll sein, Informationen auf das Wesentliche zu reduzieren, oder mehrere gleichartige Informationen zusammenzufassen. Beides dient der Erhöhung von Übersichtlichkeit und Verständlichkeit.

Abbildung 3 fasst noch einmal zusammen, welche Kombinationen bei der Wahl von Präsentationsort, -zeitpunkt, -medium und -detailgrad empehlenswert sind:

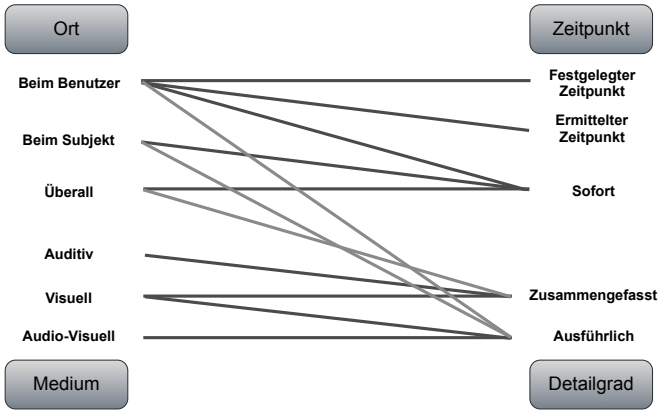


Abbildung 3. Zweckmäßige Kombinationen bei der Präsentation von Information.

3. Herausforderungen bei der Umsetzung

Möchte man die soeben vorgestellten Prinzipien bei der Entwicklung oder Implementierung eines entsprechenden Systems, z. B. für Home Automation, beherzigen, so steht man in der Praxis vor einer Reihe von Herausforderungen, die es zu bewältigen gilt.

Für den Benutzer reicht es nicht aus, vom System über Entscheidungen informiert zu werden oder Handlungsvorschläge zu erhalten. Er muss die Funktionsweise der eingesetzten Technologie verstehen können und wissen, wann und auf welche Weise er mit dem System interagieren kann.
Das System muss unabhängig von Aufgabenstellung oder Gerätekonstellation zuverlässig arbeiten und darf die Arbeit, die es dem Benutzer abnimmt, nicht durch Administrationsaufwand ersetzen.

Um Interessenskonflikte, Missverständnisse und Fehlverhalten erfolgreich zu vermeiden, genügt es infolgedessen nicht, die in Kapitel 2 behandelten Aspekte bei der Systementwicklung zu berücksichtigen. Es muss zusätzlich Augenmerk auf eine sorgfältige und gesamtheitliche Lösung der folgenden Problemstellungen gelegt werden.

3.1 Komplexität der Technologie

Man könnte annehmen, dass jedes Problem mit Hilfe hochentwickelter Technologie lösbar ist. Wie in Abschnitt 2.1 gezeigt, verhindert diese Herangehensweise allerdings das Erreichen eines der wichtigsten Ziele: die Transparenz des Systems für den Benutzer.

Wenn man Technologien miteinander kombiniert und dann unter Umständen auch noch in ein Umfeld integriert, das nicht dafür ausgelegt ist, diese Technologien zu beherbergen, so entspricht das tatsächliche Verhalten einzelner Komponenten oder des Gesamtsystems manchmal nicht den Erwartungen des Benutzers. Fehlverhalten, wenn auch nur in den Augen des Benutzers, führt bei ihm zu Frustration und unnötig hohem Arbeitsaufwand zum Auffinden und Beheben der Fehler. Jedoch stellen sich, spätestens beim Auftreten einer solchen Situation, mehrere Fragen über den Benutzer: [3]

Welche Informationen benötigt er über das System, um es verstehen zu können? Welche davon muss ihm das System selbst liefern?

Wie kann er erkennen, welche Geräte auf welche Art und Weise interagieren?

Was sind die tatsächlichen Grenzen des Systems?

Welche Konstellationen und Konfigurationen der vorhandenen Geräte sind möglich? Welche davon sind sinnvoll?

Wo und wie interagiert er mit einem System, dass sich nicht an einem bestimmten Ort befindet, sondern „allgegenwärtig“ ist? (→ „Ubiquitos Computing“)

Wie kann der Benutzer einzelne Geräte oder das Gesamtsystem kontrollieren?

Bevor der Benutzer lernen kann mit einer Technologie umzugehen, benötigt er ein Modell von ihrer Funktionsweise. Für den Ingenieur eines Geräts oder Systems ist nun die Überlegung erforderlich, wie und mit welchem Ergebnis sich der Benutzer ein solches Modell bilden kann.
Will man erreichen, dass der Benutzer mit hoher Wahrscheinlichkeit problemlos mit dem System interagieren kann, so muss man hier entsprechende Modelle vorbereiten und dem Benutzer von Anfang an vermitteln. Je verständlicher und intuitiver dies geschieht, desto steiler ist die Lernkurve. (Gemeint ist die akademische Definition, die das erworbene Wissen über der Zeit betrachtet und bei der ein steiler Anstieg einen erfolgreichen Lernvorgang kennzeichnet.)

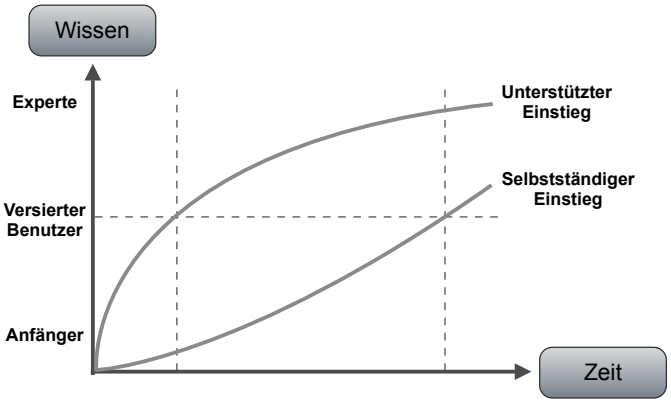


Abbildung 4. Unterschiedlicher Verlauf von unterstütztem und selbstständigem Einstieg in die Benutzung eines Systems.

Abbildung 4 verdeutlicht diesen Zusammenhang, indem der Einstieg zweier Benutzer in das selbe System verglichen wird. Ein Benutzer wird dabei unterstützt, z. B. durch Dokumentation, integrierte Hilfefunktion und Assistenten, während der andere sich das Verständnis über die Funktionsweise des

Systems selbst aneignen muss.
Wie deutlich zu erkennen ist, erlangt der unterstützte Benutzer nach der gleichen Zeit mehr Wissen über das System, d. h. er kommt besser damit zurecht. Der auf sich allein gestellte Benutzer benötigt deutlich mehr Zeit, um einen ähnlichen Wissensstand zu erlangen – er wird öfter mit für ihn unlösbaren Problemen konfrontiert werden und den Umgang mit dem System als anstrengend und frustrierend empfinden.

3.2 Neue Modelle von Technologie

Eine neu aufgetretene Problematik und damit auch ein gutes Beispiel für die Notwendigkeit neuer Modelle ist die Entwicklung von Konnektivität.
Das sogenannte „explizite“ Modell Konnektivität, wie es Kabel repräsentieren, beinhaltet sichtbare und damit einfach zu kontrollierende Verbindungen, die sich nicht unbemerkt ändern können. Mit zunehmender Verbreitung werden Kabel aber durch drahtlose Verbindungen ersetzt. Neben den praktischen Vorteilen bringt diese neue Art von Konnektivität aber auch ein neues, „implizites“ Modell mit sich: Verbindungen sind nicht sichtbar und damit auch nicht ohne weiteres zu kontrollieren. Sie können unbemerkt abbrechen und sich, evtl. in anderer Art und Weise als beabsichtigt, wieder aufbauen. [3]
Dabei gibt es verschiedene Standards für drahtlose Verbindungen, die unterschiedliche Verhaltensweisen aufweisen. Das reicht von schnellen Ad-hoc-Verbindungen (z. B. Infrarot) über mäßig konfigurierbare Varianten (z. B. Bluetooth) bis hin zu komplexen, sicherheitsbewussten Verbindungsarten (z. B. Wireless LAN).

Wird ein Benutzer, dem das „implizite“ Modell von Konnektivität unbekannt ist, mit dieser Technologie konfrontiert, ergeben sich daraus mehrere Probleme. Er kann nicht erkennen, welche Geräte theoretisch miteinander in Verbindung stehen können, und auch nicht, welche Geräte tatsächlich miteinander kommunizieren.
Auch neue Sicherheitsrisiken müssen berücksichtigt werden: während die Manipulation einer Kabelverbindung mehr oder weniger direkten Zugriff erforderte, können nun Dritte auch aus größerer Entfernung Zugang zu einem Netzwerk aus drahtlosen Verbindungen erlangen.
Die Technologie bleibt für den Benutzer so lange undurchsichtig, bis ihm das zugrunde liegende Modell vermittelt wird. Erst dann kann er die Technologie verstehen und in vollem Umfang nutzen. [3]

3.3 Interoperabilität innerhalb des Systems

Die Technologielandschaft in einem modernen Gebäude ist meist häufigen Änderungen unterworfen – gerade im häuslichen Bereich, wo keine Vorgaben regulierend wirken können. Es kommt immer wieder neue Hard- und Software in das System, wird entfernt, aufgerüstet oder aktualisiert. Zudem existieren Geräte verschiedener Hersteller und Generationen nebeneinander, was ein hohes Maß an Kompatibilität voraussetzt. Ist diese Kompatibilität nicht gegeben, werden Flexibilität, Zuverlässigkeit, Transparenz und Kontrollierbarkeit des Systems erheblich eingeschränkt. [3]

Interoperabilität bezeichnet nun die Fähigkeit eines Gerätes, mit anderen Geräten eine Verbindung einzugehen und zu interagieren. Ohne diese Fähigkeit entstehen innerhalb der immer unübersichtlicher werdenden Technologielandschaft Gerätekonstellationen, die man als „Inseln der Funktionalität“ bezeichnen könnte. Sie bestehen beispielsweise aus Geräten eines einzigen Herstellers, welche untereinander kommunizieren können, mit herstellerfremden Geräten allerdings nicht. Meist kommt noch ein Aspekt hinzu: für diese Fähigkeit soll vorher möglichst kein Konfigurationsaufwand anfallen. Diese erweiterte Form wird als „Spontane Interoperbilität“ bezeichnet. [3]

Standards bieten hier nur einen vermeintlichen Ausweg, denn sie legen meist nur den Syntax der stattfindenden Kommunikation fest. Die Semantik muss vom Entwickler hinzugefügt werden, was zu individuellen Interpretationen von ein und derselben Kommunikationsweise führen kann. Außerdem sind auch Standards manchmal proprietär und werden daher nicht von allen erhältlichen Geräten unterstützt.
Die Lösung ist auch hier, neue Modelle für Kommunikation und Interoperabilität zu finden, welche Protokolle, Schnittstellen sowie syntaktische und semantische Standards bereitstellen. Sie existieren noch nicht in etablierter Form, sondern befinden sich noch in der Entwicklungsphase, wobei verschiedene Hersteller oder Gruppierungen jeweils eigene Wege gehen (z. B. in Stanford, bei HP mit „CoolTown“, bei Xerox mit „Speakeasy“). [3]

3.4 Zuverlässigkeit und Bedienbarkeit

Die angesprochene Veränderlichkeit der Technologielandschaft birgt auch ein gewisses Risiko von Instabilität. Während bei Desktop- oder Laptop-Computern häufige Abstürze oder von Installationen verursachte Neustarts immer noch in Kauf genommen werden, werden an „Haushaltsgeräte“ wie Fernseher, DVD-Playern oder Pay-TV Decodern hohe Erwartungen gestellt. Hier würde niemand akzeptieren, dass der Betrieb durch unerwünschte Ereignisse unterbrochen wird. Auch Geräte mit komplexer Elektronik müssen zuverlässig funktionieren und sich nahtlos in die bestehende Umgebung einfügen. [3]

Der Grund dafür ist, dass diese Anforderung von eingebetteten Systemen (Embedded Computing) im Gegensatz zu PC-Systemen erfüllt wird, ist folgender: Da die Wartung solcher Geräte meist umständlich ist und sie selten über einen Internetzugang verfügen, über den Updates oder Patches automatisch eingespielt werden könnten, muss sich der Hersteller bereits im Voraus Gedanken über eventuell eintretende Ereignisse (Störungen, Fehler, Ausfälle) machen und den Umgang mit ihnen bei der Entwicklung des Gerätes berücksichtigen. [3]

Die Stabilität muss somit in der Systemarchitektur verankert sein, was eine entsprechende technologische Herangehensweise und Entwicklungskultur erforderlich macht. Zudem muss ein als „Graceful Degradation“ bezeichnetes Systemverhalten ermöglicht werden, d. h. das System reagiert auf einen Teilausfall, indem es schrittweise Funktionalität oder deren Qualität einschränkt und so versucht, weitgehend unbeeinträchtigt zu bleiben. [3]

3.5 Administration und Wartung

Wie bereits behandelt, unterliegt die Technologielandschaft in einem modernen Gebäude stetigen Veränderungen und steigender Komplexität. Die Veränderungen (Hinzufügen, entfernen und aktualisieren von Hard- und Software) müssen von jemandem durchgeführt werden, sofern sie nicht automatisch ablaufen (was ohnehin nur bei Software möglich ist). Dabei muss der Überblick behalten und es müssen gleichzeitig weitere relevante Aspekte, z. B. Sicherheit, beachtet werden.

Viele Benutzer haben nun weder die Fähigkeit, diese Administrationsaufgaben zu übernehmen, noch das Interesse daran. Im betrieblichen Umfeld mag speziell für diese Aufgaben ausgebildetes Personal vorhanden sein, im häuslichen Umfeld ist jedoch davon auszugehen, dass es keinen Systemadministrator gibt. [3] Das bedeutet, dass man Systeme schaffen muss, bei denen sich der Aufwand für Installation, Wartung und Betrieb in minimalen Grenzen hält.

Es gibt bereits zwei bewährte Modelle von quasi wartungsfreien Systemen: [3]

a) Eigenständigkeit:
Eigenständige Geräte weisen eine hohe Zuverlässigkeit auf, können aber durch ihre Komplexität bei Bedarf nur durch Fachkräfte repariert werden. Solche Geräte können auch untereinander Interaktionen eingehen, ohne dass die Bedienbarkeit darunter leidet. Ein Beispiel für diese Variante wären Fernsehgeräte.

b) Nutzung eines Netzwerks:
Netzwerke nutzende Geräte stellen das Endgerät für ein System dar, dessen Komplexität in diesem Netzwerk verborgen liegt. Der Benutzer kann somit von Verbesserungen profitieren, das tatsächlich bei ihm vorhandene System bleibt aber von Änderungen verschont. Ein Beispiel hierfür wäre das Telefon.

Es gibt selbstverständlich auch Mischformen, wie z. B. Set-Top-Boxen. Diese Geräte beinhalten selbst ein hohes Maß an Komplexität und beziehen zusätzlich Funktionalität aus dem (Kabel-)Netzwerk, beides unter Einhaltung hoher Zuverlässigkeit und einfacher Bedienbarkeit. [3]

3.6 Künstliche Intelligenz

Zu guter Letzt soll noch ein Kernproblem bei Mensch-Maschine-Interaktionen aufgezeigt werden: Wie schafft es das System, intelligent auf seine Umwelt zu reagieren und dabei die Interessen seines Benutzers zu berücksichtigen?

Häufig wird der Mensch bei der Benutzung von maschinellen Assistenten in bestimmte Verhaltensmuster gezwungen, weil der Assistent sonst nicht in der Lage ist, dessen Absichten zu erkennen. Diese Einschränkung soll aber möglichst gering gehalten werden, deshalb muss das System nicht nur den Zustand seiner Umwelt möglichst realitätsnah zu erfassen, sondern es muss auch in der Lage sein, aus diesem Zustand die Absichten des Benutzers abzuleiten. Inwiefern dabei auf künstliche Intelligenz zurückgegriffen wird, kann unterschiedlich stark ausgeprägt sein, was anhand eines Beispielszenarios veranschaulicht werden soll: Mehrere Personen nehmen an einem Meeting teil, das von einem intelligenten System als solches erkannt und unterstützt werden soll. [3]

Das System kann nun in einem ersten Schritt die Bedeutung verschiedener Sensordaten interpretieren, um den Zustand der Umgebung festzustellen. Beispielsweise werden mehrere RFID-Tags, die von den Personen getragen werden, in einem kleinen Umkreis erkannt und das System leitet daraus ab, dass sich mehrere Personen im selben Raum aufhalten.

Anschließend kann das System mehrere Faktoren (z. B. erfasste Sensordaten und vorhandenes Wissen) aggregieren, um ein genaueres Bild der Umwelt zu erhalten. Beispielsweise wird die zuvor erkannte Gruppenbildung mit dem Gebäudeplan abgeglichen und das System erkennt, dass sich die Personengruppe in einem Besprechungsraum befindet, und schließt daraus, dass es sich um ein Meeting handelt.

Das System kann nun versuchen, aus dem erkannten Zustand die Absichten des Benutzers abzuleiten, indem es Wissen (z. B. Regeln und Erfahrungswerte) anwendet. Beispielsweise ist es üblich, dass Meetingteilnehmer sich Notizen machen und diese miteinander teilen, oder dass jemand eine Präsentation hält.

Wenn das System eine in der vorliegenden Situation wahrscheinliche Absicht erkannt hat, kann es nun versuchen, in unterstützender Art und Weise darauf zu reagieren. Dies kann sofort geschehen, oder aber erst, wenn der Benutzer durch eine bestimmte Handlung die Vermutung des Systems bestätigt hat. Beispielsweise stellt das System sofort ein gesichertes, drahtloses Netzwerk zur Verfügung, in dem Daten ausgetauscht werden können. Oder es unterstützt eine Präsentation, indem es den Raum abdunkelt, sobald ein Projektor eingeschaltet wird.

Abbildung 5 verdeutlicht diese Zusammenhänge von Messungen bzw. Vermutungen des Systems und den daraus resultierenden Handlungen. Man kann hier wieder eine Abhängigkeit von Sensordaten als Ausgangspunkt des Systemverhaltens feststellen. Wie schon mehrfach erwähnt, besteht hier das Risiko von Messfehlern und Irrtümern. Und unabhängig davon, wie viel Aufwand man betreibt, um Sensordefizite durch künstliche Intelligenz auszugleichen, stellt sich immer die Frage, ob die von den Sensoren gelieferten Daten auch

tatsächlich den Zustand der Umwelt oder nur den Zustand der Sensoren reflektieren. Hier scheint der Einsatz von Systemen mit künstlicher Intelligenz und Lernfähigkeit erforderlich zu werden.

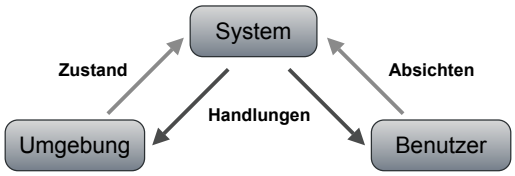


Abbildung 5. Zusammenhänge beim Einsatz von künstlicher Intelligenz.

4. Zusammenfassung und Ausblick

Ein System, das ein Gebäude zu einem intelligenten Gebäude machen soll, muss dabei offensichtlich viele Aspekte berücksichtigen, um seine Benutzer auf optimale Weise unterstützen zu können:

Es sollte sich für den Benutzer nachvollziehbar verhalten, in dem es über Status, Entscheidungen und Handlungen informiert. Dazu sollte es dem Benutzer die richtigen Informationen zur richtigen Zeit am richtigen Ort und in der richtigen Form präsentieren.
Weiterhin sollte es dem Benutzer die übergeordnete Kontrolle über Entscheidungen und Handlungen überlassen, ihm dabei aber auch die Möglichkeit geben, festzulegen, wie viel Handlungsfreiheit er dem System zugestehen möchte.
Zudem sollte das System auch noch möglichst zuverlässig sein, indem es möglichst fehlerfrei funktioniert, einfach zu warten und zu erweitern ist und eine einwandfreie Kommunikation und Interaktion aller Geräte untereinander ermöglicht.
Zu guter Letzt sollte sich das System seiner Umwelt und der Interessen seiner Benutzer bewusst sein, Ungenauigkeit und Fehleranfälligkeit bei Messungen durch künstliche Intelligenz ausgleichen und auf Unklarheiten flexibel reagieren.

Doch auch der Benutzer sollte seinen Teil dazu beitragen, wenn er problemlos mit dem System interagieren können möchte. Dazu sollte er sich auf eine Interaktion mit dem System einlassen und sich ggf. neue Modelle von (neuer oder bekannter) Technologie aneignen, um sein Gegenüber überhaupt verstehen zu können. Schließlich sollte er seine Erwartungen an das System der Realität anpassen, was wiederum ein Verständnis für dessen Funktionsweise voraussetzt.

Ausblickend lässt sich feststellen, dass sich die bereits vorhandenen Formen solcher intelligenten und unterstützenden Systeme in die richtige Richtung entwickeln.
Mehrere wissenschaftliche Institute, aber auch Hersteller kommerzieller Systeme erforschen neue Möglichkeiten, wie man die vorgestellten Aspekte bei der Entwicklung intelligenter Systeme optimal umsetzen kann. Man ist allerdings noch lange nicht bei dem Potential angelangt ist, das erforderlich wäre, um die Zukunftsvisionen auf diesem Gebiet wirklich in die Realität umsetzen zu können.

Um wirklich autonome und nicht nur automatisierte Systeme zu verwirklichen, muss sich die dafür benötigte Technologie in allen Bereichen noch gewaltige Fortschritte machen. Es lässt sich aber feststellen, dass die sich Forschung auf dem richtigen Weg befindet.

Literatur

[1] Neng-Shian Liang, Li-Fhen Fu, Chao-Lin Wu, *An integrated, flexible, and Internet- based control architecture for home automation system in the Internet Era* Proceedings of the 2002 IEEE International Conference on Robotics & Automation, Washington, DC, 2002.

[2] Stephen S. Intille, *Designing a Home of the Future* IEEE: Pervasive Computing, 2002.

[3] W. Keith Edwards & Rebecca E. Grinter, *At Home with Ubiquitous Computing: Seven Challenges* Computer Science Laboratory, Xerox Palo Alto Research Center.

[4] S. Dobson, S. Denazis, A. Fernandez, d. Gaiti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt, F. Zambonelli, *A Survey of Autonomic Communications* ACM Transactions on Autonomous and Adaptive Systems, Vol. 1, No. 2, December 2006.

[5] A. R. Al-Alo & M. Al-Rousan, *Java-Based Home Automation System* IEEE Transactions on Consumer Electronics, Vol. 50, No. 2, May 2004.

[6] Randall Davis, Howard Shrobe & Peter Szolovits, *What Is a Knowledge Representation?* AAAI Articles, Spring 1993.

3GPP Long Term Evolution (LTE)

Krisna Haryantho
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
haryantk@in.tum.de

Kurzfassung
Bei dieser Arbeit werden die Basisanforderungen der 3GPP UTRAN Long Term Evolution (LTE) Spezifikation für Mobilfunk-Telekommunikation diskutiert. Die grundlegende Einführung in die wichtigen Technologien, wie OFDMA und MIMO, die LTE unterstützen, werden erbenfalls behandelt. Das Kernnetzwerk (SAE), die Protokolle, sowie die ganzen Prozesse auf der physikalischen Sicht werden hier nicht diskutiert.

Schlüsselwörter
3GPP, Mobilfunk, OFDMA, MIMO, SC-FDMA, E-UTRAN, UMTS.

1. EINLEITUNG
Um eine erfolgreiche Kommunikation zwischen zwei oder mehreren Instanzen, benötigt man einen Standard, der von allen Kommunikationsteilnehmern akzeptiert und verstanden wurde. Für die Mensch-Mensch Kommunikation gilt beispielsweise das Sprechen schon längst als allgemein gültiger Standard. Der Standard beschreibt genau, wie eine erfolgreiche Kommunikation errichtet werden kann. Für das Sprechen gelten der Mund und die Ohren als Sende- bzw. Empfangsapparat, die Luft als Transportmittel, die Sprache als Informationscodierung, usw.

In der Mobilfunktelefonie gibt es den 2G *Global System for Mobile Communactions* (GSM) Standard, der eine globale Kommunikation zwischen Mobilfunkgeräten ermöglicht, solange ein GSM Netz verfügbar ist. GSM wurde später erweitert durch *Global Packet Radio Access* (GPRS), der eine höhere Datenrate hat, und einen erweiterten Service bietet, wie MMS oder WAP.

Der frühere 2G Standard wurde für Sprachverkehr gedacht. Die spätere Entwicklung unterstützt auch Datenverkehr mit relativ geringer Datenrate (noch geeignet für SMS, WAP). Die heutigen Anwendungen wie z.B. Videotelefonie und Internet setzen eine größere Datenrate voraus. Der 2G Standard ist deswegen nicht mehr geeignet für diese Zwecke.

Dies im Hintergrund definiert das *3rd Generation Partnership Project* (3GPP) eine Reihe von Radio Technologien, die Flexibilität für alle Servicearten (z.B. Sprach- und Datenverkehr) bietet. Diese sind bekannt unter dem Namen UMTS (*Universal Mobile Telecommunications Services*). UMTS basiert auf WCDMA (*Wideband Code Division Multiple Access*) und wurde später in HSDPA (*High Speed Downlink Packet Access*) weiter entwickelt, der stellt eine größere Downlinkdatenrate zur Verfügung. Die Uplinkdatenrate wurde durch die Einführung von HSUPA (*High Speed Uplink Packet Access*) gesteigert. HSDPA und HSUPA bilden zusammen HSPA (*High Speed Packet Access*)

LTE ist auch ein Radiostandard neben den oben genannten Technologien. Sie enthalten eine Reihe von Spezifikationen für ein neues paket-orientiertes Radionetzwerk und wurde unabhängig von HSPA entwickelt. Die ersten Spezifikationen für LTE wurden Ende 2007 veröffentlicht.

Von der höchsten Ebene betrachtet lässt sich das 3G Mobilfunk-Netzwerk in drei Elemente zerlegen. Diese sind *User Equipment* (UE), *Radio Access Network* (RAN) und *Core Network* (CN). Die UEs sind z.B. Mobiltelefone, Laptop (mit dem entsprechenden Benutzer-Identifikationsmodul), usw. und diese bauen eine Kommunikation über die Luftschnittstelle mit Elementen im RAN. Im UMTS Fall besteht das RAN aus NodeBs und Radio Network Controllers (RNC).

Während der RNC für die meisten Radio-Funktionalitäten (wie z.B. Resourcenallokation, Scheduling) verantwortlich ist, übernimmt das NodeB die Aufgabe einer Basisstation, die hauptsächlich die physikalische Verbindung mit den UE gewährleistet. Ein NodeB ist genau mit einem RNC verbunden. Es kann aber zur Optimierung der Systemperformanz auch mit einem oder mit mehreren NodeBs verbunden werden. Die NodeBs stehen in einer n:1 Beziehung zu den RNCs. Es kann aber auch mehrere RNCs in einem System geben.

Die RNCs kommunizieren miteinander über die Iu-Schnitstelle. Alle RNCs werden dann direkt an das Kernnetzwerk (CN). angeschlossen. Das CN stellt Dienste wie IP-Paketdienste, Abrechnungsdienste, Mobilitätsmanagement, Session-Management.

Im LTE Fall besteht das RAN (das RAN bei LTE heißt *Evolved UMTS Terrestrial Radio Access Network*/E-UTRAN) nur noch aus einem logischen Element, nämlich das eNodeB. Grob gesagt übernimmt das eNodeB gleich die Funktionalitäten des NodeBs und des RNCs. Die eNodeBs sind mit SAE (*System Architecture Evolution*) verbunden, dem Kern-Netzwerk von LTE.

Der Schwerpunkt dieser Arbeit ist die Luftschnittstelle zwischen dem eNodeB und UE. Im zweiten Abschnitt werden einige wichtige Anforderungen, die in der LTE Spezifikation verlangt werden, diskutiert. Im dritten Abschnitt werden die Mehrfachzugriffsverfahren für LTE besprochen, sowohl für Downlink- als auch für Uplink-Übertragungen. Im vierten Abschnitt wird Mehr-Antennen-Verfahren vorgestellt, das die Performanz von LTE nochmal verbessert.

2. HINTERGRUND & ANFORDERUNGEN
„3GPP ist eine weltweite Kooperation von Standardisierungs–gremien für die Standardisierung im Mobilfunk“ [10]. Seit der Entstehung im Jahr 1998, hat 3GPP zahlreiche Spezifikationen für ein global einheitliches (im Sinne von Technologie und Standard) Mobilfunksystem gemacht. Die von 3GPP veröffentlichten Standards sind chronologisch: Release 99 (das erste UMTS Netzwerk mit WCDMA), Release 5 (HSDPA), Release 6 (HSUPA), Release 7 (HSPA+). Der Name LTE wurde dem 3GPP Release 8 vergeben.

Das Ziel von LTE ist die Verbesserung der Spektraleffizienz, der Dienste, und der Interoperabilität mit anderen bestehenden Standards (GSM, UMTS, usw.), sowie eine Senkung der Betriebskosten (operative Kosten) [9]. Dafür hat 3GPP eine Liste von Anforderungen zusammengesetzt, die in [1] zu erfahren sind. Die wichtigen Anforderungen sind unter anderen:

- Spitzendatenrate von 50/100 Mbps für jeweils Uplink/ Downlink. Uplink bezeichnet man die Datenübertragung vom UE zur Basisstation, Downlink die von der Basisstation zum UE. Mit Spitzendatenrate meint man die maximal erreichbare Nutzdatenrate.
- Server-UE *Round Trip Time* (RTT) von geringer als 30ms und Zugriffsverzögerung von geringer als 300ms.
- Verbesserte Energieeffizienz.
- Flexibilität für Bandbreitenallokation mit 1.4, 5, 10, 20 MHz Spektrum.
- Bessere Spektraleffizienz, d.h. mehr Bits pro Sekunde pro Hz Frequenzband.
- Möglichkeit für Zusammenbetrieb mit älteren Systemen (z.B. UMTS, GSM) sowie mit anderen nicht von 3GPP spezifizierten Systemen
- Höhere Mobilität und Sicherheit
- Höhere Kapazität im Vergleich zu HSDPA/HSUPA
- Unterstützung für ein gepaartes und nicht-gepaartes Spektrum

Die Spitzendatenrate hängt linear von der Spektrumallokation ab. Je breiter das allokierte Spektrum, desto größer ist die resultierende Datenrate. Die Spitzendatenrate kann auch von der Anzahl der Sende- und Empfangsantenne abhängen. Die Spezifikation geht von der Benutzung von zwei Empfangsantennen und einer Sendeantenne jeweils für Downlink und Uplink aus. Dies wird im Abschnitt 4 genauer behandelt.

Mit dieser Konfiguration soll ein LTE System mit 20 MHz Downlink-Spektrum-Allokation eine Downlink-Spitzendatenrate von 100Mbps; das sind 5 bps/Hz. Zum Vergleich, WCDMA Systeme verwenden ein 5 MHz Spektrum und haben eine Downlink - Spitzendatenrate von 384 Kpbs, das ist 0,00768 bps/Hz. HSDPA Systeme mit 5 MHz Spektrum können eine Downlink-Spitzendatenrate von 14,4 Mbps erreichen, das ist 2,88 bps/Hz.

Im Uplink soll das LTE System unter 20 MHz Uplink-Spektrum-Allokation eine Uplink-Spitzendatenrate von 50 Mpbs erreichen; das ist 2,5 bps/Hz. WCDMA bzw. HSUPA Systeme haben eine Spitzendatenrate von 384 Kbps bzw. 5,7 Mbps. Beide verwenden ein 5 MHz Spektrum und schaffen somit eine Spektraleffizienz von 0,0768 bps/Hz bzw. 1,14 bps/Hz.

Da die Datenrate linear von der Spektrumbreite abhängt, egal welche Spektrumallokation (z.B. 5, 10 oder 20 MHz) man nimmt, ist das LTE System immer spektraleffizienter als WCDMA und HSPA.

Die Effizienz der Energieverbrauch (Strom) ist ein sehr wichtiger Punkt, besonders beim UE. Eine verbesserte Energieeffizienz kann erreicht werden, in dem man die Ressourcen (hier das Spektrum) effektiv ausnutzt. Z.B. wenn die Sendenantenne aktiv ist, dann sendet sie immer mit dem besten Modulationsverfahren für die entsprechende Kanalkondition. Mehr dazu wird im nächsten Abschnitt diskutiert.

Für die Mobilfunkanbieter bietet LTE eine flexiblere Spektrum-Allokation in dem Sinne, dass ein Anbieter frei wählen kann, wie breit die Bandbreite für sein LTE System sein wird. Die möglichen Allokationen sind 1.4, 3, 10, 15, 20 MHz. Der Anbieter hat auch die Möglichkeit, die Allokation später zu vergrößern. Da der komplette Umstieg auf LTE kostenintensiv ist und es noch einige Zeiten dauert, bis das ganze Land mit LTE Service bedeckt wird, kann beispielsweise ein Mobilfunkanbieter, der ein 20 MHz Band hat, in der Einführungsphase erst ein 5 MHz Band für LTE zuteilen. Die drei restlichen 5 MHz Bänder kann er für HSPA oder HSPA+ verwenden. In der späteren Phasen kann er stufenmäßig seine Bandbreitenallokation für LTE vergrößern. So dient HSPA+ als Übergang von UMTS zu LTE. Dies ist aus wirtschaftlichen Gründen günstiger als wenn er gleich in der Einführungsphase das komplette 20 MHz Band für LTE allokieren bzw. ganz auf HSPA+ verzichten würde.

3. MEHRFACHZUGRIFFSVERFAHREN

3GPP hat für LTE OFDMA im Downlink und SC-FDMA im Uplink als Mehrfachzugriffsverfahren gewählt. OFDMA ist eigentlich schon ein bekanntes Verfahren, das z.B. in WLAN und DVB-T angewandt wird. Es war damals technisch schwierig, OFDMA in Mobilfunknetzwerke effektiv einzusetzen und deshalb hat man WCDMA als Mehrfachzugriffsverfahren für UMTS gewählt.

Einige wichtige Gründe für die Wahl von OFDMA in LTE sind [4] :

- *Größere Bandbreite und Bandbreitenflexibilität*. LTE will aber eine hohe Datenrate unter Verwendung von großer Bandbreite von bis zu 20MHz erzielen. Mit steigender Bandbreite bleiben die OFDMA Subträger noch orthogonal zueinander, während WCDMA Systeme ihre Leistung verlieren. In UMTS verwendet man deshalb 5 MHz Bandbreite.
- *Flache Architektur*. Diese ist gekennzeichnet durch die Belegung aller Radio-Funktionalitäten inklusive das Paket- und Frequenzdomäne-Scheduling in der Basisstation (eNodeB). Das Scheduling der Frequenzdomäne, das die Zellkapazität bis zu 50% vergrößern kann, kann nicht in CDMA gemacht werden.
- *Energieeffiziente Uplink-Übertragung mit SC-FDMA*. Siehe Abschnitt 3.2.

3.1 Downlink-Übertragung

Im folgenden werden die OFDMA Grundprinzipien erklärt. Zur Verständnishilfe werden zunächst zwei sehr bekannte und einfachere Mehrfachzugriffsverfahren vorgestellt, FDMA und TDMA

3.1.1 FDMA

FDMA steht für *Frequeny Division Multiple Access*. Wie der Name sagt, werden bei FDMA die verfügbare Frequenzbänder in kleinere Frequenzbänder zerlegt. Diese werden als Träger bezeichnet. Im trivialen Fall sind die Träger gleich groß. Jeder Nutzer bekommt einen von diesen Träger zugeteilt. Somit sendet und empfängt ein Nutzer mittels dem Träger, der ihm vorher zugeteilt worden ist.

Der Vorteil von diesem Verfahren ist, dass die Nutzer gleichzeitig senden oder empfangen können ohne auf die anderen zu warten, da jeder Nutzer seinen eigenen Träger hat.

Dies ist aber auch gleich ein Nachteil, denn es ist nicht sinnvoll für jeden Nutzer immer die gleich große Bandbreite zuzuteilen. Es scheint zwar als eine faire Verteilung, aber nicht alle Nutzer benötigen immer zu einem Zeitpunkt dieselbe Bandbreite. Außerdem sind bei einem einfachen FDMA System die Ressourcenzuteilungen (die Ressourcen sind die viele kleine Frequenzbänder) relativ starr, d.h. wenn die verfügbaren Frequenzenbänder schon einmal zugeteilt werden, dann muss jeder Nutzer immer denselben Träger benutzen.

Bei guter Kanalkondition stellt dies kein Problem dar. Sobald aber eine Störung auf diesem Kanal auftaucht, wird eine Datenübertragung über dieser Kanal nicht optimal für den betroffenen Nutzer. Für einen anderen Nutzer könnte derselbe Kanal allerdings optimal sein. Wenn man die Resourcenzuteilung in FDMA System effizienter und flexibler macht, hat das System noch bessere Leistung.

3.1.2 TDMA

TDMA steht für *Time Division Multiple Access*. Analog zu FDMA wird bei TDMA die Zeit zugeteilt. Ein Nutzer bekommt einen festen Zeitschlitz zugeteilt und darf nur innerhalb dieses Zeitschlitzes senden oder empfangen. In jedem Zeitschlitz darf nur ein Nutzer senden oder empfangen. In TDMA Systemen sendet und empfängt ein Nutzer mit der vollen Systembandbreite.

Der Vorteil ist, dass man keine Frequenzsynchronization machen muss, was meistens sehr aufwändig ist. Die Zeitsynchronisation dagegen ist mit heutiger Hardware sehr genau und einfach. Der Kritikpunkt bei TDMA ist aber die Verzögerung der Zugriffszeit und die uneffiziente Resourcenverteilung (die Ressourcen sind die Zeitschlitze).

Wenn alle Nutzer immer aktiv am Senden oder Empfangen sind, so mag es kein Problem sein, denn die gesamte Datenrate bleibt erhalten. Wenn aber ein oder mehrere Nutzer nichts senden oder empfangen, so senkt er die gesamte Datenrate des Systems, denn er gibt seinen Zeitschlitz nicht ab, was für den anderen Nutzer mit höherem Ressourcenbedarf eine große Bedeutung haben könnte.

3.1.3 OFDMA

OFDMA steht für *Orthogonal Frequency Division Multiple Access*. Die Basisidee ist die beiden vorherigen Verfahren sehr geschickt zu kombinieren. In Abbildung 1 sieht man sowohl Zeitschlitze als auch Frequenzschlitze, dargestellt als Rechtecke.

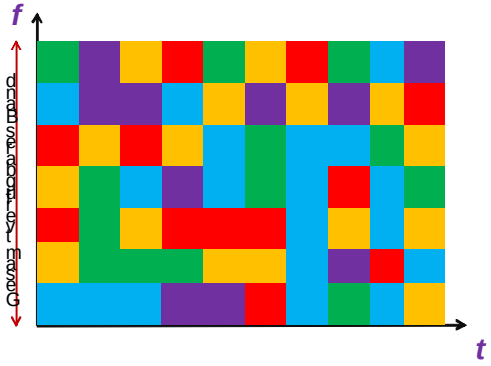


Abbildung 1: Resourcenzuteilung in OFDMA. Farben repräsentieren verschiedene Nutzer

Die Kisten werden in LTE als *Resource Block* (RB) bezeichnet. Ein RB besteht aus 12 Subträgern und ist eine Millisekunde lang.

Ein Subträger dient zur Übertragung von Symbolen. Symbole sind in der Nachrichtentechnik Modulationszeichen, die Bitfolgen repräsentieren/modellieren. Die Anzahl der Bits, die durch ein Symbol repräsentiert werden können, hängt von dem Modulationsverfahren ab. Es muss nicht für alle Subträger immer dasselbe Modulationsverfahren angewandt werden.

In OFDMA wird das gesamte Frequenzband ähnlich wie FDMA in viele sehr kleine Subträger zerlegt.

Der obere Teil von Abbildung 2 zeigt die Frequenzzerlegung i, reinen FDMA System, das untere zeigt das von OFDMA. Ein wichtiger Punkt in OFDMA ist, dass die Subträger zu einander orthogonal stehen. D.h. der Abtastpunkt von jedem Subträger ist immer beim Nullgang der anderen Subträger. Somit können Intersymbol-Interferenzen vermieden werden. Das führt anschließend zu einer besseren Symbolrückgewinnung/erkennung auf der Empfängerseite. Außerdem können die Subträger durch diese Eigenschaft näher aneinander positioniert werden. Man beachte den Bandbreitengewinn im Vergleich zum reinen FDMA System. In LTE beträgt die Entfernung zwischen zwei benachbarten Subträgern 15 KHz[3].

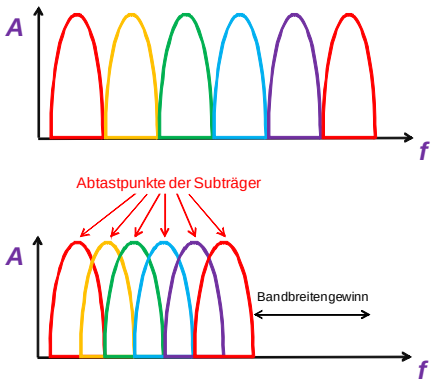


Abbildung 2: Orthogonalität der Subträger

Wie vorhin schon gesagt, besteht ein RB aus 12 Subträger (das entspricht 180KHz Bandbreite) und ist eine Millisekunde lang.

Die Mehrwegeausbreitung zeigt das Phänomen, dass ein Signal sich verbreitet und über mehrere Wege ans Ziel ankommt. Es ist durchaus möglich, dass das gleiches Signal mehrmals am Empfänger ankommt. Das kann Intersymbolstörungen verursachen, wo das Empfangen von dem nächsten Symbol durch das verzögerte Wiederempfangen von dem früheren gestört wird.

Eine Besonderheit bei OFDMA ist die Verwendung von Zyklischem Präfix, um Intersymbolstörungen wegen Mehrwegeausbreitung zu bekämpfen. Ein zyklisches Präfix ist Teil eines Symbols (normalerweise der Fußteil eines Symbols), der kopiert und am Kopf jenes Symbols angehängt wird. Ein zyklische Präfix soll die Rolle eines Schutzintervalls übernehmen. Die Länge des zyklischen Präfixes muss größer als die maximal mögliche Mehrwegeausbreitungsverzögerung sein, damit eine Intersymbolstörungen verhindert werden kann. Auf der Empfängerseite wird das zyklische Präfix ignoriert. Die LTE Spezifikation definiert zwei Varianten des zyklischen Präfixes, das normale und erweiterte zyklische Präfix. Die beiden unterscheiden sich nur in ihrer Länge/ Duration (in Mikrosekunde).

Ein RB hat die Dauer von 14 Symbolen (eine Millisekunde), wenn das normale zyklische Präfix verwendet wird. Kommt das erweiterte zyklische Präfix in Frage, so reduziert sich die Dauer auf 12 Symbole. Die Anzahl der Bits, die ein Symbol repräsentieren, hängt von dem angewandten Modulationsverfahren ab, wie in Tabelle 1 zu sehen ist. Man spricht hier von der Ordnung des Modulationsverfahrens. Je höher die Ordnungszahl ist, desto mehr Bits kann ein Modulationssymbol repräsentieren.

Tabelle 1:Modulationsverfahren

Modulationsverfahren	*Bits pro Symbol*
PSK	**1**
QPSK	**2**
16QAM	**4**

64QAM	**6**

Jedem Nutzer wird nach seinem Bedarf und nach seinem Kanalkondition RB zugeteilt. Ein Scheduler im eNodeB ist für diese Aufgabe zuständig. Der Scheduler beurteilt anhand eines Feedback-Signals die Kanalkondition für jeden Nutzer, und versucht immer den besten nutzer-spezifischen Kanal für ihn zu benutzen. Dies ist möglicherweise ein Kanal mit keiner oder sehr geringer Störung.

Modulationsverfahren hoherer Ordnung wie 64QAM haben zwar eine hohe Datenrate zur Folge, aber je größer die Ordnungszahl ist, desto komplexer und fehleranfälliger werden die Modulationssymbole sein. Ein möglichst störungsfreier Kanal ist deswegen wichtig, weil der Empfänger das übertragene Symbole wieder erkennen muss. Da der Scheduler immer die besten Kanäle sucht, kann im Durchschnitt immer das beste Modulationsverfahren eingesetzt werden. Die Allokation von RB anhand der Kanalkondition geschieht einmal pro Millisekunde im eNodeB.

An dieser Stelle lässt sich die Spitzendatenrate mittels folgender Formel berechnen:

$$Max(bps) = \#ST \times \#\frac{Symbole}{ST \cdot 1ms} \times \#\frac{Bit}{Symbol} \times 1000\frac{ms}{s}$$

Angenommen, dass ein 20 MHz Spektrum allokiert ist, 12 Symbole pro RB (1 ms), und 64QAM immer angewandt werden kann, so erreicht man eine Spitzendatenrate von 86,4 Mbps. Man beachte, dass bei der Berechnung die Anzhal von Subträger in einem 20 MHz Spektrum auf 1200 abgerundet ist. 64QAM liefert 6 Bits/Symbol.

Die LTE Spezifikation beschreibt eine Standardkonfiguration mit dem Einsatz von 2×2 MIMO (Siehe Abschnitt 4). In diesem Fall verdoppelt sich die Spitzendatenrate noch bis zu 170Mbps.

Dieser theoretische Wert kann aus den folgenden Gründen in der Realität nicht erreicht werden [8]:

- Fehlerkorrigierender Code (*Error correcting code*). Er wird verwendet, um die Fehlerrate zu senken. Die Länge von einem solchen Code beträgt oft ca. 30% der gesamten Codelänge
- *Retransmission*. Bei der Datenübertragung über die Luft können unkorrigierbare Bitfehler vorkommen. Die betroffenen Bits müssen in dem Fall erneut übertragen werden. Dies geschieht bis zu ca. 20% der Zeit.
- Sehr viele Kontroll- und Signalbits müssen neben den Nutzdaten mitgesendet werden.
- Die 64QAM Modulation arbeitet nur bei sehr kleiner Entfernung zur Basisstation [8]. Bei großer Entfernung zur Basisstation kann man die Symbole nicht mehr zurück–gewinnen.

3.1.4 Stärken & Schwächen von OFDMA in LTE

Die Vorteile von OFDMA sind:

- *Größere Bandbreite ist möglich*. Siehe Anfang Abschnitt 3.
- *Flexiblere Resourcenallokation*. Ein klarer Vorteil gegenüber FDMA und TDMA. Wie man in Abbildung 1 sieht, bekommt der Nutzer nun zu jedem Zeitpunkt ein Resource, d.h. minimale bis keine Zugriffsverzögerung. Der Nutzer kann auch jedem Frequenzbereich zugeteilt werden. Sobald ein Kanal gestört ist, wird ein anderer besserer Kanal für ihn verwendet.
- *Sehr spektraleffizient*. Da das eNodeB nach jeder Millisekunde eine neue mit der Zeit immer bessere Ressourcenallokation berechnet, kann im Durchschnitt fast immer das beste Modulationsverfahren verwendet werden.
- *Sehr robust gegen Mehrwegeausbreitung* [6]. Dies ist möglich durch Einsatz vom zyklischen Präfix.
- *Einfachere MIMO Operation*

Die bedeutendsten Nachteile sind:

- *Aufwändige Frequenz-Synchronization*. Da jeder Nutzer zu jedem Zeitpunkt praktisch alle verfügbare RBs zugeteilt werden kann, so muss der Sender sehr genau wissen, wann er für welchen Nutzer in welchem Frequenzbereich senden muss. Dieser Prozess ist sehr aufwändig
- Hohe *Peak-to-Average-Power-Ratio (PAPR)*. Das ist eine der größten Herausforderungen in OFDMA. Die hohe PAPR beim übertragenen Signal benötigt Linearität am Sender. Lineare Verstärker haben geringe Effizienz [3] bezüglich Energieverbrauch. Die Folge dafür wird im nächsten Abschnitt diskutiert.

3.2 Uplink-Übertragung

Mit OFDMA schafft man eine sehr hohe Datenrate durch die effiziente Nutzung von Ressourcen. Das Problem liegt aber an der hohen PAPR, die OFDMA nicht so energieeffizient macht. Bei den Basisstationen ist das Problem nicht so groß, denn die Basisstationen können immer mit Strom versorgt werden. Aber bei UEs, vor allem bei mobilen Geräten, bei welchen der Energieverbrauch eine wichtige Rolle spielt, ist dies ein großes Problem. Aus diesem Grund ist OFDMA keine optimale Lösung für Uplink-Übertragungen.

Der Trick ist, OFDMA zu modifizieren, so dass die PAPR deutlich sinkt. Die Idee wurde in SC-FDMA, *Single Carrier Frequency Division Multiple Access*, realisiert.

SC-FDMA funktioniert prinzipiell gleich wie OFDMA. Die Benennung *Single Carrier* ist aber etwas verwirrend, denn auch in SC-FDMA werden mehrere Subträger (und nicht nur einer!) für einen Nutzer verwendet. Der einzige Unterschied besteht darin, dass in SC-FDMA nur noch benachbarte RBs einem Nutzer zugeteilt werden können. Dies wird in der Abbildung 3 deutlich gemacht.

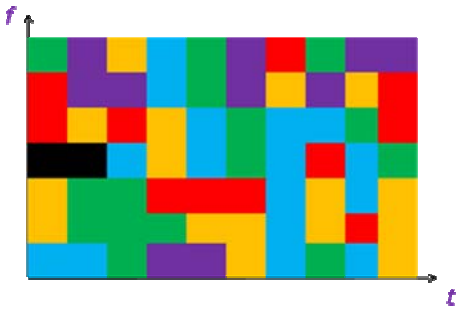


Abbildung 3: Resourcenzuteilung in SC-FDMA

Zwar übernimmt SC-FDMA die meisten guten Eigenschaften von OFDMA, es verliert aber Flexibilität bezüglich der Ressourcenzuteilung. Auch die Spektraleffizienz ist hier nicht mehr so hoch wie in OFDMA, denn benachbarte RBs müssen nicht unbedingt die optimale für den jeweiligen Nutzer sein.

SC-FDMA hat aber eine niedrigere PAPR[7] und ist somit geeigneter für Uplink-Übertragung. Die Berechnung der Spitzendatenrate ist analog zu OFDMA. Während OFDMA ein bereits weitverbreitetes Mehrfachzugriffsverfahren darstellt, ist SC-FDMA (mit zyklischem Präfix) eine relativ neue Technologie, die noch nicht in irgendeinem vorhandenen System eingesetzt wurde.

4. MIMO

Um Kapazität und Datenrate zu erhöhen, hat man einige Möglichkeiten. Am einfachsten verwendet man eine größere Bandbreite. Die Datenrate ist linear zur Bandbreite. Wird die Gesamtbandbreite verdoppelt, so kann man zwei mal höhere Datenrate erreichen. Das Problem ist aber: die Bandbreite ist beschränkt und vor allem teuer. Im Juli/August 2000 haben beispielsweise sechs Mobilfunkanbieter alle eine UMTS Lizenz in Deutschland zu einem Preis von je über 8 Milliarden Euro ersteigert [11].

Man kann auch die Zellen kleiner machen. In diesem Fall muss jede Zelle geringere Anzahl von Nutzern bedienen. Die Kosten dafür ist, man braucht noch mehr Basisstationen, um die gleiche Fläche zu decken.

Eine andere Möglichkeit wäre, ein Modulationsverfahren der noch höheren Ordnung als 64QAM zu verwenden, um mehr Bits pro Symbol zu übertragen. Dies ist allerdings schwer wegen dem im Abschnitt 3.1.3 bekannten Problem und von daher wird stattdessen oft 16QAM oder sogar QPSK verwendet.

Im vorherigen Abschnitt wurde das Phänomen der Mehrwegeausbreitung genannt. Während man den negativen Effekt der Mehrwegeausbreitung durch den Einsatz von zyklischen Präfixen bekämpfen kann, kann man dieses Phänomen nicht vermeiden. Deshalb versucht in LTE man dieses Phänomen so auszunutzen, dass man auch einen guten Effekt durch die Mehrwegeausbreitung bekommt, in dem man mehrere Antennen sowohl beim Sender als auch beim Empfänger verwendet. Jede Sendeantenne sendet Daten unabhängig von anderen Sendeantennen. Dies ist in der Funktechnik auch bekannt unter dem Begriff MIMO (Multiple Input Multiple Output).

MIMO ist ein Raummultiplexing-Verfahren. Zu den Frequenz- und Zeitdomänen wird eine weitere Domäne hinzugefügt, der Raum (Siehe Abbildung 4). Nun kann man zum selben Zeitpunkt und im selben Freuquenzbereich über verschiedene räumliche Wege senden/empfangen.

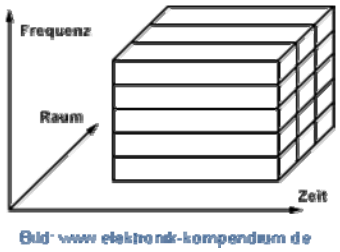


Abbildung 4: Raummultiplexing

Die wichtigsten Vorteile von MIMO System sind [5] :

- *Gruppengewinn.* Der Einsatz von mehreren Empfangsantennen führt trivialerweise zu einer gestiegenen Empfangsleistung. Durch das gezielte Senden werden Signale in Richtung der gewünschten Nutzer abgestrahlt. Die gesamte Sendeleistung wird dann für die vorhandene Sendeantenne verteilt. Der gesamte Energieverbrauch bei mehreren Sendeantennen ist nicht größer als bei nur einer Antenne. Bei UE mit mehreren Sendeantennen kann es deshalb sein, dass sie für eine Datenübertragung nur eine einzige Antenne verwendet, und trotzdem die optimale Datenrate erhält. D.h. geringer Energieverbrauch bei gleicher Datenrate.
- *Interferenzunterdrückungsgewinn.* Durch das gezielte Senden werden die Störungen für andere Nutzer auch geringer. Es gibt hier eine räumliche Trennung zwischen den Nutzern (Siehe Abbildung 5).
- *(Raum-)Multiplexgewinn.* Bei Raummultiplexing werden die Signale auf N Sendeantennen gleichmäßig verteilt. Die Datenrate kann dadurch um das N-fache erhöht werden.

Abbildung 5: Räumliche Trennung

LTE unterstützt im Downlink die 2×2 und 4×4 MIMO (Die Bezeichung N×N MIMO bedeutet den Einsatz von N Sende- und Empfangsantennen) Konfigurationen [3]. Die 2×2 MIMO ist als Basiskonfiguration in LTE festgelegt. Mit 4×4 MIMO kann theoretisch eine vierfach größere Datenrate erreicht werden. Die Vergrößerung der Datenrate betrifft nicht nur das gesamte System, sondern auch die einzelnen Nutzer.

Im Uplink wird MIMO nicht offiziel unterstützt, denn durch Einsatz von mehreren Sendeantennen steigt auch die Energieverbrauch, was nicht optimal für Mobilgeräte ist. An der Stelle kann Multi User MIMO (MU-MIMO) eingesetzt werden [8]. Hier senden zwei Mobilgeräte Daten gleichzeitig über dasselbe Frequenzband. Der Empfänger (die Basisstation) erkennt die zwei Übertragungsvorgänge zum selben Zeitpunkt. Auf diese Weise erfährt jede Nutzer keine Steigerung der Datenrate, jedoch verdoppelt sich die Datenrate des Systems.

5. ZUSAMMENFASSUNG & AUSBLICK

LTE bietet eine sehr attraktive Leistung hinsichtlich der Datenrate, Kapazität, und Kosten. Der Einsatz von OFDMA im Downlink ermöglicht eine effiziente Nutzung des Spektrums, hohe Datenrate, und geringe Antwortzeit. Im Uplink wird die Energie effizienter durch SC-FDMA genutzt, ohne dabei einen großen Verlust an Datenrate zu verursachen. Mit 4×4 MIMO im Downlink kann theoretisch bis zu vierfache Datenrate erreicht werden.

Die Netzwerk Elemente für LTE sind nicht kompatibel mit UMTS oder HSPA, d.h. ein Betreiber, der LTE einführen will, muss das RAN fast komplett neu aufbauen. Die Möglichkeit für dem synchronen Betrieb mit vorhandenen Netzwerke (wie GSM, UMTS) soll jedoch noch bestehen.

LTE soll langfristig UMTS ablösen und befindet sich am Ende der Standardisierungsphase. Man sagt LTE sei die Brücke zu 4G. LTE soll während der Einführungsphase im Frequenzbereich um 2,6 GHz operieren. Die ersten Produkte sind erst 2010 auf dem Markt zu erwarten.

6. LITERATUR

[1] 3GPP Technical Report TR 25.913, ver. 8.0.0. '*Requirements for Evolved UTRA and Evolved UTRAN*', Januar 2009

[2] 3GPP Technical Specification TS 36.211, ver 8.4.0. '*Physical Channels and Modulations*', September 2008

[3] 3GPP Technical Specification TS 36.300, ver 8.6.0. '*Overall Description Stage 2*', September 2008

[4] Holma, Harri und Antti Toskala (ed.). *WCDMA for UMTS – HSPA Evolution and LTE*. West Sussex: John Wiley & Sons, Ltd., 2007

[5] Kaiser, Thomas und Andreas Wilzek. '*MIMO – der Datenturbo für die mobile drahtlose Zukunft*'. NTZ Heft 1/2007

[6] Van Nee, L. und R. Prasad. *OFDM for Wireless Multimedia Communications*. Artech House, 2000.

[7] Dahlman, Erik, et. al. '*Key Features of the LTE Radio Interface*'. Ericsson Review No. 2. 2008

[8] Sauter, Martin. *Beyond 3G*. West Susex: John Wiley & Sons, Ltd., 2009

[9] Wikipedia. *3GPP Long Term Evolution*. 26. Januar 2009. http://en.wikipedia.org/wiki/3GPP_Long_Term_Evolution

[10] Wikipedia. *3rd Generation Partnership Project*. 24. März 2009. http://de.wikipedia.org/wiki/3rd_Generation_Partnership_Project

[11] Teltarif.de. *UMTS Auktion beendet: sechs Lizenznehmer im Boot*. 24. März 2009. http://www.teltarif.de/arch/2000/kw33/s2829.html

Femtozellen – Base Stations For The Masses

Sören Ruttkowski
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
ruttkows@in.tum.de

Kurzfassung
Um den steigenden Bandbreitenbedarf auf mobilen Endgeräten auch weiterhin zu decken und die allgemeine Leistung nicht zu verschlechtern, ist eine Verkleinerung der Zellgrößen im Mobilfunkbereich unabdingbar. Femtozellen sind heimische Basisstationen, die mit sehr geringer Leistung senden und verbreitete Mobilfunktechnologien verwenden. Auch wenn in den letzten Jahren die Vorraussetzungen für einen breiten Einsatz von Femtozellen in der Bevölkerung geschaffen wurden, z.B. durch eine hohe Anzahl an Breitbandverbindungen, so sind noch einige technische Probleme zu lösen, die in diesem Paper diskutiert und zu denen erste Lösungsansätze aufgezeigt werden.

Schlüsselworte
Femtozelle, NodeB, UMTS, 3GPP, Mobilfunk, Bandbreite

1. Einleitung
Femtozellen sind seit kurzem mehr und mehr in den Fokus der Aufmerksamkeit der gesamten Mobilfunkbranche geraten. Femtozellen sind dabei insbesondere für Mobilfunknetzbetreiber interessant, um ihren Kunden höhere Datenraten in den eigenen vier Wänden und in Gebieten mit unzureichender Abdeckung durch Makrozellen zur Verfügung zu stellen und ermöglichen somit neue Geschäftsmodelle. Da der Durchbruch der Femtozellen in den breiten Markt laut Marktprognosen [7] kurz bevorsteht, sollen im Folgenden die Technologie, sowie noch zu lösende Probleme der Femtozellen beschrieben und Lösungen sowie Lösungsansätze für diese Probleme vorgestellt werden.

2. Definition einer Femtozelle
Eine Femtozelle ist laut Femto Forum [6] ein heimischer Zugangspunkt mit geringer Sendestärke von unter 100 mW EIRP, was von der Sendeleistung her in der Größenordnung eines üblichen WLan Routers liegt. Femtozellen nutzen aber im Gegensatz zu WLan Routern Technologien, wie das Universal Mobile Telecommunications System (UMTS) und arbeiten somit im lizenzierten Spektrum, unterliegen also gesetzlichen Regulierungen und müssen vom Netzbetreiber kontrolliert und verwaltet werden können. Die Verwaltung dient hierbei dem Schutz des Mobilfunkbetreibernetzes und ermächtigt die Netzbetreiber zudem allen gesetzlichen Vorgaben bezüglich der Mobilfunklizenzen gerecht werden zu können (siehe 4.4). Femtozellen transportieren die Anrufe und Daten der Nutzer dabei anders als andere Netzbestandteile, wie etwa Mikro- oder Picozellen nicht über die hierarchische Netzstruktur des Netzoperators, sondern direkt über die Breitbandverbindung des Kunden. Das Ziel solcher Femtozellen besteht darin, für den Netzbetreiber kostengünstig Abdeckung und Kapazität bereitzustellen, so dass dieser auf weitere Makro- oder Mikrozellen verzichten können. [6]

Femtozellen und Picozellen unterscheiden sich von Makro- und Mikrozellen durch die Größe der abgedeckten Fläche, da diese bei beiden nur einige zehn Meter im Radius, bei Makrozellen aber zwischen 350m und 20km und bei Mikrozellen zwischen 50m und 300m beträgt.

3. Gründe für Femtozellen
Femtozellen sind insbesondere in der letzten Zeit immer wichtiger geworden und Ihre Umsetzung hat im Teststadium in den Vereinigten Staaten bereits begonnen, um Kunden bessere Abdeckung und günstige Tarife in den eigenen vier Wänden bieten zu können. Insbesondere die Netzbetreiber erhoffen sich große Vorteile bezüglich der Netzkosten, sowohl in den zukünftigen Anschaffungs- als auch in den Betriebskosten (CapEx und OpEx) [2].

3.1 Motivation zum Einsatz von Femtozellen
Neben diesen Kostenvorteilen für die Netzbetreiber ist eine Verkleinerung der Zellen auch der einzig wirklich praktikable Ansatz um mehr Bandbreite in die Geschäftsräume und Wohnungen der Kunden zu bringen. Nach dem Shannon-Hartley-Gesetz besitzt jeder Kanal eine maximale Kapazität, die nicht überschritten werden kann. Somit ist es einfacher die Zellgröße zu verkleinern, was durch die geringere Entfernung von Sender und Empfänger auch die Kanalqualität bezüglich Signal-Rausch-Abstand steigert, als die Qualität des Kanals in den Makrozellen zu verbessern. Des Weiteren stellt die Abschirmung durch Wände für Femtotzellen kein Problem dar, da die Femtozellen direkt im Haus arbeiten und sie somit nicht wie die Makrozellen abgeschirmt werden. Hierdurch kann insbesondere die Abdeckung in entlegenen Regionen deutlich verbessert werden. [12]

Die Vorteile der Netzbetreiber dienen auch den Mobilfunknutzern, da diese mehr Bandbreite und eine bessere Abdeckung erfahren. Des Weiteren sollen die Kostenvorteile der Netzbetreiber an die Endkunden weitergegeben werden, so dass die ersteren gegenüber Voice over Internet Protocol (VoIP) und Triple Play Angeboten konkurrenzfähig bleiben. Durch die Nutzung einer einheitlichen Funktechnologie (z. B. UMTS) ist es dem Benutzer zudem möglich nur ein einzelnes Endgerät zu besitzen. Anstelle einen Festnetzanschluss und ein Mobilfunkgerät zu verwenden und diese für gleiche Zwecke zu nutzen. Somit werden alle Kontaktdaten eines Nutzers auf einem Endgerät gebündelt. Diese Bündelung von Diensten wird allgemein als Fixed Mobile Convergence (FMC) bezeichnet, also dem Zusammenschluss von Festnetz- und Mobilfunkgeräten.

3.2 Enabler - Voraussetzungen für Femtozellen
Auch wenn eine Verkleinerung der Zellgröße, bereits seit sehr langem genutzt wird, um die Übertragungskapazität zu erhöhen, so ist dennoch die Idee und Technologie der Femtozellen sehr neu. Dies ist zu einem großen Teil darauf zurückzuführen, dass die Bereitstellung und Umsetzung der Femtozellen erst seit kurzem machbar ist und aufgrund gestiegener Nachfrage auch sinnvoll erscheint.

Vor allem die hohe Anzahl an Breitbandverbindungen in privaten Haushalten ist eine wichtige Vorraussetzung um Femtozellen bei den Kunden zuhause zu betreiben, da nur so der Rücktransport der Anrufe und Daten ins Betreibernetz mit ausreichender Geschwindigkeit sichergestellt werden kann. Auch die nötige IP Technologie hat sich erst in den letzten Jahren aufgrund der zunehmenden Vernetzung entwickelt und bietet somit die Möglichkeit einer sicheren und skalierbaren Anbindung der Femtozellen an das Kernnetz der Mobilfunknetzbetreiber.

Ein weiterer wichtiger Faktor ist das in den letzten Jahren stark gewachsene Interesse der Bevölkerung an Datendiensten auf ihrem mobilen Endgerät. Als wichtigste Applikationen sind hier die Social-Network-Plattformen und Videoportale der Web 2.0 Bewegung zu nennen, die hohe Anziehungskraft auf Nutzer haben und somit bei diesen das Interesse an schnellen Datendiensten auf dem mobilen Endgerät wecken. Da Nutzer sich größtenteils in Häusern aufhalten, bieten sich lokale Zugänge an um das hohe Datenaufkommen zu decken. Von Betreiberseite hat zudem die Konkurrenz durch VoIP Dienste zu einem großen Interesse an Femtozellen beigetragen.

Ein Faktor der bisher noch nicht befriedigend gelöst wurde, jedoch große Fortschritte gemacht hat, ist die kostengünstige Implementierung der Femtozellen. Hierbei müssen Preise deutlich unterhalb der 150€ Marke erreicht werden, um wirtschaftlich sinnvoll eingesetzt und eine hohe Durchdringung der Bevölkerung erreichen zu können. Dank des starken Preisverfalls von Prozessoren, insbesondere der Field Programmable Gate Arrays (FPGA) und der Digital Signaling Processors (DSP), ist der Preis für die Implementierung einer Femtozelle bereits stark gesunken. Ein großer Kostenfaktor bleibt jedoch eine hochgenaue Uhr, die momentan oft ein OCXO Kristall darstellt, welcher jedoch alleine bereits mehrere hundert Euro kostet. Alternativ gibt es seit kurzem sogenannte TCXO Kristalle (ca. 70€ bei 500 ppb Genauigkeit), die über 6-18 Monate hoch genau die Zeit messen und somit eine Nutzung der Mobilfunkfrequenzen überhaupt erst erlauben. [1]

4. Offene Probleme mit Lösungsansätzen

Auch wenn dank Breitband und günstigerer Zeitgebung die Vorraussetzungen für Femtozellen gelegt sind, so bleiben noch einige Probleme bestehen, die zu lösen sind. Einige dieser Probleme werden in diesem Kapitel besprochen und erste Lösungen und Lösungsansätze werden vorgestellt.

Um die folgenden Abschnitte dieses Kapitels leichter verständlich zu machen, soll an dieser Stelle kurz die UMTS Netzstruktur erläutert werden.

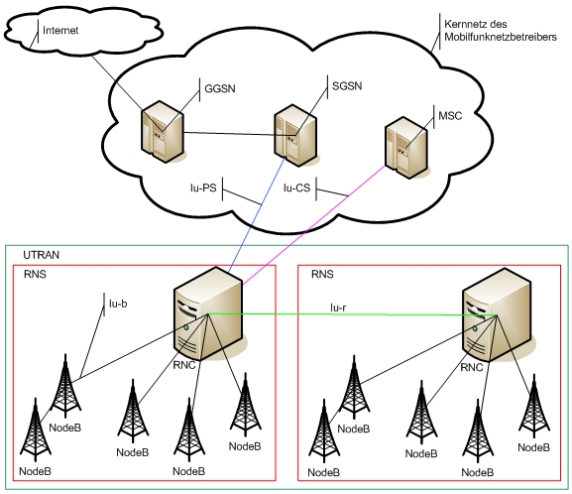


Abbildung 1: UMTS Netzstruktur

UMTS ist ein hierarchisch aufgebautes Mobilfunknetz, dass grob in zwei Teile gegliedert werden kann. Das Kernnetz des Mobilfunknetzbetreibers und das sogenannte UMTS Terrestrial Radio Access Network (UTRAN). Im UTRAN finden sich die Basisstationen, welche bei UMTS „NodeB“ genannt werden um sie von ihrem 2G Gegenstück abzugrenzen. Die NodeB sind wiederum mit einem Radio Network Controller (RNC) über das mobilfunkspezifische Iu-b Protokoll verbunden, der die NodeB verwaltet, bündelt und deren Nachrichtenaufkommen ins Kernnetz weiterleitet. Das Iu-b Protokoll unterscheidet sich dabei von Hersteller zu Hersteller und bündelt Daten und Kontrollinformationen für die Übertragung zwischen NodeB und RNC.

Im Kernnetz sind insbesondere der Serving GPRS Support Node (SGSN), welcher für den packetbasierten Nachrichtenverkehr und das Mobile Switching Center (MSC), welches sich um die Leitungsvermittlung kümmert, wichtige Komponenten für den Netzbetrieb. Die einzelnen Bestandteile kommunizieren hierbei über eigene mobilfunkspezifische Protokolle mit dem Präfix Iu, und Endungen wie z.B. CS für Channel Switched und PS für Packet Switched. Um den Nachrichtenverkehr wieder ins öffentlich zugängliche Internet zu leiten gibt es im Kernnetz einen Gateway GPRS Support Node (GSGN), der diesen Dienst bereitstellt.

4.1 Integration der Femtozellen ins Mobilfunknetz

Um mit dem Mobilfunkendgerät einen Anruf tätigen, oder Datendienste nutzen zu können, müssen sich die Femtozellen mit dem Kernnetz des Betreibers verbinden. Hierbei ist insbesondere die schiere Anzahl an erwarteten Femtozellinstallationen eine große Herausforderung, da die Netze der Netzbetreiber nicht auf eine so große Menge an Kleinstbasisstationen ausgelegt ist. Die Netzstruktur muss deswegen stark abgeändert werden.

Um Femtozellen in die beschriebenen Netze der Mobilfunkbetreiber einzubinden sind sehr viele unterschiedliche Möglichkeiten denkbar. In letzter Zeit haben sich drei Architekturen als die Sinnvollsten herauskristallisiert und wurden akkreditiert. Eine Festlegung auf eine der drei Architekturen wird Ende 2008 erwartet, so dass die Femtozellenhersteller die nötigen Protokolle und Technologien implementieren können und Kunden nicht für jeden Mobilfunknetzbetreiber eine andere Femtozelle benötigen.

4.1.1 Integrationslösung Iu-b over IP

Eine Herangehensweise, die große Investitionen vermeidet, ist es Femtozellen als normale Basisstationen, also im wahrsten Sinne des Wortes als „Home NodeB“ zu implementieren, so dass diese auch über das übliche Iu-b Protokoll kommunizieren. Den einzelnen RNCs wird dann ein Gateway bereitgestellt, der einen über das öffentliche Internet aufgebauten IPSec Tunnel einer Femtozelle entgegennimmt. Dabei sendet die Femtozelle die Iu-b Informationen über den IPSec Tunnel an den Gateway, welcher diese aus dem IPSec Tunnel entnimmt. Die Iu-b Informationen werden dann direkt von den bereits vorhandenen RNCs verarbeitet und in das Kernnetz über die bestehende hierarchische Netzstruktur weitergeleitet. [6]

Dieses Verfahren kann, wie weiter oben beschrieben, sehr kostengünstig mit der vorhandenen Infrastruktur umgesetzt werden, hat jedoch den großen Nachteil, dass die RNCs eine so große Anzahl von tausenden oder zehntausenden NodeBs nicht unterstützen, das Netz also nicht mit der erwarteten hohen Anzahl an Femtozellen skaliert. [7]

Des Weiteren ist das Iu-b Protokoll für diesen Zweck nicht vorgesehen und stellt Anforderungen an die Übertragung bezüglich Packet Loss, Delay und Jitter, die über eine normale Breitbandverbindung nicht immer eingehalten werden können. Das Protokoll müsste also robuster und toleranter implementiert werden, als dies bisher der Fall ist. Da sich die Implementierungen des Iu-b Protokolls von Hersteller zu Hersteller unterscheiden, ist es zudem problematisch diese Variationen in den Femtozellen umzusetzen und die Femtozellen würden sich von Netz zu Netz in Ihrer Funktionsweise unterscheiden.

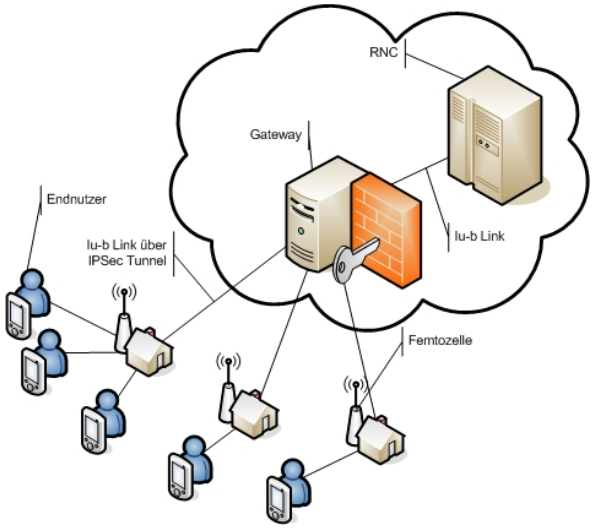


Abbildung 2: Integration durch Nutzung der bestehenden hierarchischen Netzinfrastruktur

4.1.2 Integrationslösung RAN Gateway

In diesem Ansatz, wird ein zentraler Radio Access Network (RAN) Gateway im Kernnetz eingerichtet der als UMA Network Controller (UNC) bezeichnet wird und über den sich die Femtozellen mit dem Kernnetz verbinden [10]. Die Femtozellen bauen hierbei wieder einen sicheren IPSec Tunnel zum Gateway auf, kommunizieren allerdings nicht mit dem Iu-b Protokoll, sondern über das UMA Protokoll. Der Gateway ist nun wieder dafür verantwortlich die Verbindungen der einzelnen Femtozellen anzunehmen, wobei der Gateway hier sehr viele parallele Verbindungen unterstützen muss, den Verkehr bereits zu bündeln und ins Kernnetz einzuspeisen. Für die Einspeisung ins Kernnetz werden wieder die üblichen Protokolle des Mobilfunknetzes genutzt, also Iu-CS oder Iu-PS. [13]

Diese Technologie wird momentan bereits bei sogenannten Unlicensed Mobile Access (UMA) Lösungen eingesetzt, wie z.B T-Mobiles HotSpot@Home. Auf diese Weise ist es möglich die bei einigen Netzbetreibern bereits vorhandene UMA Lösung auch für das Femtozellendeployment zu nutzen oder ermöglicht es UMA Dienste anzubieten. Das UMA Protokoll ist standardisiert und für den Zweck der Integration von vielen Dual-Mode Endgeräten (Mobilfunk und WLan) ins Kernnetz entwickelt worden. Es liegt nahe, dass bestehende und erprobte Protokoll an die Femtozellenlösung anzupassen und es für diese neue Lösung zu nutzen. Durch die Integration von gewissen Netzbestandteilen in die Femtozellen (Collapsed Stack) skaliert der Ansatz sehr gut und kann kostengünstig umgesetzt werden. [10]

Problematisch bleibt die Anbindung des RAN Gateways an die Kernnetzkomponenten der Netzbetreiber, da die hier eingesetzten Iu-PS und Iu-CS Protokolle nicht standardisiert sind und die Gateways sich somit in Ihrer Bauweise unterscheiden müssen. [14]

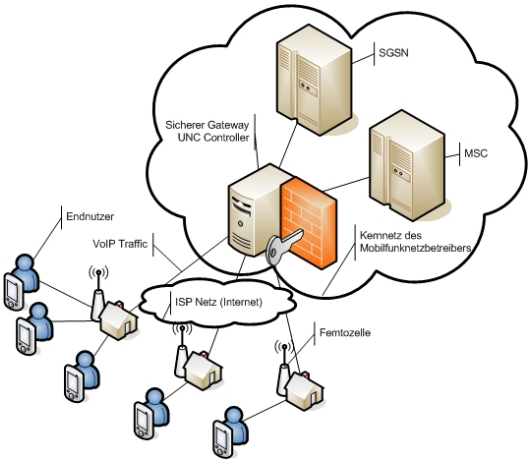


Abbildung 3: Integration durch Einsatz eines RAN Gateways

4.1.3 Integrationslösung IMS/SIP

In der Integrationslösung mithilfe vom Internet Media Sub-System (IMS) und dem Session Initiated Protocol (SIP) wird ein zweites Kernnetz für die Verbindung der Femtozellen geschaffen, dass parallel zum bestehenden Kernnetz für die Makro- und Mikrozellen betrieben wird und auf IMS basiert. Der Einsatz eines zweiten Kernnetzes, welches rein auf IP Technologien basiert, ist eine zukunftsweisende Entscheidung um künftig komplett auf ein All-IP Netz umzustellen und das bisherige hierarchische Netz abzuschaffen. [9]

Das IMS Interface wandelt dabei den aufkommenden Nachrichtenverkehr an der Femtozelle in IP Pakete um und sendet diese mithilfe des SIP Protokolls als VoIP. Diese Architektur hat den Vorteil, dass sie auf bekannten Protokollen aufbaut und somit eine schnelle Umsetzung ermöglicht. Des Weiteren skaliert der Ansatz sehr gut mit der steigenden Anzahl an Femtozellen, da bereits einige der Komponenten, wie SGSN und GGSN in der Femtozelle als sogenannter Collapsed Stack gebündelt werden. Der größte Nachteil sind die hohen Anschaffungs- und Betriebskosten, bei zwei parallelen Netzen. [17]

Ein weiteres Problem ist die fehlende Servicetransparenz, so dass die Services einmal im Kernnetz und einmal für die Femtozellen angeboten werden müssen. Um die getätigten Anrufe wieder ins Telefonnetz zu leiten, muss ein SIP-enabled MSC geschaffen werden, der die Anrufe auf SIP-Basis entgegennimmt und alle Dienste bereitstellt, die von eigentlichen MSC bereits angeboten werden. Diese Lösung hat somit mehr mit der gewöhnlichen UMTS Lösung gemeinsam, als dass es eine wirkliche IMS Implementierung darstellt.

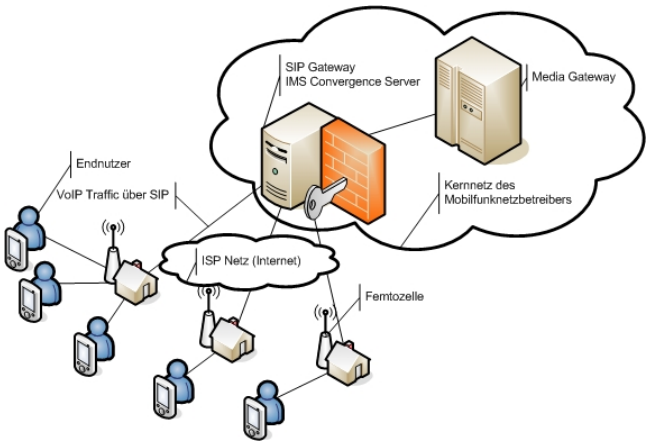


Abbildung 4: Integration durch Nutzung von IP Technologien

4.2 Interferenzen

Interferenzen treten dann auf, wenn zwei Sender auf ein und derselben Frequenz arbeiten. Da die Anzahl an zur Verfügung stehenden Frequenzen sehr begrenzt ist und jeder Betreiber nur ein bis zwei Frequenzbereiche für UMTS besitzt, kann es bei allen Betreibern zu Interferenzen kommen.

Femtozellen am Rande einer Makrozelle bieten für den Kunden und die auf der Femtozelle registrierten Endgeräte guten Empfang und hohe Datenraten. Allerdings überlagert die Femtozelle die Signale der Makrozelle für alle Nutzer. Mobilfunknutzer, die nicht auf der Femtozelle registriert sind und diese somit nicht nutzen können, können die Signale der Makrozelle aufgrund des „Lärms“ der Femtozelle also nicht mehr empfangen. Dies führt zu Verbindungsabbrüchen oder deutlich schlechterer Servicequalität für letztere.

Bei Femtozellen, welche sich sehr nah an einer Makrozelle, bzw. deren NodeB befinden, wird die Femtozelle mit seiner sehr geringen Sendeleistung durch die Makrozelle überlagert. Somit ist der von der Femtozelle abgedeckte Bereich sehr klein und der Kunde kann die Dienste und Konditionen der Femtozelle nur in einem sehr kleinen Umkreis um die Femtozelle nutzen. Um dies zu vermeiden werden Smart Antennas eingesetzt, die eine bessere Filterung der Signale des Endgerätes ermöglichen.

Um Interferenzen zu vermeiden, kann zudem die Sendeleistung der Sender angepasst werden, so dass sich ihre Wellen räumlich weniger überlagern. Hierfür werden adaptive Algorithmen eingesetzt, die die Sendeleistung der Femtozellen so steuern, dass es zu möglichst wenig Interferenzen kommt. Dabei werden die Zellwechsel der Nutzer als Anhaltspunkt genommen, so dass die Sendeleistung angepasst wird, damit ein Benutzer aufgrund der veränderten Reichweite der Femtozelle die zugeordnete Zelle möglichst selten wechseln muss. Eine weitere Größe zum Steuern der Sendeleistung ist die Anzahl an Verbindungsabbrüchen im Bereich der Femtozelle, da dies ein Zeichen von Interferenzen ist, wie oben beschrieben.[5]

Hauswände, die die Signale der Makrozellen abschirmen und es den Netzbetreibern somit erschweren dem Kunden eine ausreichende Abdeckung innerhalb seiner Wohnung zu ermöglichen, wirken sich bei Femtozellen positiv aus, da sie die Umwelt von den Funkwellen der Femtozelle abschirmen, wie auf Abbildung 5 zu sehen ist. [11]

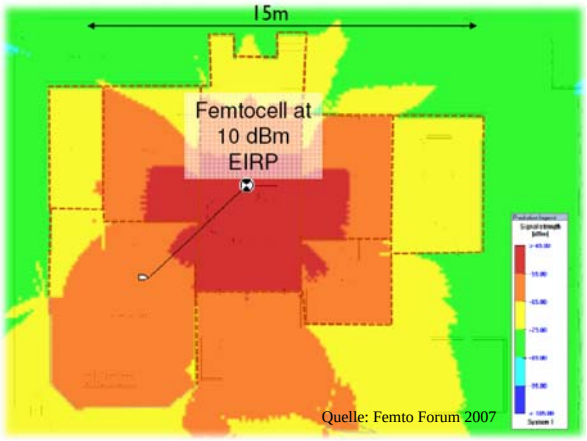


Abbildung 5: Sendestärken im Umkreis der Femtozelle (rot=hoch) [6] F. 7

Eine weitere Methode um Interferenzen zu vermeiden ist der Einsatz von Wideband Code Division Multiple Access (W-CDMA). Hierbei werden orthogonale Scrambling Codes verwendet. Der Empfänger, kann nun aus einem Signal mehrere Kanäle herausdividieren, indem er einen gewissen, zwischen Sender und Empfänger bekannten Code an das Signal anlegt. Somit ist es mehreren Sendern möglich auf einer Frequenz zu operieren, was auch den gleichzeitigen Betrieb von Femtozellen und Makrozellen in ein und demselben Bereich ermöglicht, solange diese unterschiedliche Codebereiche verwenden.

Da die Anzahl an orthogonalen Codes jedoch begrenzt ist, müssen diese sinnvoll verteilt werden. Diese Verteilung wird bisher statisch durch die Techniker der Mobilfunknetzbetreiber festgelegt, die die Zellgrößen und dazugehörigen Scrambling Codes optimal aufeinander abstimmen. Da die Konfiguration der Femtozellen jedoch nicht zentral geplant werden und die Anzahl an Femtozellen in einem Bereich sehr hoch sein kann, kommt es zu Überlagerungen, auf einer Frequenz und einem Scrambling Code. [16]

Eine endgültige Lösung für das Problem der Interferenzen ist noch nicht gefunden. Allerdings wird auch weiterhin in diesem Bereich geforscht um den Einfluss von Interferenzen bestmöglich zu minimieren.

4.3 Einfache Installation der Femtozellen

Ein operativer Aspekt, der jedoch auch hohe technische Ansprüche hat, ist eine einfache Installation der Femtozellen durch die Endnutzer, da nur auf diese Weise die Femtozellen wirtschaftlich sinnvoll in die Heime der Nutzer gebracht werden können. Bisher wird die Errichtung und Einrichtung einer Mobilfunkzelle von Mitarbeitern des Mobilfunknetzbetreibers vorgenommen. Die Radioparameter der einzelnen Zelle wurden dabei so eingestellt, dass sie Ihre Dienste an dieser Position optimal erfüllt. Für die bisherige Installation gibt es dabei die Begriffe des „Man-in-a-Van“ oder bei größeren Zellen des „Truck-Roll“, welche schon den Aufwand und die hiermit verbundenen Kosten erahnen lässt.

Femtozellen hingegen sollen sich möglichst ohne Konfigurationsaufwand des Nutzers installieren lassen, so dass dieser die Femtozelle kauft, zu hause an die Breitbandverbindung anschließt und innerhalb kürzester Zeit die Dienste der Femtozelle nutzen kann (Zero-Touch Installation).

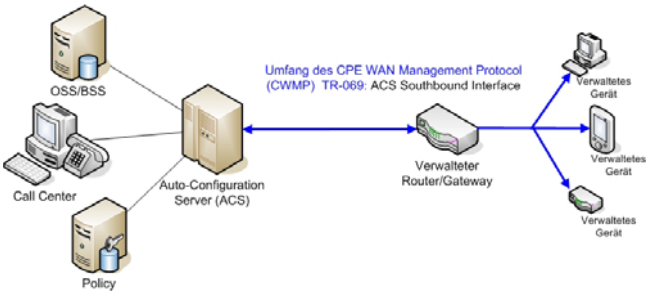


Abbildung 6: Überblick über TR069 Fernwartungsprotokoll

Da die Femtozelle möglichst automatisch konfiguriert werden soll, ist das Fernwartungsprotkoll TR-069 [3], welches bisher jedoch nur für Router definiert ist, für die Netzbetreiber von hohem Interesse, da diese Technologie als Orientierung verwendet werden kann.

Ein mögliches Szenario unter Einsatz des TR-069 wäre hierbei:

1. Femtozelle wird mit Standardsoftware und SIM Karte ausgeliefert
2. Beim ersten Start baut die Femtozelle eine Verbindung zum Netzbetreiber auf und authentifiziert sich mit der SIM Karte
3. Vom Autoconfiguration-Server des Netzbetreibers werden die Firmware, Updates und weitere betreiberspezifische Konfigurationsdaten (Frequenz, ID, Sendeleistung, …) heruntergeladen
4. Hierbei wird auch eine Liste aller berechtigten Endgeräte geladen
5. Die Femtozelle ist somit im Betreibernetz registriert und mit dem Kernnetz über Internet verbunden
6. Das Endgerät meldet sich an der Femtozelle an
7. Der Nutzer kann die Dienste über die Femtozelle verwenden

Der Netzbetreiber hat somit jederzeit die Möglichkeit die Femtozelle zu verwalten und kann diese bei schadhaftem oder illegalem Verhalten auch komplett abschalten.

Neben der Installation und dem Laden der Firmware, muss sich die Femtozelle selbst kalibrieren und in ihre Umgebung einbinden, ohne große Interferenzen zu verursachen, wie in 4.2 dargestellt. Dies betrifft die Konfiguration aller Radioparameter und der Wahl von Scrambling Codes und Sendestärke, die hier jedoch nicht vorgegeben, sondern von der Femtozelle selbst gewählt werden.

Obwohl sich dieses Protokoll bei DSL Gateways bereits bewährt hat, gibt es viele kleine Probleme, wie z. B. die Erreichbarkeit der Server, weswegen die Lösung am besten funktioniert, wenn alle Dienste von einem Anbieter genutzt werden. Dazu müssten die Mobilfunknetzbetreiber mit den Anbietern von Breitbandkabellösungen kooperieren.

4.4 Lokalisierung und Notrufe

Zum illegalen Verhalten der Femtozellen würde gehören, dass sie in einem Bereich außerhalb der Sendeberechtigung des Netzbetreibers auf dessen Frequenz sendet, da UMTS Frequenzen im lizenzierten Spektrum liegen. Die Position der Femotzelle muss also bekannt sein, um Ihr Verhalten den gesetzlichen Bestimmungen anzupassen. So darf die Femtozelle auch erst dann anfangen zu senden, wenn sichergestellt ist, dass sie an ihrer momentanen Position überhaupt senden darf.

Eine weitere gesetzliche Vorgabe, die die Lokalisierung der Femtozelle nötig macht findet sich in §108 des Telekommunikationsgesetzes (TKG) [4]. Dieser Artikel macht es nötig bei einem Notruf innerhalb der Femtozelle der nächstgelegenen Notrufstelle auch „Daten, die zur Ermittlung des Standortes erforderlich sind, von dem die Notrufverbindung ausgeht“ zu übermitteln. Um die Position einer Femtozelle zu bestimmen gibt es mehrere Ansätze.

Bei einer momentan in den USA eingesetzten Femtozellenlösung, wird zur Positionsbestimmung das Global Positioning System (GPS) eingesetzt. Dies hat jedoch den Nachteil, dass der GPS Empfänger eine Sichtverbindung zu seinen Satelliten benötigt. Des Weiteren kosten GPS Lösungen zwischen 600 und 700 €, was wiederum den Preis der Femtozellen erhöht und damit einer breiten Durchdringung der Gesellschaft mit Femtozellen entgegenwirkt. [12]

Ein anderer Ansatz ist die Hyperbelortung, die die Zeitabstände der Signale umliegender Makrozellen ausnutzt um Ihre eigene Position zu berechnen. Dieser Ansatz benötigt jedoch den Empfang von drei oder mehr Makrozellen, um die Position sicher zu bestimmen, was insbesondere in abgelegenen Bereichen, in denen die Femtozellen für eine Verbesserung der Abdeckung wohl mit als erstes eingesetzt werden, eher selten der Fall ist.

Eine weitere Lösung stellt eine eindeutige Identifizierung der Femtozellen dar, mit einer Subscriber Identiy Module (SIM) Karte, wie sie von Endgeräten bekannt ist. Jede der Femtozellen wird dann in einem sogenannten Master Street Access Guide (MSAG) eine Adresse zugeordnet. Bei einem Notruf wird diese Adresse dann abgefragt und als Position der Femtozelle herangezogen. Eine Straßenadresse ist dabei auch den Notdiensten hilfreicher als bloße Koordinaten, da somit eine Anlaufstelle bekannt ist und den Menschen schneller geholfen werden kann. Allerdings muss die Adresse im MSAG verifizierbar sein, da sich in der Vergangenheit bei UMA Installationen gezeigt hat, dass Nutzer den Netzbetreibern Adressänderungen nicht immer mitteilen [12]. Die Netzbetreiber müssen aber den gesetzlichen Bestimmungen genüge tun.

Eine letzte Möglichkeit, bei der allerdings alle Dienste, also Breitbandanbindung und Mobilfunk von einem Anbieter bezogen werden müssen ist die Nutzung einer dem Festnetzanschluss zugewiesenen „LineID“. Bei dieser Lösung ist die Femtozelle an eine gewisse LineID gebunden und kann nur von diesem Anschluss aus Ihre Dienste anbieten. Da der LineID immer ein Anschluss und damit auch eine Adresse zugewiesen werden kann, ist eine ständige Kontrolle möglich. Um diese Zuordnung zu tätigen, müssen aber alle Daten bezüglich LineID und Adresse, sowie der zugeordneten Femtozelle bei einem Anbieter liegen.

Dies alles spricht für einen hybriden Ansatz aus MSAG und Lokalisierung, da nur so die Position der Femtozelle garantiert werden kann und die passenden Informationen für die Notrufdienste zu Verfügung gestellt werden können.

4.5 Zeitliche Synchronisierung

Eine Synchronisierung der Uhren einer Femtozelle ist besonders wichtig, damit diese überhaupt auf den Frequenzen der Netzbetreiber operieren und die Mobilfunkstandards anbieten können, da die Signale im Mobilfunkbereich sehr präzise sein müssen. Laut Standard dürfen Makrozellen dabei nur um 50 parts-per-billion, also 50 Nanosekunden abweichen. Für Femtozellen wurde dies in Release 5 der 3G Standards bereits auf 100 Nanosekunden ausgeweitet, jedoch ist auch diese Genauigkeit noch eine hohe Anforderung an Femtozellen. [7]

In Makrozellen werden momentan Oven Controlled Crystal Oscillator (OCXO) Quarzkristalle eingesetzt, die eine hohe Genauigkeit über 12-24 Monate beibehalten. Nach dieser Zeit werden die Quarzkristalle durch Techniker kalibriert oder es wird ein GPS Signal verwendet, welches mit sehr genauen Zeitgebern arbeitet. Da diese beiden Technologien allerdings sehr kostspielig sind und somit für Femtozellen wirtschaftlich nicht sinnvoll eingesetzt werden können, müssen andere Lösungen gefunden werden.

Als eine gute Möglichkeit der Synchronisierung haben sich die erst vor kurzem entwickelten Temperature Compensated Crystal Oscillator (TCXO) Quarzkristalle gezeigt, die deutlich kostengünstiger sind (bei 500 ppb Genauigkeit ca. 70 €), auch wenn sie immer noch für den Großteil der Kosten einer Femtozelle verantwortlich sind. Diese TCXO Kristalle bleiben über 6-18 Monate hochgenau und werden mithilfe des IEEE 1588 Synchronisierungsprotokolls kalibriert, welches eine Genauigkeit von 100 Nanosekunden erlaubt. Somit werden die Signalvorgaben der Netze eingehalten und die genaue Uhr kann auch für die Positionsbestimmung durch die beschriebene Hyperbelortung herangezogen werden. [16]

4.6 Sicherheit der Femtozellen und Daten

Um die Sicherheit ihrer Kernnetze zu gewährleisten, müssen Maßnahmen getroffen werden um Hackern und jeglichem schadhaftem Code den Zugang zum Kernnetz zu verweigern.

Die Sicherheit fängt dabei in der Femtozelle an, welche gegen Manipulationen abgesichert werden muss, so dass sie nicht dazu gebracht werden kann Ihre Dienste an anderen als den gesetzlich erlaubten Orten zu erbringen. Da an die Femtozellen oftmals auch günstigere Tarife gebunden sind, wären durch eine ungenügende Sicherung der Femtozellen auch wirtschaftliche Nachteile für die Mobilfunknetzbetreiber zu erwarten. Kunden könnten die Femtozellen ständig mit sich herumtragen, um ihre günstigeren Tarife auch außerhalb der eigenen Wohnung zu nutzen. Um die Hardware einer Femtozelle zu sichern gibt es bereits bewährte Verfahren, wie den Einsatz von verschlüsselten Flashimages und ARM11 Trustzone Prozessoren. Durch SIM Karten in den Femtozellen ist zudem eine Authentifizierung durch den Mobilfunkbetreiber möglich und die erste Verbindung von Femtozelle und Kernnetz des Netzbetreibers kann durch bewährte Sicherheitsmaßnahmen z.B. IPSec garantiert werden.

Um die Kommunikation zwischen Femtozelle und Kernnetz zu ermöglichen wurden in 4.1 einige Verfahren und Lösungen vorgestellt. Die Übertragung der Gespräche und Daten wurden hierfür, um Abhörsicherheit im Internet zu gewährleisten, mithilfe von IPSec Tunneln übertragen. Auf diese Weise sind alle Daten verschlüsselt und Gespräche sowie Daten können geschützt zum Kernnetz übertragen werden. Da aber sehr viele IPSec Tunnel der Femtozellen am Gateway der Netzbetreiber ankommen, müssen diese sehr leistungsfähig sein und hohen Schutz für das Kernnetz bieten. [15]

4.7 Probleme durch die große Zahl an Femtozellen

Wie bereits weiter oben beschrieben, kommt es durch die hohe Anzahl und Dichte der Femtozellen zu einigen Problemen, wie Interferenzen und eine große Anzahl an einzelnen Verbindungen zwischen Femtozellen und Kernnetz der Netzbetreiber.

Dabei muss insbesondere daraufgeachtet werden, dass durch die adaptiven Algorithmen in der Interferenzvermeidung keine Femtozelle eine zu geringe Sendeleistung verwendet und somit seine Dienste nur in einem sehr kleinen Bereich (1 m²) um die Femtozelle anbietet. Dies würde zu verärgerten Nutzern und schlechter Publicity führen, welche eine weitere Verbreitung der Femtozellen behindert.

Ein großer Nachteil ist die schiere Zahl an Femtozellen auch deswegen, weil ein Nutzer während eines Anrufes nicht aus einer Makrozelle an die heimische Femtozelle übergeben werden kann (Handover), da die Makrozellen nicht für den Fall einer derart großen Anzahl an möglichen Übergabepartnern entworfen wurden. Die Übergabe eines Gespräches an eine Makrozelle, welches in der Femtozelle begonnen wurde, ist bei Verlassen des Hauses möglich. Die Übergabe des Gespräches an die Femtozelle beim Betreten des Hauses und die Nutzung der Konditionen innerhalb der Femtozelle dagegen ist nicht möglich. Dies liegt daran, dass das Mobilfunknetz hierfür nicht genügend unterschiedliche Cell-IDs anbietet und eine Makrozelle, bei zwei von der ID her gleichen Zellen in seinem Gebiet nicht weiß, an welche der Anruf übergeben werden soll. Zudem ist auch die Verzögerung durch die Internetverbindung ein Problem bei der Übergabe, da die Daten nicht schnell genug an eine Femtozelle übergeben werden können.

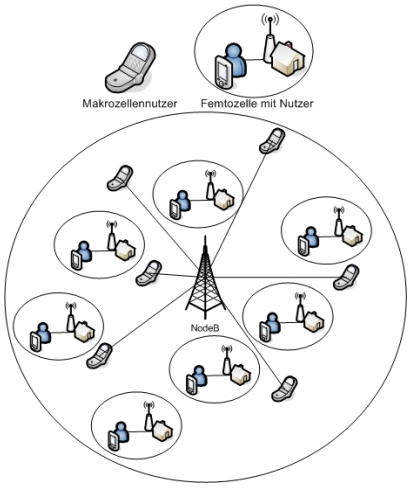


Abbildung 7: Makrozelle mit einer Vielzahl an Femtozellen in ihrem Sendebereich

5. Technologien

Aufgrund des hohen Interesses der Mobilfunknetzbetreiber gibt es eine Vielzahl von Architekturen und Chiplayouts für Femtozellen.

Hier soll beispielhaft das Referenzdesign für eine Femtozelle der Firma picoChip vorgestellt werden. picoChip ist als einer der Gründer des Femto Forums maßgeblich an der Standardisierung der Technologien beteiligt und stellt somit das beste Beispiel für eine Referenz dar. [7]

Eine Femtozelle kann dabei zum einen als Stand-Alone Lösung entwickelt werden, so dass sie an eine Breitbandverbindung über einen Router oder ein Kabelmodem angeschlossen wird. Zum anderen kann die Femtozelle jedoch auch direkt mit einem Router kombiniert werden, was zu Kosteneinsparungen aufgrund gemeinsam genutzter Komponenten auf den Platinen führt. Das auf Abbildung 8 gezeigte Design wird dabei über eine Ethernetverbindung an die Breitbandverbindung angeschlossen und stellt über die Radio Frequency Integrated Circuits (RFIC) die Mobilfunkdienste bereit. Die empfangenen Signale werden vom Digital Signal Processor (DSP), hier das picoArray, verarbeitet und schließlich über die Ethernetschnittstelle und die Breitbandanbindung ins Kernnetz des zugehörigen Netzbetreibers übermittelt. Der ARM926 Mikroprozessor ermöglicht dabei die Ausführung von in Software implementierten Protokollen und Funktionen, und reduziert somit die Anzahl an benötigten Chips. Die Basisbandverarbeitung wird dabei vom picoArray übernommen, alle Kontrollfunktionen und höheren Verarbeitungen laufen aber wie beschrieben auf dem Mikroprozessor ab. Die gesamte Femtozelle wird durch einen Flash-Speicher gestartet und ermöglicht somit Software Upgrades über das Internet (Femto-on-a-chip).

Die auf Abbildung 8 gezeigte Lösung bietet 3GPP Release 5 entsprechende Dienste wie High Speed Downlink Packet Access (HSDPA) Dienste an und kann per Software Upgrade mit High Speed Uplink Packet Access (HSUPA) nachgerüstet werden. Die Möglichkeit solche Änderungen per Upgrade nachzurüsten schafft Flexibilität, vereinfacht die Entwicklung und verkürzt die „Time-to-Market“. [7]

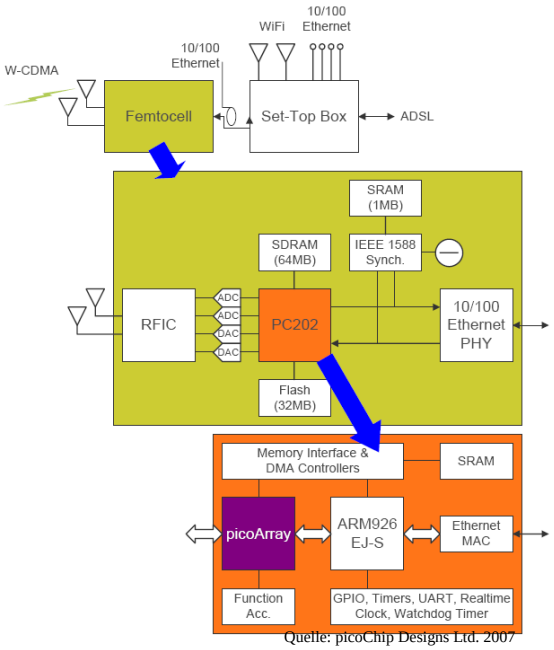


Abbildung 8: Mögliche Architektur einer Femtozelle [7] S. 13

6. Aktuelle Anwendungen

Momentan gibt es noch sehr wenige Femtozellinstallationen. Als größte ist der Femtozellentarif Airave der Firma Sprint in den USA zu nennen, die ihre Samsung Femtozellen bereits landesweit anbieten. Sprint setzt hierbei auf GPS für die Synchronisierung und die Positionsbestimmung der Femtozellen. Dabei bietet die Femtozelle in diesem Tarif drei Nutzern gleichzeitig die Möglichkeit kostenlos über die Breitbandverbindung der Femtozelle zu telefonieren. Der Handover beim Verlassen der Femtozelle funktioniert Berichten zufolge reibungslos, allerdings ist ein Handover an die Femtozelle nicht möglich, so dass der Nutzer den Anruf beenden und dann innerhalb der Femtozelle neu beginnen muss. Sollte eine vierte Verbindung im Bereich der Femtozelle getätigt werden, nutzt diese die umgebende Makrozelle. Sprint liefert die Femtozellen bereits konfiguriert aus, allerdings dauert es nach Anschluss an die Breitbandverbindung aufgrund der automatischen Konfiguration noch etwa eine Stunde, bis die Dienste genutzt werden können.

Die Kosten für die Femtozelle belaufen sich hierbei auf 75€ Anschaffungskosten, sowie weitere 3,8€ pro Monat über das erste Jahr. Insgesamt entstehen somit bei einem typischen Zweijahresvertrag Kosten von 530€, allerdings können alle Sprint-Kunden die Dienste der Femtozelle nutzen und sie ist nicht auf registrierte Nummern begrenzt.

7. Zusammenfassung

Auch wenn Femtozellen momentan noch einige technische und wirtschaftliche Hürden nehmen müssen, so ist doch aufgrund des hohen Interesses der Mobilfunknetzbetreiber und des Marktes anzunehmen, dass Femtozellen auch in Deutschland bald zu den Standardtarifen gehören werden. Denn nur durch eine zunehmende Zellverkleinerung können die gewünschten und nachgefragten hohen Datenraten den Kunden zu Hause und aufgrund der Entlastung der Makrozellen auch unterwegs zur Verfügung gestellt werden, sodass neue internetbasierte Dienste auf mobilen Endgeräten möglich werden.

Die Standardisierung wird stark vorangetrieben, wobei hier insbesondere das Femto Forum als Zusammenschluss aus mittlerweile fast der gesamten Mobilfunkindustrie zu nennen ist. Die Funktion des Femto Forum beschränkt sich jedoch darauf koordinierte Verhandlungen in der Mobilfunkindustrie zu ermöglichen und Vorschläge zu tätigen. Diese haben aufgrund der Anzahl und Wichtigkeit der Mitglieder hohes Gewicht. Die Entscheidungsgewalt bezüglich neuer Standards liegt weiterhin bei der 3GPP, dessen Partner das Femto Forum ist. [6]

Die hier vorgestellten technischen Probleme sind nicht unbedingt trivial zu lösen, doch zeigen bereits durchgeführte Studien der Netzbetreiber, dass es durchaus möglich ist Femtozellen sinnvoll einzusetzen. Da diese Studien noch nicht die große Anzahl der Femtozellen und die damit entstehenden technischen Probleme berücksichtigen können, ist abzuwarten, ob Marktprognosen, welche 32 Millionen Femtozellen mit 102 Millionen Nutzern weltweit bis 2011 vorhersagen, eintreffen werden. [7]

8. Literatur

[1] Airvana (2007), 'Femtocells: Transforming The Indoor Experience', White Paper.

[2] Aricent (2008), 'Challenges in Deployment of UMTS/HSPA Femtocell', White Paper.

[3] Broadband Forum (2007), TR-069 CPE WAN Management Protocol v1.1', Technical Report

[4] Bundesrepublik Deutschland (2007), 'Telekommunikationsgesetz - §108 Notruf'

[5] Claussen, H. (2007), Performance of Macro- and Co-Channel Femtocells in a Hierarchical Cell Structure, *in* 'IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007.'

[6] Femto Forum (2007), 'Femto Forum Intro', http://www.femtoforum.org/femto/Files/File/Femto%20Forum%20Intro.pdf , zugegriffen: 27.09.2008.

[7] picoChip Designs Ltd. (2007), 'The Case for Home Base Stations', White Paper.

[8] picoChip Designs Ltd. (2007), '3GPP Long-Term Evolution: fit or flawed?', White Paper.

[9] Glenn LeBrun, Andrew Bender, (G. (2008), 'The Future Role of Media Gateways in All-IP Networks', White Paper.

[10] Kineto Wireless, Inc. (2007), 'UMA: The 3GPP Standard for Femtocell-to-Core Network Connectivity', White Paper.

[11] Lester T. W. Ho, H. C. (2007), Effects of user-deployed, co-channel femtocells on the call drop probability in a residential scenario, *in* 'PIMRC 2007. IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, 2007.'.

[12] Nextpoint Networks, 'Connecting when it counts: The role of femtocells in emergency calls', White Paper.

[13] Nextpoint Networks, 'Fixed-Mobile Convergence Border Architecture', White Paper.

[14] Nextpoint Networks, 'Integrated Border Gateway', White Paper.

[15] Sanjay Bhatia (Genband), (2008), 'Femtocells Challenges and Opportunities', White Paper.

[16] Vikram Chandrasekhar, Jeffrey G. Andrews, Alan Gatherer (2008), 'Femtocell Networks: A Survey', IEEE Communications Magazine, vol. 46, no.9, Sept. 2008.

Network Traffic Visualization

Fabian Popa
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
fabian.popa@mytum.de

ABSTRACT

This paper discusses Network Traffic Visualization, covering the motivation, challenges, and different approaches. In starts off by analyzing the relevant types of data needed, as well as collection methods employed in generating data sets. It then moves on to the ways in which these sets of network traffic information are mapped onto an image, in order to accentuate specific characteristics of the traffic. Sample architecture is presented for a distributed information collection system. In the second part of the paper, the focus shifts toward real-world systems and methods currently in use, providing sample outputs and emerging benefits. Finally, the paper extends a conclusion on the visualization methods discussed, revisiting specific applications, and glimpsing into the future of network traffic visualization.

1. INTRODUCTION

Computer networks, and specifically the Internet, are the foundation and enabler of our fast-moving information society. In our day to day lives, we are explicitly and implicitly relying on computer systems, connected in networks that range from home or corporate LANs (Local Area Network) to global and ever growing WANs (Wide Area Network). Weak, unstable networks, as well as unpredictable network activity, can severely damage or hold back operations of all magnitudes. But how can we make sure that a network is not vulnerable? This is where network traffic information comes in.

The analysis of network traffic data can provide indicators about the state of a network. By monitoring the network and looking for specific markers, anomalous behavior can be identified and addressed in a timely manner. Unfortunately, not all data bears relevance, and, while readable by a machine, it is usually difficult to interpret, in its raw form, by a human. Therefore, the right data has to be found and then converted into an accessible form.

Toward this purpose, methods have been devised to visually represent traffic information, making it human-readable, but also analyzable from a different standpoint, that of image processing. Weak points in the network, and malicious behavior such as DDoS attacks and scanning activities, are specifically targeted and highlighted. There are multiple reasons for visualizing network traffic information, which will be discussed in the following.

1.1 Motivation

Visualizing information is a technique that can encode large amounts of complex interrelated data, being at the same time easily quantified, manipulated, and processed by a human user. Therefore, it is an obvious candidate for the representation of network traffic information. Using specific techniques, the state of the network can be depicted in such a way that anomalous activities will display as objects inside the traffic image. These can be identified by image-processing algorithms, as well as the system administrator looking at the image. Furthermore, images can be compressed, enabling size reductions and faster communication and analysis of the data.

Indeed, not only information on anomalous and malicious activities is desired, but also on the connectivity and performance of the system. In this case, topological and geographical representations bear more meaning. In a topology map or network graph, specific measures can be applied, such as the critical paths between two subnets, or the shortest path between two peers. Furthermore, looking at a single node or subnetwork, notions like "reachability" are highly important and can be measured and displayed in an intuitive manner using a topological approach.

In the end, network traffic visualization aims at providing a clear overview of the state of a network or subnetwork, in a way that aids system administrators and network architects in maintaining the integrity, availability, and reliability of the network, as well as plan for capacity increases, new communication protocols, and expansions.

1.2 Challenges

Traffic visualization is effective in dealing with specific network issues, as studies discussed later in this paper show, but it faces important challenges, mostly due to the sheer size of the Internet.

Monitoring activities produce large volumes of data, which need to be efficiently communicated, stored, and processed. These become an even bigger concern when real-time representation is desired, although this is not always the case.

Furthermore, when providing visualizations for the human reviewer, they should be efficient and easy to understand, without the need for lengthy in-depth analysis. The right method and the right data for a specific task have to be found and combined to reveal the important aspects visually.

The structure, technology, and bandwidth of networks are changing at a fast pace. It is increasingly difficult to measure large networks, let alone analyze the traffic.

When looking at changes over a period of time, differences in traffic states need to be accentuated as clearly and effortlessly as possible.

It is important to choose the right data and methods of processing, but it is equally important to look at the right data. Finding the best combination of the two is not a light task.

2. METHODOLOGY

Network traffic data is acquired in a single or distributed environment, by different means. Specific to the application, it is then communicated and processed or directly processed, at which point visualization is employed. The resulting images can be passed on, archived, or further analyzed programmatically (e.g. object recognition in the image).

Different participants in the provision of the Internet Service have different concerns and would look at different network information. For instance, an ISP (Internet Service Provider) will look at data regarding the usage of the service by its users, and aim at optimally filling its network capacity and choosing its network partners (other ISPs or large network providers). In this process, it would consider the bandwidth and types of content

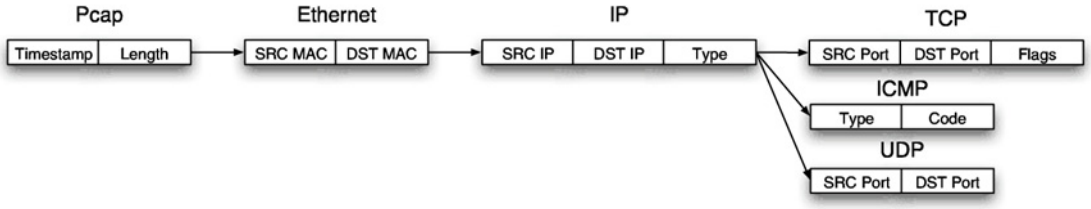


Figure 1. Packet header fields parsed for statistics gathering [3]

consumed by the users and reasonably plan capacity and pricing models. On the other hand, the regular Internet user might judge the performance of his or her connection only by looking at how promptly a given website is loading. While this may be a more naïve approach, it is nevertheless valid.
How would we go about measuring the traffic of a network properly? The steps to acquiring and preparing the data for visualization are discussed below.

2.1 Types of data

Some types of traffic data bear more meaning than others. Before we can look at how to obtain the required data, first we need to identify the right data for our purpose. To analyze behavior inside a network, the following traffic characteristics are typically considered:

- **Packet Header Fields (D1)**:
  source/destination IP, source/destination port, time to live (Figure 1). The packet header information characterizes the flow of the data packet through the network (direction, time). It provides parameters for visualization methods and statistical procedures.
- **Round Trip Time (D2)**:
  time elapsed for a packet to reach a destination address and return to the sender. This is a network connectivity and performance indicator.
- **Packet Hop (Routing) (D3)**:
  the route of a packet through the network nodes. The communication route between two peers can differ in one direction from the other. Therefore, the data packet may be directed on a longer path in one direction and may take longer to reach its destination.
- **Bandwidth (D4)**:
  bandwidth consumption for incoming/outgoing traffic indicates where the activities taking up the most resources are occurring. Consequently, based on the destinations/sources of the traffic from/to one address, network attacks can be identified and restricted.
- **BGP Tables (D5)**:
  a network router looks at the destination IP address of the data packet and uses a BGP (Border Gateway Protocol) Table of addresses to figure out the next hop toward the target.

There are additional metrics as well, but the aforementioned provide a sound depiction of the network state.
The Internet data cannot be recorded in its entirety, due to the giant scale. Therefore, it needs to be sampled and looked at over a specific timeframe. Now we need to choose a method to acquire the data.

2.2 Data acquisition

The basis for visualization is the data set, collected over a given timeframe. There are two ways in which traffic data can be collected:

- **active means**:
  specially crafted traffic is introduced into the network. It can be observed at receivers and it can trigger a response, which returns to the sender. Sample applications include estimating the bandwidth of an Internet link and determining the path connecting two computers.
- **passive means**:
  data is collected at strategic locations, without introducing new traffic into the network. Sample applications include "telescopes", listening to incoming packets and requests (traffic with no legitimate destination).

"In typical (and simple) cases, the *active* measurements can be used for direct quality investigation of end-to-end communications (traffic performance), while the *passive* method is used to collect figures for network-internal statistics (traffic load, traffic matrix, etc)". [10]
When looking at measurements regarding the Internet, simple approaches can often not be employed. Here, the need for distributed systems becomes evident. Such architecture is presented in the following paragraph.
With regard to the global Internet infrastructure, CAIDA [4] provides tools and analyses promoting the engineering and maintenance of robustness and scalability. They also provide measurement, topology, and routing data sets, which are the basis for some visualizations discussed later on.

2.3 System structure

Designing an effective data acquisition system can determine the speed, accuracy, and, most of all, reach of traffic measurements.
In a simple setting, a network survey (active means), for example, can be conducted from one location where traffic is sent from this one source to many destinations, recording the results. Figure 2 depicts such a setting.

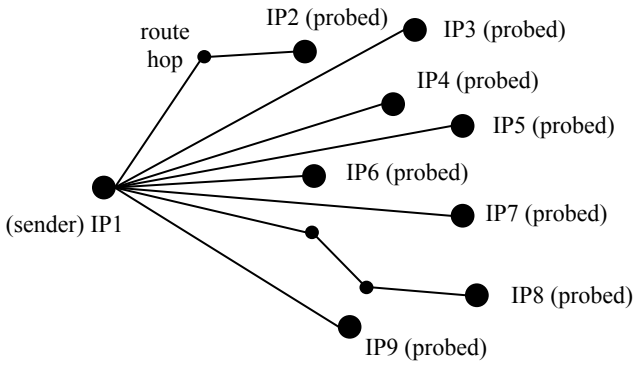


Figure 2. One location network survey. [no ref.]

However, given the sheer size of the Internet, even a straightforward measurement can require multiple senders. In practice, more frequently employed are complex distributed systems. This applies for passive data collection as well. Telescopes, for instance, could be installed in one or more locations, depending on the size or reach of the network. The bigger the scope (telescope's "lens"), the more accurate the inference about the overall state will be.
The configuration of a distributed system (Figure 3) typically contains the following components, although some modules are specific to a visualization technique discussed later on (2D plane conversion, Space-Filling Curves):

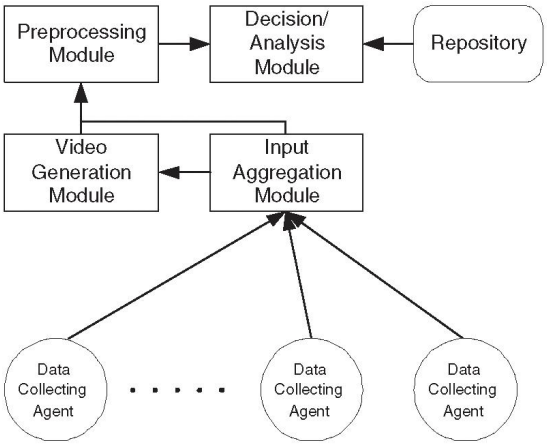


Figure 3. Distributed system design. [1]

a) **Data Collection Agent**:
these are distributed sensing modules that collect real time statistics from the locations they have been installed in. The gathered data is processed on-site. Here, the initial visualizations (images) are generated over a given time period, then aggressively compressed to save bandwidth. The images are sent to the aggregation module.

b) **Aggregation Module**:
after receiving the data (i.e. compressed images) from the agents, it is prepared for analysis. Normalization and synchronization operations are carried out, so that a consistent data set can be passed on to the video generation module.

c) **Stream (i.e. Video) Generation Module**:
the incoming multiple streams of data are converged into a single stream, that depicts the state of the system as a whole (i.e. video composed from the images received over successive timeframes). Information resulting from the formation process, such as peak intensity locations, is passed along.

d) **Preprocessing Module**:
objects in the stream (i.e. video) are identified and their characteristics are recorded (location, trajectory, speed, brightness, shape, size, etc.)

e) **Decision/Analysis Module**:
based on the information provided from the previous modules, it can be inferred if the activity is normal or anomalous. In the decision process, the engine draws historic information from a repository, where known patterns, special cases, and system history are saved. The Decision Module can automatically take action or notify the administrator of certain changes or happenings.

The basic architecture of collection (single/multiple location), preprocessing, analysis, and repository is present in one form or another in every distributed network traffic analysis system.
Looking at the network traffic visualization system as a whole, the Aggregation Module concludes the first step (data acquisition), by outputting the raw data set. The data can be processed automatically as-is, or it can be visualized. Visual data, as mentioned before, can be handled not only by a human reviewer, but also by image processing algorithms.
In the paragraph following, different traffic measurement and visualization systems are discussed and their imaging approaches evaluated.

3. VISUALIZATION

Different visualizations are adequate for different tasks, and work best with specific data sets. For the imaging (mapping) of Internet Address Space, 2D plane conversions have been heavily discussed, mostly because of two reasons: they provide a comprehensive overview of the state of the Internet (they can map the whole IPv4 address space), and they retain traffic properties when certain mapping approaches are used and the resulting images compressed. This achieves bandwidth and processing time savings. On the other hand, when looking at issues such as address reachability, graph-based representations are preferred. Furthermore, for human readability, geographical map overlays are popular, in conjunction with more freely-chosen, but still adequate, visualization techniques. For each of these categories, we will now discuss methodology and applications. We will follow the structure discussed: data set, collection method, visualization, inference (benefits).

3.1 Ant census of the Internet Address Space

Starting in 2003, researchers at ISI [5] have been collecting data about the Internet. As part of this work, they have been probing all addresses in the allocated IPv4 Internet Address Space for their reachability. This is a type of active "one location network survey" application (Figure 2). The researchers at the ISI have sent an ICMP ping message to all the addresses in the IPv4 address space. ICMP ping is an "echo request" packet. If the packet reaches its destination, it will trigger an ICMP "echo response". The sender listens for this reply, and records Round Trip Time (**D2**), as well as packet loss. In this case, the quality of response from a destination IP constitutes the data set.
Naturally, because the IPv4 (Internet Protocol, version 4) provides 2^{32} (around 4 billion) addresses, choosing the right way of visualizing the address space is of great importance.
The researchers have chosen a layout first suggested in the popular web-comic xkcd [6]. Here, the one-dimensional, 32-bit address space is converted into two dimensions using a Hilbert Curve, as shown in Figure 4.

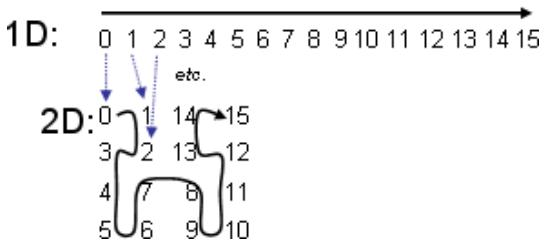


Figure 4. Hilbert curve 1D to 2D conversion. [5]

The Hilbert Curve was first described by the German mathematician David Hilbert in 1891, and presents a number of properties relevant to the mapping of IP Address Space. First, it is space-filling, which means that it will fill a "square unit" entirely (if the 1D data has 2^n points, the resulting 2D image is always square). Secondly, it is fractal, which allows zoom-in and zoom-out without resolution loss. Thirdly, it is continuous, meaning that consecutive points in 1D will be consecutively mapped in 2D (never breaking the curve). And fourth, and most important, it preserves locality. The curve keeps adjacent addresses physically near each other. Subnets will be visually represented as clusters. The Hilbert curve bears strong benefits to Internet Traffic Visualization and is discussed in paragraph 3.2 as well, but in a different application.

Internet addresses are allocated in blocks of consecutive addresses. The map constructed by ISI [5] shows who controls each of the 256 numbered blocks corresponding to /8 subnets ($2^{32}/2^8$ = ~16 mil. addresses). Block number *n* represents the 16 million addresses of the form *n*.-.-.- .

As an example, Figure 5 shows a subset of the ISI Internet map. Two blocks (196/8 and 199/8) are allocated geographically, while 198/8 is used by many groups, and 197/8 is still unallocated.

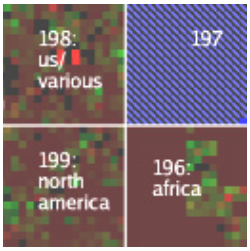


Figure 5. Subset of the Internet allocation map. [5]

Geographic allocations can reduce routing table sizes (**D5**), and round trip time (**D2**), delivering better network performance. The map provides a good overview in planning for new address allocations, by depicting the taken and available blocks. Given that it is a "one location network survey", the map also depicts the reachability of all the Internet from a specific source. If that source is an important content provider, for instance, it might be worth such an investigation.

When visualizing network information through Space-Filling Curves (3.2), in this case Hilbert Curves, color coding is usually employed to depict the data. In summary, Hilbert Curves map the Internet Address Space, and the color intensity of each pixel infers the amount/quality of data for the subnet that the pixel represents.

In this ISI visualization, each point's color coding depicts the average ICMP "echo response" of a /16 subnet (65.536 addresses). The brighter the point, the more replies were received.

3.2 Space-Filling Curves Mapping

The Hilbert Curve is a Space-Filling Curve (SFC). Figure 6 shows other SFCs, which can be employed in network data visualization. The mapping of 1D data to a 2D plane using SFCs is thoroughly discussed in [1]. In the paper a set of different SFCs are utilized to map the statistics collected to images that emphasize traffic patterns. Anomalies such as large scale DDoS (Distributed Denial of Service) attacks and scanning activities are identifiable, due to the enhanced locality of SFC clustering.

The Hilbert Curve bears one more advantage, which is explained in the paper [2]. Aggressive compression can be applied on the resulting Hilbert images, without major loss of traffic information, but with high savings in space and bandwidth, outperforming other SFCs.

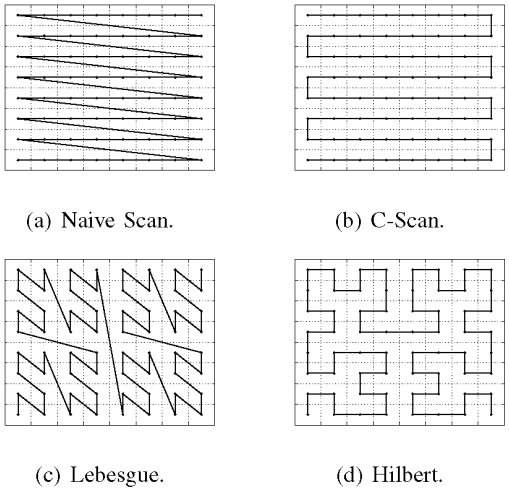


Figure 6. Space-Filling Curves. [1]

The paper [1] puts forward a visualization method for detecting anomalous network activities with the help of SFC mapping. In a distributed architecture (Figure 3), statistics based on packet header fields (**D1**) are obtained by the collection agents. The paper does not directly specify the data collection means, but it can be inferred that passive means are employed, as new traffic is not introduced into the network, but it is rather listened for in incoming packets. The data set is constituted by the number of packets arrived during an investigated time slot. Incoming packets are sorted, based on their source IP's lowest byte, into 256 classes. The paper points out that the lowest byte is the one with "the most interesting characteristics". The 256 classes are the 1D space, while the number of packets in each class will determine the color intensity.

Figure 7 shows a comparison of normal traffic and anomalous traffic, using this visualization data and method.

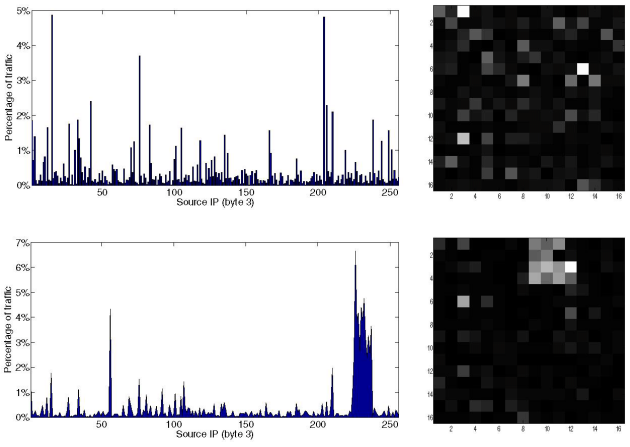


Figure 7. Sample histogram and Hilbert images of normal(top) and anomalous (bottom) traffic activity [2]

This method was chosen because of the curves' ability to preserve locality in converting traffic to intensities. The paper stresses that locality is a crucial in making attacks such as address scanning or DDoS distinguishable, "even if they are dispersed and not perfectly contiguous". This feature can be clearly observed in Figure 7, in the histogram, where most of the traffic comes from the range 225-240, as well as in the image, where due to the locality of the classes 225-240, a visual object has been formed in the cluster of classes.
Such visual objects will withstand aggressive compression, as it is pointed out in [2]. The size reductions of the images will help in processing large scale traffic, but also in communication between agents and the central analysis engine.
The visualization method can be extended to include more than one packet header characteristic. In that case, the intensity value of each pixel on the image would the number of data samples that contained the set of values <v1,v2,v3,…,vn>. In the previous example, the set of values contained, in fact, one value, the source IP of the incoming packets (<sourceIP>). Possible data sets would be <sourceIP,destIP>, or <sourcePort,destPort>, or <sourceIP,sourcePort,destIP,destPort>, etc.

The visualization procedure is the following:

1. the data set (i.e. packet IP headers) is recorded by the sensing agents in their specific locations and principal fields are extracted (source/destination addresses and ports, etc.)
2. the timeframe data is partitioned into windows (i.e. 2 min each).
3. for each window, the histogram is calculated (frequencies of each field value tuples). This is where the data is serialized (n-D to 1-D). Serialization means that different n-D tuples are numbered in a consecutive order (<v1,v2,v3> is 1, <v1,v2,v4> is 2, etc.).
4. the image is generated for each window, mapping the pixel intensities from the frequencies in the histogram using SFCs (1-D to 2-D). The serialized tuples each get a pixel on the image, which will be colored with an intensity directly related to the number of occurrences of that given tuple in the time window.

On the efficiency of SFCs in Network Traffic Visualization and anomaly detection, the paper [2] offers a discussion, based on a DDoS attack sample data set.
Analyzing the errors resulting from lossy compression of the images obtained with SFCs, it concludes that using space-filling curves, especially Hilbert mapping, has very small traffic data loss upon aggressive compression. It goes on to point out that, even after decreasing the space requirements of the image by a factor of 1024, the Hilbert mapping made the image robust enough and capable of resisting the ruining degradation of quality. The Hilbert curve has consistently the lowest error of all the mappings. Image sizes resulting from compressing Hilbert achieved smaller errors, while guaranteeing the smallest sizes.
Considering the DDoS trace analyzed in [2], the original space used to store the traffic information was ~3.5 MB, while the compressed and then 4x downscaled images required only ~1 KB (excluding the common header of the images). However, this storage saving comes at the expense of a small error value, and loss of individual packet information.
We can now see the importance of Space-Filling Curves in the visualization of network traffic information, and specifically, the strongly positive properties of Hilbert Curves. The methods discussed in this paragraph produce images which make anomalous network activities or attacks visible to the naked eye, as well as to image-processing algorithms, which can identify and process the visual objects.

3.3 Skitter

Moving away from 2D plane conversions, we will discuss Skitter data, a graph-based visualization tool and data set developed by CAIDA [8] for actively probing the Internet in order to analyze topology and performance. Its specific goals include the measuring of Forward IP Paths (**D3**), Round Trip Time (**D2**), and persistent routing changes (**D5**). It also offers a visualization of network connectivity, such as in Figure 8.
Skitter employs active means in measuring network characteristics. It is a type of "one location network survey", which records the unidirectional IP path from the source to multiple destination IPs probed. A unidirectional path is the path a packet takes from the source to the destination IP, passing through other IP devices along the way. No routing is imposed on the packets by the source, so that the network will determine the route the Skitter probes take.
Skitter accomplishes its goal in an ingenious manner. Because it would not receive responses from each IP device that the packet travels through on the way to its destination, the application is, in fact, probing each hop along the path by incrementing the time-to-live (TTL) in the IP packet header (Figure 1). This way, every hop will send a ICMP TIMEXCEED message in response to a packet with an expired TTL. The application receives the message, increments the TTL and sends it this time directly to that hop (IP device). The packet will get diverted to another hop which will send the ICMP TIMEXCEED and so on, until it reaches its destination.
The visualization Skitter uses is a graph-based one, where communication is depicted as a link between nodes. In this sample, data is plotted onto a globe in 3D, for easier handling.

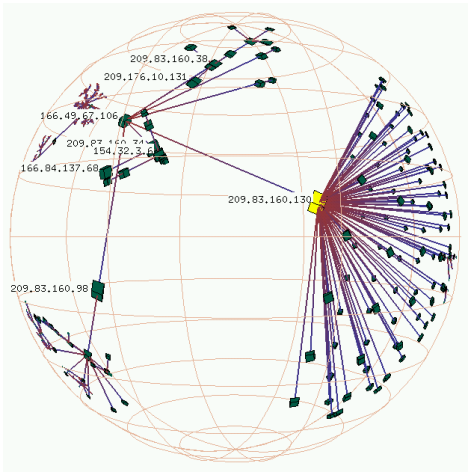


Figure 8. Sample visualization from Skitter data. [8]

The yellow square in Figure 8 is the source IP address. All other nodes are hops or destinations.
Skitter has been used to identify critical paths (i.e. routers or network nodes that might be vulnerable points), and to map dynamic changes in Internet topologies (by looking at Skitter data collected over time).

3.4 Akamai real-time Web Monitor

Finally, we take a look at geographical overlays and examples of nonconventional visualization methods.
Akamai is one of the largest CDNs (Content Delivery Network) in the world. They monitor their global private network around the clock. With this real-time data, they can identify the global

regions with the greatest attack traffic, cities with the slowest Web connections (latency), and geographic areas with the most Web traffic (traffic density).

In the case of attack traffic, Akamai collect data on the number of attempted connections, and on source and destination IPs and Ports (**D1**) of packets flowing through their network (passive measurement). Malicious activity generally comes from automated scanning trojans and viruses, searching for new computers to infect, by randomly inquiring IP addresses. The number of attacks in the last 24 hours in a region is depicted by tones of red of varying intensity covering that specific region on the map.

The network latency between most major cities is measured via automated scripts. These are tests consisting of connections, downloads, and ICMP pings (active measurement). Two latency quantifications are provided (in milliseconds), as vertical bars, for each monitored city: the absolute current latency, and the relative latency, in comparison with its historical average latency.

The third visualization, network traffic, is the amount of data being currently requested and delivered, by geography. This is perhaps the most important measurement for Akamai, as it is the basis of their revenue. The values are provided as percentages of global network traffic. Again, a color overlay is used, in relating the metric to different countries.

Figure 9. Akamai in the last 24 hours: network attacks (top), latency (middle), traffic (bottom).[7]

Although the data sets are significant, they are still limited to a private, more controlled network, and can only suggest the state of the whole Internet.

The geographical overlays and maps are used to clarify the data and present it in an appealing way. Regions are displayed as countries.

4. CONCLUSION AND OUTLOOK

Monitoring traffic activity is a necessity in ensuring the health of networks. Traffic Visualization encompasses the tools and metrics necessary for such a task. In this paper, we have broken down visualization systems and looked at types of data relevant for specific tasks, data acquisition and visualization methods, as well as real-world examples, usage scenarios, and benefits.

The current techniques, albeit effective, can only be applied on small networks for rapid information provision.

Improvements in processing power, and advances in analysis, modeling, visualization, and simulation tools, particularly those capable of addressing the scale of the Internet, will enable system administrators and network architects to plan for the next-generation Internet, a safer, more reliable and interconnected place.

5. REFERENCES

[1] T. Samak, A. El-Atawy, E. Al-Shaer, and M. Ismail. A novel visualization approach for efficient network-wide traffic monitoring. End-to-End Monitoring Techniques and Services, 2007. E2EMON apos; 07. Workshop on Volume , Issue , Yearly 21 2007-May 21 2007 Page(s): 1 - 7

[2] T. Samak, S. Ghanem, and M. Ismail. On the efficiency of using space-filling curves in network traffic representation. Computer Communications Workshops, 2008. INFOCOM. IEEE Conference on Volume , Issue , 13-18 April 2008 Page(s): 1 - 6

[3] K. Abdullah, C. Lee, G. Conti, and J. Copeland. Visualizing Network Data for Intrusion Detection. Proceedings of the 2002 IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, 17-19 June 2002

[4] CAIDA: Cooperative Association for Internet Data Analysis. http://www.caida.org

[5] Information Sciences Institute (ISI) at the University of Southern Carolina: ANT censuses of the internet address space. http://www.isi.edu/ant/address/

[6] xkcd: A webcomic of romance, sarcasm, math and language. #195 Map of the internet. http://www.xkcd.com/195/

[7] Akamai Real-time Web Monitor http://www.akamai.com/html/technology/dataviz1.html

[8] CAIDA Skitter http://www.caida.org/tools/measurement/skitter/

[9] B. Irwin, N. Pilkington. High-level Internet Traffic Visualization using Hilbert Curve Mapping. VizSEC ’07 Presentation – 29 October 2007.

[10] Information Society, Traffic Measurement and Monitoring Roadmap http://www.ist-mome.org/documents/traffic_ngni.pdf

Verkehrscharakterisierung durch Methoden des maschinellen Lernens

Benjamin Wiesmüller
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
benny.w@mytum.de

KURZFASSUNG

In dieser Arbeit werden Ergebnisse aktueller (2004-2006) Forschungsarbeiten zur Internet-Verkehrscharakterisierung mit Hilfe von Methoden des maschinellen Lernens vorgestellt. Zu Beginn folgt in der Einleitung ein kleiner Überblick über Anwendungsmöglichkeiten für Verkehrscharakterisierung und es werden bisher dafür verwendete Verfahren und deren Probleme vorgestellt. Anschließend wird einen Einstieg in das Thema maschinelles Lernen mit Definitionen und Überblick über verschiedene Methoden geboten, speziell im Hinblick auf die Verkehrscharakterisierung. Danach folgen Zusammenfassungen von drei konkreten Forschungsarbeiten zum Thema mit unterschiedlichen Verfahren zur Verkehrscharakterisierung mit:

- Clustering (Unsupervised Learning),
- Expectation Maximization (Unsupervised Learning),
- und Classification (Supervised Learning).

Schlüsselworte

Verkehrscharakterisierung, Maschinelles Lernen, Clustering, Classification

1. EINLEITUNG

In dieser Arbeit, werden Ergebnisse aus aktuellen Forschungsarbeiten ([1],[2],[3],[4],[5]) zur Netzwerkverkehrs-charakterisierung zusammengefasst und einige der möglichen Anwendungen und verwendeten Techniken vorgestellt.

Zunächst was ist mit Verkehrscharakterisierung genau gemeint? Es sollen an einem Knotenpunkt, z.B. dem Internetzugang eines Unternehmens, die verschiedenen Verbindungen untersucht und anschließend automatisch kategorisiert werden. Beispielsweise könnte man sie zu einer konkreten Anwendung, oder einer bestimmten Verhaltensklasse zuordnen. Dies kann für Netzwerkadministratoren aus verschiedenen Gründen nützlich sein: Netzüberwachung, Optimierung der Netzwerkarchitektur, Analyse des aktuellen Verkehrs, Simulation für verschiedene Netzbelastungen, Quality of Serivce (Priorisierung von Verbindungen), Sicherheitsaspekte (Erkennen von verdächtigem Netzwerkverkehr), etc.

Bisher wurden hauptsächlich zwei Vorgehensweisen verwendet um den Verkehrsfluss zu charakterisieren: Analyse der Paket-Header und des Paket-Payloads. Aus dem Header eines Pakets kann man IP-Adresse und Portnummer, jeweils von Ziel und Quelle, sowie das verwendete Protokoll erkennen. Eine Möglichkeit eine Verbindung nun zu einer Anwendung zuzuordnen, ist mit Hilfe der Portnummer, über die von der IANA festgelegten Well-Known- oder anderweitig bekannten Ports von Anwendungen. Dies ist zwar einfach und schnell möglich, aber bei einigen neuen Anwendungen kaum mehr aussagekräftig [1]. Handelt ein Programm den verwendeten Port dynamisch aus, oder tunnelt ihren Daten gar über einen der Well-Known-Ports, so ist eine Erkennung anhand der Portnummern nicht mehr möglich. Die Analyse des Paket-Payloads kann dieses Problem umgehen, indem in der Nutzlast eines Pakets nach typischen Signaturen einer Anwendung gesucht wird. Jedes Paket so zu analysieren stellt aber hohe Anforderungen an Speicherbedarf und Rechenleistung, deswegen sind dafür auch spezielle Hardwarelösungen erhältlich. Außerdem stößt auch dieses Verfahren an seine Grenzen, wenn die Nutzlast verschlüsselt wird [1].

Die hier zugrunde liegenden Forschungsarbeiten untersuchten die Verwendung und Möglichkeiten von Methoden des maschinellen Lernens um diese Probleme zu umgehen und den Verkehr zuverlässig zu charakterisieren.

Die Arbeit ist im weiteren wie folgt eingeteilt: In Kapitel 2 folgt eine Überblick über maschinelles Lernen mit Hinblick auf die Verkehrsklassifizierung. In den nächsten drei Kapiteln werden die konkreten Verfahren vorgestellt. Kapitel 3 geht dabei über die Verwendung von Clustering (Unsupervised Learning) zur Zuordnung einer Verbindung zu einer Anwendung, mit den Algorithmen k-Means und DBSCAN. Kapitel 4 stellt ein Verfahren für eine schnelle Charakterisierung, nur anhand der ersten paar Pakete, vor, wobei die Expectation Maximization-Verfahren Gaussian Mixture Model, Hidden Markov Model und erneut k-Means verwendet werden. Kapitel 5 zeigt die Verwendung von Classification (Supervised Learning) um Verbindungen einigen vordefinierten Quality-of-Service-Klassen zuzuordnen. Dabei werden Nearest Neighbour und Linear Discriminant Analysis verwendet. Kapitel 6 schließt mit einem Fazit ab, das auf Möglichkeiten und Probleme der neuen Verfahren eingeht.

2. WAS IST MASCHINELLES LERNEN?

2.1 Definition

DIE festgelegte Definition von maschinellem Lernen gibt es nicht, deswegen findet man oft leicht unterschiedliche Definitionen dafür.

In [6] findet sich, wiederum geliehen von der englischen Wikipedia, übersetzt folgende Definition:

Maschinelles Lernen ist ein Untergebiet der künstlichen Intelligenz, die sich mit der Entwicklung von Algorithmen und Techniken beschäftigt, die dem Computer das „Lernen“ beibringen. Das induktive maschinelle Lernen extrahiert Regeln und Muster aus großen Datensätzen. Der Fokus der Forschung beim maschinellen Lernen liegt hauptsächlich darauf automatisch, mit rechnerischen und statistischen Methoden, Informationen aus Daten zu extrahieren.

Im Hinblick auf die Techniken die in dieser Arbeit erklärt werden, soll dass heißen, dass die Daten anhand verschiedener Attribute automatisch in verschiedene Kategorien/Gruppen eingeteilt werden sollen, indem Muster in diesen Attributen erkannt werden. So werden z.B. einzelne TCP-Verbindungen

anhand der Paketgrößen, Inter-Arrival-Time, Verbindungsdauer etc., zu den zugehörigen Anwendungen zugeordnet.

Üblicherweise werden beim maschinellen Lernen (mindestens) folgende drei Arten unterschieden [6]:

Überwachtes Lernen (Supervised Learning):

Für gegebene Eingaben werden dem lernenden System auch die dazugehörigen gewünschten Ausgaben geliefert. Das Ziel ist es korrekte Ausgaben für nicht weiter benannte Eingaben zu produzieren.

Hierbei sind die Eingaben also schon mit den zugehörigen Gruppenbezeichnungen versehen. Das System soll nun Muster erkennen anhand derer es zukünftige Eingaben unbekannter Zuordnung korrekt einordnen kann.

Unüberwachtes Lernen (Unsupervised Learning):

Hier ist das Ziel ein Modell aus den Eingaben zu erstellen ohne im Voraus zu wissen wie diese geartet sein sollen.

Das System soll hier anhand bestimmter Attribute der Daten diese zu ähnlichen Gruppen zusammen fassen. Hier wird häufig dem System die Anzahl der gewünschten Gruppen vorgegeben.

Bestärkendes Lernen (Reinforcement Learning):

Anstatt nur einfacher Ausgaben/Zuordnungen, sollen hier Aktionen produziert werden, die den Zustand der „Welt“ des Systems verändern. Je nach Aktion erhält das System „Belohnungen“ und es soll lernen diese zu maximieren (in dieser Arbeit nicht weiter von Bedeutung).

2.2 Unsupervised Learning

Beim Unsupervised Learning wird das Ermitteln der Gruppen, in die die Daten eingeteilt werden, Clustering genannt. Die entstehenden Gruppen entsprechend Cluster. Dies geschieht meist offline in einer Trainingsphase, bei der Trainingsdaten als Eingabe für den Algorithmus benötigt werden. Dazu werden sogenannte Traces verwendet. Dabei handelt es sich um Aufzeichnungen von Netzwerkverkehr, wobei Paket-Header-Traces, die nur die Header der IP-Pakete enthalten und Payload-Traces, die auch die Nutzlast enthalten, unterschieden werden. Um an solche Traces zu gelangen kann man z.B. frei erhältliche Aufzeichnungen des Internetverkehrs einiger Universitäten verwenden. Anschließend kann man die Traces mit diversen Tools analysieren, z.B. gibt es Programme die Payload-Traces analysieren können und so ermitteln zu welchen Anwendungen die jeweiligen Pakete gehören – mit entsprechendem Zeitaufwand. So kann man bestimmen welche Arten von Verbindungen sich in einem der entstandenen Cluster befinden. Wie schon in der Einleitung erwähnt stoßen solche Verfahren aber unter Umständen an ihre Grenzen, z.B. bei verschlüsselter Nutzlast. Eine mögliche Lösung in einem solchen Fall trotzdem geeignete Trainingsdaten zu erhalten, wäre manuell einen Trace zu erzeugen, z.B. den isolierten Verkehr einer einzelnen Anwendung aufzuzeichnen, und diesen als Eingabe zu verwenden [1].

Bevor man den Clustering-Algorithmus starten kann, muss man aus den Traces noch eine geeignete Repräsentation der Daten erzeugen. Wie schon angedeutet muss man sich dafür zunächst entscheiden welche Attribute der Daten betrachtet werden. Diese Auswahl unterscheidet sich je nach gewünschtem Ziel und wird in den jeweiligen Kapiteln zu den einzelnen Methoden erläutert.

Die spätere, meist online bei laufendem Verkehr stattfindende, Zuordnung von neuen Verbindungen zu den vorher entstandenen Clustern wird Classification genannt. Diese ergibt sich meist einfach aus den verwendeten Algorithmen, kann aber auch weitere Tricks verwenden um die Ergebnisse zu verbessern.

2.3 Supervised Learning

Beim Supervised Learning sind die Cluster und die Zuordnung der Eingabedaten zu diesen vorgegeben. Man spricht dann meist von Classification-Algorithmen. Nachteile solcher Verfahren sind, dass neue unbekannte Anwendungen nicht erkannt werden können [3]. Will man aber alle Eingaben von vornherein nur in eine feste Anzahl an Cluster einordnen, kann man diese durch ausgesuchte Eingabedaten charakterisieren, ohne genau zu wissen wie die Attributwerte der Eingaben genau geartet sind. Hier erkennt dann wieder der Algorithmus die Muster in den Werten. In Kapitel 5 werden die Eingabedaten z.B. in vier Verhaltensklassen eingeteilt, unabhängig von der Anwendung, die hinter den Verbindungen steckt.

Zur abschließenden Überprüfung kann wieder eine genaue Payload-Analyse der Ergebnisse benützt werden. So kann man z.B. die tatsächliche Anwendung einer Verbindung ermitteln und mit den Zuordnungen des neuen Verfahrens vergleichen. Dabei spricht man von True-Positives (TP) wenn eine Verbindung richtig klassifiziert worden ist und von False-Positives (FP) bei einer falschen Klassifikation.

2.4 Anwendungsgebiete

Übliche Anwendungsgebiete von maschinellem Lernen sind z.B. das Data-Mining, was manchmal auch synonym zu maschinellem Lernen verwendet wird. Data-Mining bezeichnet das Gewinnen von Wissen und Daten sowie dessen Darstellung und Anwendung. Die verwendeten Methoden kommen meist aus der Statistik oder der KI und sollen auch auf sehr große Datenmengen mit vertretbarem Aufwand anwendbar sein. Ein Beispiel dafür ist das Kundenempfehlungssystem bei Amazon, bei dem Ähnlichkeiten zwischen Produkten ermittelt werden und dem Kunden solche, die ähnlich zu von ihm gekauften sind, vorgeschlagen werden. Allgemein fällt benutzerspezifische Werbung in diesen Bereich. Auch bei Spam-Filtern für e-mails werden Techniken des maschinellen Lernens verwendet [7]. Ein weiteres Anwendungsgebiet ist Pattern Recognition, wie z.B. das automatische Erkennen von Gesichtern auf Bildern oder bei der Spracherkennung.

2.5 Verkehrsklassifizierung

Die Möglichkeit den Verkehr in einem Netzwerk analysieren und in verschiedene Kategorien, z.B. nach Anwendung oder Verhalten, einteilen zu können ist ein wichtiger Teil vieler Netzwerk-Management Aufgaben, wie die Priorisierung von bestimmten Verbindungen, Traffic-Shaping/Policing, diagnostische Überwachung etc. So könnte ein Netzwerk-administrator z.B. Verkehr von p2p-Anwendungen im Netz drosseln um genügend Leistung für wichtige Business-Anwendungen eines Unternehmens sicher zu stellen. Ähnlich dazu helfen obige Möglichkeiten auch bei netzwerktechnischen Problemstellungen wie Workload-Charakterisierung, Planung der benötigten Kapazität usw.

Im Folgenden werden einige Forschungsarbeiten vorgestellt, die versuchen mit Hilfe von Methoden des maschinellen Lernens Verfahren zu entwickeln, die zum Lösen der obigen Problemstellungen beitragen können.

3. CLUSTERING

In einer Arbeit von der University of Calgary in Kanada [3] wird versucht den Netzwerkverkehr zu verschiedenen Anwendungen zuzuordnen. Dabei werden Clustering-Techniken (Unsupervised Learning) und Statistiken aus der Transportschicht verwendet. Untersucht wurden hier zwei Algorithmen, nämlich k-Means und DBSCAN. Diese werden mit den Ergebnissen, des schon in einer anderen Arbeit untersuchten AutoClass-Algorithmus verglichen. Das Clustering beruht hier darauf, dass unterschiedliche Anwendungen unterschiedliches Verhalten im Netzwerk zeigen. So besitzt z.B. eine Dateiübertragung mit FTP über lange Verbindungszeiten und große durchschnittliche Paketgrößen, während ein Instant-Messaging Programm nur gelegentlich kleine Nachrichten abschickt.

Das Verfahren ist in zwei Schritte aufgeteilt. Zuerst einer Offline-Trainingsphase und anschließend einer Klassifikationsphase in der online oder offline der Netzwerkverkehr klassifiziert wird. Für die Trainingsphase werden zwei Traces von unterschiedlichen Universitätsnetzen verwendet (von Calgary und Auckland). Aus diesen werden die verschiedenen Verbindungen extrahiert und deren Attribute auf Transportlevelschicht untersucht. Dabei wurden hier die Gesamtzahl der Pakete, die mittlere Paketgröße, die mittlere Nutzdatengröße ohne Header, die Anzahl der übermittelten Bytes, jeweils in beide Richtungen und zusammen und die mittlere Inter-Arrival-Time der Pakete betrachtet.

3.1 k-Means

Eine Möglichkeit für das Clustering der Eingabedaten ist es sie in einem euklidischen Raum mit n Dimensionen zu betrachten. Wobei n der Anzahl der betrachteten Attribute entspricht, z.B. Paketgrößen, Verbindungsdauer, usw. Jedes Objekt aus der Eingabe wird somit durch einen Vektor in diesem Raum dargestellt.

Der k-Means-Algorithmus findet k Cluster, die durch ihre Mittelpunkte im Raum festgelegt werden. Zuerst werden die Mittelpunkte $m_1,...,m_k$ zufällig oder manuell initialisiert. Dann werden wiederholt die folgenden beiden Schritte ausgeführt:

- Zuordnung aller Daten zum nähesten Clustermittelpunkt
- Neuberechnung der Clustermittelpunkte

Beim ersten Schritt kann als Abstandsmaß die euklidische Distanz d_e der Vektoren verwendet werden, die für zwei Punkte $x = (x_1,...,x_n)$ und $y = (y_1,...,y_n)$ folgendermaßen definiert ist:

$$d_e(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Die neuen Clustermittelpunkte m ergeben sich aus den Objekten $x_1,...,x_l$, die zum jeweiligen Schritt dem Cluster zugeordneten sind, zu:

$$m = \frac{1}{l} \sum_{i=1}^{l} x_i$$

[7]

Das Verfahren wird solange fortgesetzt, bis sich keine Veränderung der Mittelpunkte mehr einstellt. Dabei wird die Intra-Cluster-Varianz V minimiert:

$$V = \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - m_i\|^2$$

Wobei k die Anzahl der Cluster C_i, i=1...k und m_i der Mittelpunkt des jeweiligen Clusters ist. Der Algorithmus konvergiert immer, allerdings kann es sein, dass er dies nur gegen ein lokales Minimum tut [8].

Die folgenden Bilder veranschaulichen das Verfahren für einen zweidimensionalen Raum und k=3 Clustern:

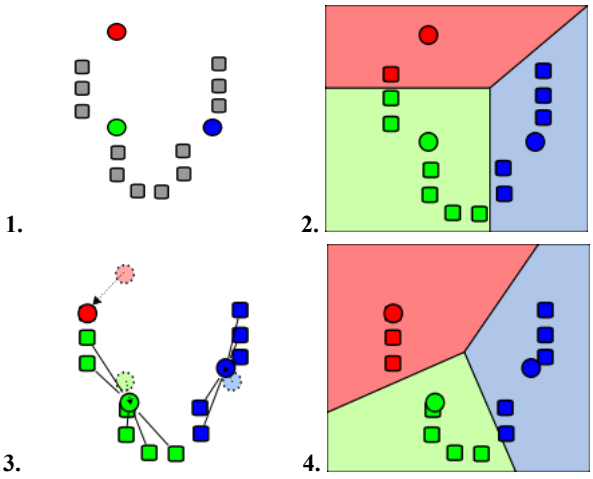


Abbildung 1. Visualisierung des k-Means-Algorithmus [9].

Bild1: Die Runden Punkte sind die zufällig gewählten Mittelpunkte der Cluster.
Bild2: Die Datenobjekte werden dem Cluster mit dem nähesten Mittelpunkt zugeordnet.
Bild3: Die Mittelpunkte werden neu berechnet.
Bild4: Neue Einteilung der Cluster.

3.2 DBSCAN

DBSCAN betrachtet Cluster als Gebiete mit Objekten die dicht beieinander liegen, die untereinander getrennt von Gebieten geringerer Dichte sind. Im Gegensatz zu k-Means können dadurch die Cluster beliebige Formen annehmen. Bei DBSCAN sind zwei Parameter von Bedeutung: eine gewisse epsilon-Umgebung und eine Mindestanzahl an Objekten. Liegen in der Umgebung eines Objekts mindestens die festgelegte Anzahl an anderen Punkten, wird zu diesem Objekt ein Cluster erstellt. Diesem Cluster werden anschließend alle weiteren Objekte zugewiesen die innerhalb der epsilon-Umgebung der enthaltenen Objekte liegen usw. Die epsilon-Umgebung gibt hierbei die maximale Distanz der Objekte voneinander für eine Zuweisung an. Objekte die am Ende übrig bleiben und keinem Cluster angehören werden als Rauschen gewertet – im Gegensatz zum klassischen k-Means oder AutoClass, bei denen jedes Objekt zugewiesen wird. Als Abstandsmaß wird hier wieder die euklidische Distanz gewählt.

AutoClass arbeitet mit einem Wahrscheinlichkeitsmodell für die Cluster. Der Algorithmus ermittelt von selbst die optimale Anzahl an Clustern und die Objekte können mit unterschiedlichen Wahrscheinlichkeiten mehreren Clustern zugeordnet sein, wobei hier ein Objekt einfach dem wahrscheinlichsten Cluster zugeordnet wird. Jedes Cluster besitzt eine Wahrscheinlichkeitsverteilung, deren Parameter mit einem Expectation Maximization-Algorithmus ermittelt werden (vgl. Kapitel 4.2).

3.3 Resultate

Die Testdaten werden mit Hilfe von Payload-Analyse-Tools untersucht, um die tatsächlichen Anwendungen der Verbindungen zu ermitteln. Wurden mehrere Anwendungen in einem Cluster zusammengefasst, werden bei der Klassifizierung alle Verbindungen zur in diesem Cluster dominierenden Anwendung zugeordnet. Die allgemeine Genauigkeit wird wie folgt definiert:

$$overall\ accuracy = \frac{\sum true\ positives\ for\ all\ clusters}{total\ number\ of\ connections}$$

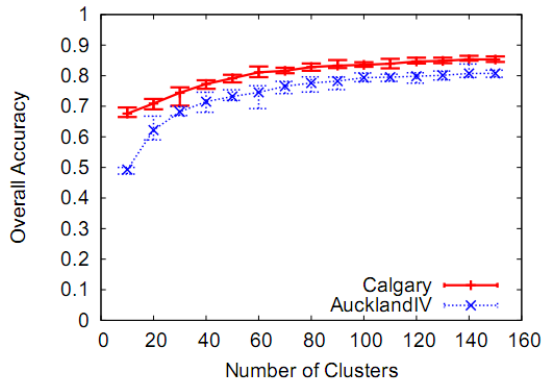


Abbildung 2. Overall Accuracy bei k-Means [3].

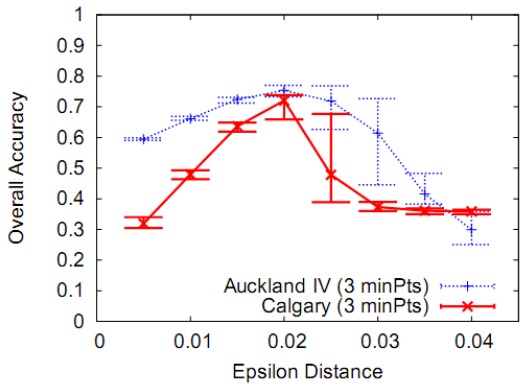


Abbildung 3. Overall Accuracy bei DBSCAN [3].

Die Diagramme der Abbildungen 2 und 3 zeigen die ermittelten Ergebnisse mit den verschiedenen Algorithmen. Bei k-Means sieht man die Ergebnisse für verschiedene Anzahlen an Cluster. Die Anzahl muss dem Algorithmus hierbei vorgegeben werden. Man kann hier erwarten das jedem Anwendungstyp mindesten ein Cluster zugewiesen wird. Bei Anwendungstypen mit unterschiedlichem Verhalten wie HTTP (browsing, streaming, …) werden mehrere Cluster entstehen, da jedes Verhalten sich in den untersuchten Attributen unterscheidet.

Auch DBSCAN wurde mit unterschiedlichen Parametern getestet. Hier müssen, wie erwähnt, die Größe der epsilon-Umgebung und die Mindestanzahl minPts an Objekten in einer solchen angegeben werden. Der gewählte Wert von drei für die Mindestanzahl minPts führt zu vielen kleinen Clustern, die ja auch schon bei k-Means zu guten Ergebnissen führten. Für zu große epsilon-Umgebungen verschmelzen die Cluster zu wenigen großen und die Genauigkeit sinkt rapide. Hier sei nochmal die Besonderheit des DBSCAN erwähnt Verbindungen auch als Rauschen klassifizieren zu können. Dadurch wird die allgemeine Genauigkeit zwar gesenkt, denn solche Verbindungen werden als falsch klassifiziert gewertet, jedoch ist das Verhältnis von TP zu FP innerhalb der Cluster bei DBSCAN dadurch am höchsten, so dass es weniger falsch klassifizierte Verbindungen im Verhältnis zu den richtigen gibt. Somit entstehen Cluster mit höherer Präzision was je nach Ziel der Klassifizierung von Vorteil sein kann.

AutoClass erweist sich als der beste Algorithmus in Sachen allgemeine Genauigkeit.

Um später beim Klassifizieren Rechenaufwand zu sparen, was ein wichtiger Faktor bei bei online Klassifikation ist, können kleine Cluster, die nur eine unbedeutende Anzahl an Verbindungen im Vergleich zur Gesamtzahl enthalten, verworfen werden. So verliert man nur wenig Genauigkeit, spart aber Rechenzeit bei der Zuweisung zu den Clustern.

Ein weiterer wichtiger Unterschied zwischen den drei Algorithmen ist, dass das Clustering bei k-Means und DBSCAN in wenigen Minuten, bei AutoClass hingegen erst nach 4,5 Stunden beendet war.

4. EXPECTATION MAXIMIZATION

Das hier vorgestellte Verfahren wurde an der Université Pierre et Marie Curie in Frankreich erarbeitet (siehe [1], [2]). Hierbei soll eine Verbindung „on the fly“ schon anhand der ersten paar verschickten Pakete zu einer Anwendung zugeordnet werden. Das Verfahren arbeitet dabei erneut in zwei Phasen und verwendet Clustering (Unsupervised Learning) mit k-Means, Gaussian Mixture Models (GMM) und Hidden Markov Models (HMM). Wobei die letzten beiden stochastische Modelle sind, die mit Expectation Maximization (EM) Verfahren arbeiten.

Folgende Abbildung zeigt einen Überblick des hier vorgestellten Verfahrens:

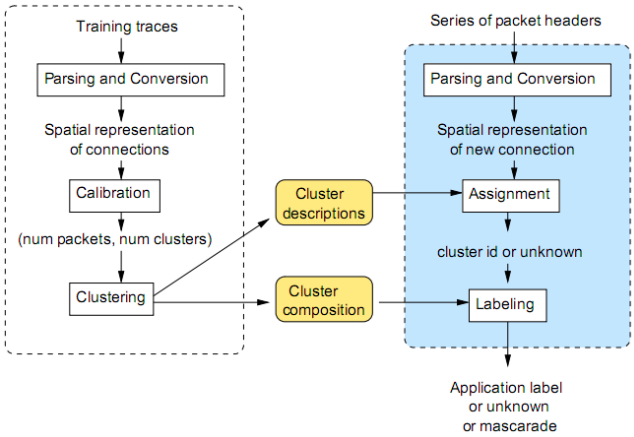


Abbildung 4. Überblick über das Verfahren [1].

4.1 Offline Training Phase

Als Repräsentation einer TCP-Verbindung werden hier zu jeder Verbindung nur die Größe der ersten paar Pakete und deren Richtung benötigt. Somit kann schon bevor eine Verbindung abgeschlossen ist und ohne großen Aufwand eine Zuordnung erfolgen. Im Schritt „Parsing and Conversion“ werden aus dem verwendeten Trace die benötigten Daten extrahiert.

Im Schritt „Calibration“ wird die optimale Anzahl der verwendeten Cluster und die Anzahl der zu untersuchenden Pakete ermittelt. Die optimale Anzahl der Cluster wird wieder ermittelt indem man die Ergebnisse des Clustering-Algorithmus mit der perfekten Zuordnung vergleicht (Payload-Analyse). Hier wurden 40 Cluster als guter Wert ermittelt. Die ersten fünf Pakete nach dem Three-Way-Handshake erweisen sich schon als aussagekräftig, da hier die Verbindungsverhandlung auf Anwendungsebene stattfindet.

4.2 Gaussian Mixture Models

Bei GMMs handelt es sich um einen Expectation Maximization-Algorithmus. EM-Algorithmen liefern keine harte Zuordnung der Daten zu den Clustern, sondern es wird für jedes Objekt die Wahrscheinlichkeit der Zugehörigkeit zu jedem der Cluster ermittelt. Dabei wird davon ausgegangen, dass die Art der Wahrscheinlichkeitsverteilung der Daten bekannt ist. Oft wird hier die Normalverteilung angenommen. Dabei wird meist eine Mischverteilung, d.h. eine gewichtete Summe von so vielen Verteilungen wie Cluster vorhanden sind, verwendet. Ein EM-Algorithmus bestimmt nun die Parameter (Mittelwert und Standardabweichung) der Verteilung für jeden Cluster.

Ähnlich wie bei k-Means werden wiederholt die beiden folgenden Schritte ausgeführt:

- **Expectation**: Für jedes Objekt x wird berechnet, mit welcher Wahrscheinlichkeit $P(C_j|x_i)$ es zu jedem Cluster C_j gehört.
- **Maximization**: Unter Verwendung der neu berechneten Wahrscheinlichkeiten werden die Parameter der Verteilung mit Maximimum-Likelihood-Methoden neu berechnet.

Erneut wird dies so lange fortgesetzt bis sich keine Änderung der Parameter mehr einstellt, was immer eintreten wird, jedoch unter Umständen nur bei einem lokalen Optimum [7, 8].

4.3 Hidden Markov Models

Das HMM ist ein stochastisches Modell, dem eine Markovkette zugrunde liegt. Die Zustände der Markovkette sind dabei „unsichtbar". Bei jedem Zeitschritt wird aber vom aktuellen Zustand ein sichtbares Symbol erzeugt. Beim Training von HMM werden Expectation Maximization-Algorithmen verwendet, die die Parameter, also Übergangswahrscheinlichkeiten zwischen den Zuständen und die jeweiligen Beobachtungswahrscheinlichkeiten für die Symbole, des HMM ermitteln.

Man kann dann bei gegebener Symbolfolge das HMM ermitteln, welches am wahrscheinlichsten die Folge erzeugt hat. Im hier vorliegenden Fall bestehen die Symbolfolgen aus den Paketgrößen der Verbindungen und jeder Cluster wird durch ein HMM repräsentiert, welches über so viele Zustände wie die Anzahl der betrachteten Pakete verfügt.

Schlussendlich werden die Cluster noch mit den darin enthaltenen Anwendungen gekennzeichnet.

4.4 Online Classification Phase

Beim „Assignment" werden nun „on the fly" Paket-Header aufgezeichnet und sobald drei Pakete einer Verbindung vorhanden sind, kann diese schon einem Cluster zugewiesen werden. Die Verfahren dazu ergeben sich aus dem verwendeten Clustering-Algorithmus. Für k-Means kann die Verbindung dem am nähesten liegenden Cluster zugeordnet werden, für GMM und HMM werden Maximum Likelihood-Methoden verwendet.

Beim „Labeling" wird die Verbindung schließlich mit der für sie ermittelten Anwendung gekennzeichnet. Hierbei wird einfach die im Cluster vorherrschende Anwendung verwendet, die in der Trainingsphase ermittelt wurde. Dabei werden leider alle Verbindungen die eigentlich zu einer anderen Anwendung in diesem Cluster gehören falsch klassifiziert. Zusätzlich wurde außerdem noch die Verwendung der Portnummer untersucht, um eine Verbindung möglicherweise doch einer anderen Anwendung im Cluster zuzuweisen. Allerdings nur wenn dies sinnvoll ist, also z.B. wenn es sich um einen der Well-Known-Ports handelt. Eine weitere Einsatzmöglichkeit für die Betrachtung der Portnummer ist das Erkennen von Täuschungsversuchen. Sollte eine Verbindung einen Well-Known-Port verwenden und einem Cluster zugeordnet werden, der die eigentlich zum Port gehörende Anwendung nicht enthält, dann kann dies ein Hinweis auf eine Anwendung sein, die versucht als sicher eingestufte Ports zu verwenden, um Firewalls zu umgehen. Solche Verbindungen könnten speziell gekennzeichnet werden und bei häufigem Auftreten könnte ein Netzwerk-Administrator dem Verdacht nachgehen.

Verwendet man Algorithmen, die Wahrscheinlichkeiten für die Zuordnung der Verbindungen verwenden, so kann zusätzlich ein Grenzwert für diese angegeben werden. Sollte eine Verbindung für alle Cluster unter dieser Grenzwahrscheinlichkeit liegen, wird ihre Anwendung als unbekannt eingetragen. Steigende Anzahlen von unbekannten Einträgen könnten dann auf eine neue Anwendung im Netzwerk hinweisen. Dann wäre ein erneutes Training zur Erkennung der neuen Anwendung angebracht.

4.5 Resultate

Tabelle 2 zeigt, nach den verschiedenen Anwendungen aufgeschlüsselt, die Ergebnisse von GMM und HMM beim Klassifizieren eines Test-Trace. Die Prozentangaben beziehen sich jeweils auf die Anzahl der Verbindungen. Die Tabelle zeigt die Ergebnisse, wenn beim Labeling die im Cluster vorherrschende Anwendung verwendet wird. Außerdem wurden keine Grenzwerte für die Wahrscheinlichkeit verwendet – es wird also nichts als unbekannt eingestuft. Für k-Means existiert im Paper [1] keine Aufschlüsselung nach Anwendungen, es lieferte aber auch eine Gesamt-Genauigkeit von über 90%. Der 0,0% Wert von HMM bei POP3 entsteht dadurch, dass POP3 und NNTP im selben Cluster sind und NNTP dort vorherrschend ist. Tabelle 1 zeigt die Erkennung von unbekannten Anwendungen, die in den Trainingsdaten nicht vorhanden waren, durch GMM bei einem Grenzwert von 99% - d.h. eine Verbindung wird als unbekannt eingestuft, wenn für keines der Cluster die Zugehörigkeits-wahrscheinlichkeit über 99% liegt.

Tabelle 1. Erkennung unbekannter Anwendungen [1].

	GMM mit Grenzwert 99%	
Anwendung	FP	Unbekannt
Bittorent	33,2%	66,8%
Gnutella	100,0%	0,0%
IRC	10,0%	90,0%
LDAP	8,8%	91,2%
MSN	38,5%	61,5%
Mysql	0,0%	100,0%

Tabelle 2. Labeling Accuracy von GMM und HMM [1].

	GMM		HMM	
Anw.	TP	FP	TP	FP
NNTP	81,2%	0,0%	99,8%	3,6%
POP3	96,5%	0,7%	0,0%	0,0%
SMTP	90,1%	0,1%	83,6%	0,2%
SSH	89,4%	0,0%	89,6%	0,0%
HTTPS	60,1%	0,0%	53,1%	0,0%
POP3S	93,4%	1,1%	96,1%	1,1%
HTTP	96,2%	1,3%	99,0%	1,7%
FTP	92,4%	0,4%	92,9%	0,3%
Edonkey	94,1%	0,3%	71,4%	0,0%
Kazaa	88,9%	2,6%	67,7%	0,7%
Overall	93,7%	X	92,3%	X

5. CLASSIFICATION

An der University of Adeleide in Australien wurde die Möglichkeit untersucht mit Supervised Learning-Methoden den Netzwerkverkehr in Verhaltensklassen einzuteilen, für die unterschiedliche Quality-of-Service (QoS) Anforderungen gelten sollen [5]. Verschiedene Anwendungstypen besitzen unterschiedliche solche Anforderungen an die Netzwerkverbindung, wie z.B. hoher Datendurchsatz für Datentransfer und kurze Verzögerungszeiten für interaktive Programme. Um diesen Anforderungen gerecht zu werden, sind Netzwerkbetreiber, besonders von großen Unternehmen, deshalb an der Möglichkeit interessiert ihren Datenverkehr verschiedenen QoS-Klassen zuordnen zu können. So könnten Anwendungen, die

für das Unternehmen wichtig sind, priorisiert werden und allgemein das vorhandene Netz effektiver ausgenützt werden.

Die Festlegung des QoS für eine Verbindung könnte im Prinzip auch von den Endpunkten aus durch die jeweilige Anwendung erfolgen. Aus Vertrauensgründen und der Skalierbarkeit der nötigen Administration ist es aber von Vorteil diese im Netzwerk selbst durchzuführen.

In dem hier verwendeten Verfahren wird nicht mehr versucht jeder Verbindung konkret eine Anwendung zuzuordnen, sondern diese werden nach ihrem Verhalten unterschieden. Hierzu wurden vier Verhaltensklassen definiert:

- Interactive: Beinhaltet Anwendungen bei denen ein Benutzer in Echtzeit mit einem entfernten System interagiert. Darunter fallen z.B. Remote-Login-Sessions oder ein interaktives Web-Interface.
- Bulk Data Transfer: Für die Übertragung von großen Datenmengen ohne Echtzeit-Anforderungen.
- Streaming: Multimedia Anwendungen die in Echtzeit übertragen, z.B. Videos.
- Transactional: Beinhaltet Datenverkehr, der eine kleine Anzahl von Anfrage/Antwort-Paaren übermittelt, die jeweils zu einer Transaktion zusammengefasst werden können. Darunter fallen z.B. DNS oder Oracle Transaktionen.

Zum Training werden hier wieder Datensätze, die als Referenz für die einzelnen Klassen verwendet werden können benötigt. In diesem Fall wurden für jede Klasse ein/zwei gut bekannte Anwendungen verwendet, die möglichst ausschließlich die jeweiligen Verhaltensweisen zeigen und außerdem weit verbreitet sind, um einen repräsentativen Datensatz zu erhalten.

Gewählt wurden jeweils:

- Interactive: Telnet
- Bulk Data Transfer: FTP-Data, Kazaa
- Streaming: RealMedia Streaming
- Transactional: DNS, HTTPS

Da die Daten hier schon mit den Klassenzugehörigkeiten versehen sind, handelt es sich um Supervised Learning. Klassifiziert werden Tupel aus Server-IP und Server-Port. Zu jedem solchen Tupel wird eine Statistik über verschiedene Eigenschaften der Verbindungen zum Server gesammelt. Wurden genügend Werte in der Statistik gesammelt, kann die Klassifikation des Tupels beginnen. Anschließend können alle Pakete mit diesem Ziel-Port und Ziel-IP automatisch in die entsprechende Class-of-Service zugeordnet werden.

Zur Klassifizierung wurden zwei einfache, aber verbreitete Methoden verwendet: Nearest Neighbour (NN) und Linear Discriminant Analysis (LDA).

5.1 Nearest Neighbour

Bei NN wird ein neues Objekt einfach der Klasse des zu ihm nähesten (euklidischer Abstand) Objekts aus den Trainingsdaten zugeordnet. Diese Technik lässt sich noch zu k-NN erweitern, so dass die k-nähesten Objekte ermittelt werden und unter diesen die am stärksten vertreten Klasse gewählt wird.

5.2 Linear Discriminant Analysis

Bei LDA werden die Klassen wieder mit Wahrscheinlichkeitsverteilungen beschrieben. Dabei wird die Klasse zur Zuordnung gewählt, die bei gegebenem Eingabeobjekt am wahrscheinlichsten ist. LDA liefert die Funktion die diese Wahrscheinlichkeit berechnet, wobei die Varianz zwischen den Klassen im Verhältnis zu der Varianz innerhalb der Klassen maximiert wird, so dass diese sich minimal überlappen.

5.3 Resultate

Abbildungen 5 und 6 zeigen die Einteilung von zwei Testdatensätzen in die vier Cluster, wobei als Attribute zur Zuordnung die durchschnittliche Paketgröße und die durchschnittliche Dauer der TCP-Verbindungen gewählt wurden. Zur Erinnerung: ein Punkt entspricht hier einem Tupel aus Server-IP und -Port.

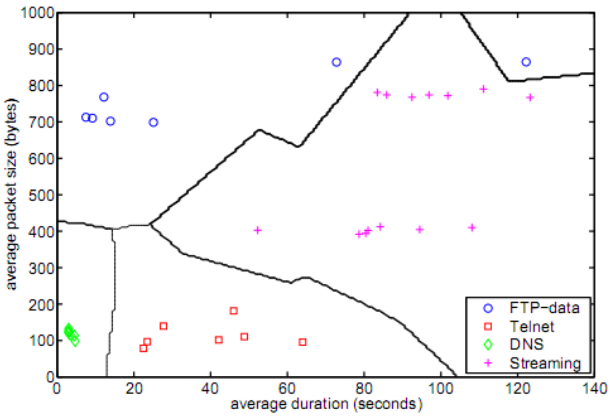


Abbildung 5. Nearest Neighbour [5].

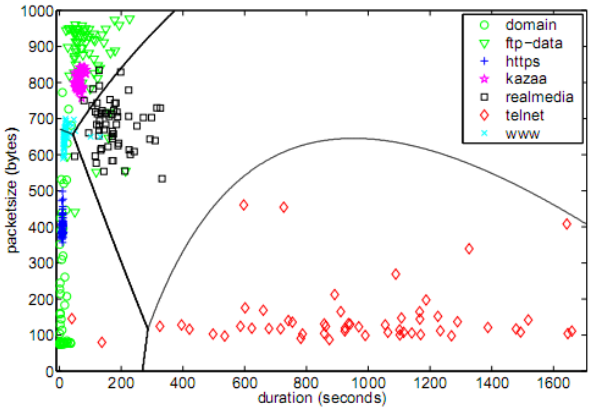


Abbildung 6. LDA [5].

Wie man in Abbildung 6 sehen kann, sind Streaming und Bulk Data Transfer am schwierigsten zu unterscheiden. Deswegen wurden weitere Attribute untersucht mit denen man diese genauer unterscheiden kann. Die Resultate davon kann man in Abbildung 7 erkennen.

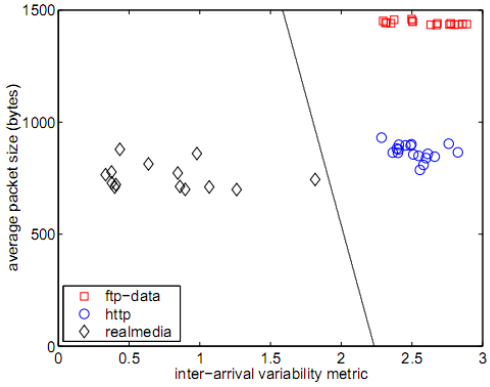


Abbildung 7. Streaming (RealMedia) vs. Bulk Data (FTP) [5].

Hier wurden nun die durchschnittliche Paketgröße und die Inter-Arrival-Variablity verwendet. Letztere ist das Verhältnis vom Mittel der Inter-Arrival-Times und der zugehörigen Standardabweichung, zusammen wiederum gemittelt über alle Verbindungen eines Tages. Diese eignet sich zur Unterscheidung, da die Zeitabstände zwischen den Paketen beim Streaming regelmäßiger sind als beim Bulk Data-Verkehr.

Zum Schluss wurden zur weiteren Überprüfung der Ergebnisse noch zwei Traces ein und desselben Netzes, aber zu unterschiedlichen Zeiten, verwendet. Der erste Trace wurde zum Training benützt, und der zweite, sechs Monate ältere, wurde anschließend klassifiziert. Während die Server-Tupel der Anwendungen zwar nicht mehr an der exakt gleichen Stelle waren, lagen sie immer noch in den ermittelten Grenzen der Cluster.

6. FAZIT

Verschiedene Forschungsgruppen beschäftigen sich mit dem Einsatz von maschinellem Lernen um alte Verkehrs-klassifizierungsmethoden zu verbessern. Der neue Ansatz liefert dabei erfolgversprechende Ergebnisse. Dabei ergeben sich sogar neue Möglichkeiten wie das automatische Erkennen von neuen Anwendungen und das Feststellen von verdächtigem Verhalten im Netz. Eine weitere etwas andere Anwendungsmöglichkeit, die hier nicht vorgestellt wurde ist z.B. der Versuch Kommunikation von Botnetzen zu erkennen [10].

Wie sich gezeigt hat gibt es unzählige Kombinationsmöglichkeiten von verschiedenen Techniken, Algorithmen und verwendeten Attributen des Netzwerkverkehrs. Es gibt deswegen noch Forschungsbedarf, welche davon sich als am zuverlässigsten erweisen und welche weiteren Anwendungs-möglichkeiten Verfahren dieser Art bieten.

Außerdem besteht auch weiterhin die Möglichkeit einer korrekten Erkennung durch die neuen Techniken zu entgehen, indem die untersuchten Kriterien durch ein Anwendungs-programm zufällig oder gezielt variiert werden. Wie sich die Verfahren unter solchen Bedingungen schlagen muss sich auch noch zeigen [1].

7. Literatur

[1] Laurent Bernaille, Renata Teixeira, and Kavé Salamatian. Early Application Identification. Université Pierre et Marie Curie - LIP6, CNRS, Paris, France. http://www-rp.lip6.fr/site_npa/site_rp/_publications/737-conextFinal.pdf.

[2] Laurent Bernaille, Renata Teixeira, Ismael Akodjenou, Augustin Soule, and Kave Salamatian. Traffic Classification On The Fly. Université Pierre et Marie Curie - Paris VI, Thomson Paris Lab. http://www-rp.lip6.fr/site_npa/site_rp/_publications/714-ccredito.pdf.

[3] Jeffrey Erman, Martin Arlitt, and Anirban Mahanti. Traffic Classification Using Clustering Algorithms. University of Calgary, 2500 University Drive NW, Calgary, AB, Canada. http://conferences.sigcomm.org/sigcomm/2006/papers/minenet-01.pdf.

[4] Anthony McGregor, Mark Hall, Perry Lorier, and James Brunskill. Flow Clustering Using Machine Learning Techniques. PAM(Passive&Active Measurement Workshop)2004. http://www.pam2004.org/papers/166.pdf.

[5] Matthew Roughanx, Subhabrata Sen, Oliver Spatscheck, and Nick Duffield. Class-of-Service Mapping for QoS: A Statistical Signature-based Approach to IP Traffic Classification. School of Mathematical Sciences, University of Adelaide,SA 5005, Australiax, AT&T Labs – Research, Florham Park, NJ 07932-0971, USA. http://www.imconf.net/imc-2004/papers/p135-roughan.pdf.

[6] Christian Osendorfer, and Martin Felder. Skript der Vorlesung "Machine Learning I"(2008). Technische Universität München. http://www6.in.tum.de/pub/Main/TeachingWs2008MachineLearning/Slides_01_Bayes.pdf.

[7] Wolfgang Ertel. Grundkurs Künstliche Intelligenz – Eine praxisorientierte Einführung.vieweg, 1. Auflage 2008, S.227-229(k-Means, Expectation Maximization) und S.184(Data Mining).

[8] Christopher M. Bishop. Pattern Recognition and Machine Learning. Springer Science + Business Media, LLC, 9. Auflage 2007, S.424 – 427(k-Means) und S. 430ff.(GMM).

[9] Englische Wikipedia. http://en.wikipedia.org/wiki/K-means.

[10] Carl Livadas, Bob Walsh, David Lapsley, Tim Strayer. Using Machine Learning Techniques to Identify Botnet Traffic. http://www.ir.bbn.com/documents/articles/lcn-wns-06.pdf.

Taming the torrent

Johannes Ranftl
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
j.ranftl@dora-showtechnik.de

***Kurzfassung* – Das Datenvolumen im Internet wächst täglich und P2P – Tauschbörsen tragen einen Großteil dazu bei. ISP's versuchen diesen P2P Traffic zu minimieren oder zu blocken. Das Bittorrent Protokoll ist eines der bekanntesten Filesharing P2P Protokolle. Im Folgenden wird ein Tool vorgestellt, das versucht den P2P Traffic im Netz des ISP's zu halten und so die Verbindungsqualität der Peers erhöht und die Kostenstruktur der ISP's verbessert.**

Schlüsselworte
Bittorrent, Downloadgeschwindigkeit, Ono, Content Distribution Network, Azureus, ISP

1. Einleitung

Durch das täglich wachsende Datenvolumen und die immer komplexeren Webanwendungen steigt der Bedarf nach Bandbreite immer stärker an. Nicht nur Unternehmen wollen heute mit schnellen Breitbandanschlüssen im Internet surfen, auch immer mehr Privatanwender wechseln auf schnelle DSL-Anschlüsse um das volle Leistungsangebot des Internets nutzen können.

Vor allem durch Web 2.0, Streaming-Medien und Plattformen wie z.B. Youtube.com oder Google-Video wird es für die Internetprovider immer schwieriger die nötigen Bandbreiten zuverlässig zur Verfügung zu stellen.

Durch das stetig steigende Angebot an P2P Software, wie Voice over IP und Tauschbörsen wird der Traffic für die Internet Service Provider (ISP) immer schwerer regulierbar. Vor allem die netzübergreifenden Anfragen, vom Netz des lokalen ISP ins Netz des ISP des anderen Peers von P2P Software treiben die Kosten für Traffic der ISP's in die Höhe. Eine Regulierung ist nur schwer möglich, da der Endknoten einer Peer to Peer Verbindung einen fixen Punkt darstellt und die Pakete somit nicht über Alternativrouten geroutet werden können.

Auch Bittorrent stellt ein Peer to Peer Netzwerk dar. Es wurde entwickelt um Dateien im Peer to Peer Betrieb zu tauschen. Eines der bekanntesten Bittorrent Programme ist Azureus.

In der folgenden Arbeit wird ein Plugin für Azureus vorgestellt, das versucht, bevorzugt auf Peers innerhalb des eigenen ISP-Netzes zuzugreifen. Somit verbessert sich nicht nur die Kostenstruktur des ISPs, was ja lediglich einen Vorteil für den Provider mit sich bringt, sondern auch die Qualität der Verbindung, worüber sich vor allem die Nutzer von Azureus freuen können, da höhere Downloadraten möglich sind und die Latenzzeit zum Peer um eine vielfaches reduziert werden kann.

Die Arbeit ist in 4 Teile gegliedert. Im zweiten Teil werden die Grundlagen vermittelt die nötig sind um einen Einblick in die Funktionsweise von Ono zu bekommen. Es wird speziell auf P2P Netze, Bittorrent und Positionsbestimmung im Netzwerk eingegangen.

Der dritte Teil beschreibt die Funktionsweise und den Nutzen von Ono. Vor allem die Netzwerkbestimmung mittels CDN-Anfragen steht hier im Mittelpunkt.

Der letzte Teil ist eine persönliche Beurteilung der Software die den Nutzen und die Funktionen des Tools kritisch in Frage stellt.

2. Grundlagen

Um auf die Funktionsweise von Ono eingehen zu können muss erst ein gewisses Grundverständnis für einige Techniken geschafft werden, die sich das Azureus-Plugin zu Nutze macht. Im nachfolgenden Abschnitt werden deshalb die nötigsten Grundlagen kurz erläutert.

2.1 P2P Netze

„In einem Peer-to-Peer-Netz sind alle Computer gleichberechtigt und können sowohl Dienste in Anspruch nehmen als auch Dienste zur Verfügung stellen. Die Computer können als Arbeitsstationen genutzt werden, aber auch Aufgaben im Netz übernehmen. Kernkomponente in einer Peer-to-Peer-Architektur ist das Overlay-Netzwerk, welches den Peers im Netzwerk die Funktionen Lookup und Suche zur Verfügung stellt.

Mit der Lookup-Operation können Peers im Netzwerk diejenigen Peers identifizieren, die für eine bestimmte Objektkennung (Object-ID) zuständig sind. In diesem Fall ist die Verantwortlichkeit für jedes einzelne Objekt mindestens einem Peer fest zugeteilt, man spricht von strukturierten Overlays. Mittels der Such-Operation können die Peers nach Objekten im Netzwerk suchen, die gewisse Kriterien erfüllen (z. B. Datei- oder Buddynamen-Übereinstimmung). In diesem Fall gibt es keine Zuordnungsstruktur für die Objekte im P2P-System, man spricht also von unstrukturierten Overlays.

Sobald die Peers in dem P2P-System identifiziert wurden, die die gesuchten Objekte halten, wird die Datei (in Dateitauschbörsen) direkt, d. h. von Peer zu Peer, übertragen. Es existieren verschiedene Verteilungsstrategien, [die entscheiden,] welche Teile der Datei von welchem Peer heruntergeladen werden sollen, z. B. BitTorrent.

In der Informationstechnik ist das Gegenteil zur Peer-to-Peer-Architektur die Client-Server-Architektur. Bei dieser bietet ein Server einen Dienst an und ein Client nutzt diesen Dienst. In Peer-to-Peer-Netzen ist diese Rollenverteilung aufgehoben. Jeder Host in einem Rechnernetz ist ein peer, denn er kann gleichzeitig Client und Server sein.“ [1]

2.2 Content Distribution Network

In der heutigen Zeit, in der durch Streaming-Medien und Web 2.0 das Datenvolumen im Internet kontinuierlich wächst wird es immer wichtiger, Inhalte im Internet möglichst effizient bereitzustellen und Lastspitzen auszugleichen. Das ist durch statische Client – Server Architekturen nicht mehr möglich. Um dieser Problematik vorzubeugen wurden CDN's entwickelt.

„Alle CDNs basieren auf dem gleichen Prinzip:

Die Inhalte eines Webanbieters werden auf verteilte Server an unterschiedlichen Orten repliziert, die sich üblicherweise innerhalb der Netzwerke von Internet Service Providern (ISP) und somit in unmittelbarer Nähe zu den Konsumenten befinden. Die hierbei verwendeten Knoten können dabei eigenständige Server sein, die von einem CDN-Betreiber ausschließlich für diesen Zweck installiert werden, Der Webanbieter beauftragt einen CDN-Betreiber (z. B. Akamai) mit der Verteilung seiner Inhalte und wird dadurch zu dessen Kunde (Customer). Oftmals erhalten die Kunden entsprechende (Software-)Werkzeuge, um die zu publizierenden Inhalte selbst auf den nächstgelegenen Server innerhalb des CDN hochzuladen – die Verteilung aus andere Server des CDN erfolgt automatisch. Über umfangreiche Tools hat der Anbieter Zugriff zu Aufrufstatistik, Netznutzung, usw. Beim Zugriff auf das Online-Angebot des Webanbieters erfolgt eine entsprechende ‚Umleitung' der Anfragen, so dass die an die Clients zurück gelieferten Inhalte überwiegend oder sogar vollständig von Servern des CDNs bedient werden. Bei der Auswahl des die Anfrage beantwortenden Servers wird immer der gerade günstigste Server ermittelt, wobei hier durch load balancing durchaus auch ein anderer als der nächstgelegene in Frage kommt, sofern dieser z. B. bereits stark ausgelastet ist." [2]

2.3 Positionsbestimmung im Netzwerk

„Um die Netzwerkdistanzen, bzw. Netzwerklatenzen, zwischen Knoten vorhersagen zu können, ohne dabei für jedes Knotenpaar die Latenz messen zu müssen, wurden Ansätze entwickelt, die meist auf der Berechnung synthetischer Koordinaten beruhen. Aus diesen Koordinaten kann dann mit einer Distanzfunktion die Latenzen zwischen zwei Knoten abgeleitet werden kann.

Im Folgenden werden grob einige gängige Verfahren skizziert:

IDMaps Einer der ersten Ansätze war IDMaps, ein infrastruktur-basierter Dienst zur Abschätzung der Latenzen zwischen Internetknoten. IDMaps basiert auf der Verwendung von einigen 1000 so genannter Tracer-Knoten. Alle Tracer-Knoten kennen ihre Latenzen untereinander. Außerdem gibt es eine eindeutige Zuordnung von Tracer-Knoten zu CIDR-Adresspräfixen. Die Latenzen zwischen diesen Präfixen und den so genannten Präfix-Tracer-Knoten sind bekannt. Die Berechnung der Latenz zwischen zwei Knoten k1 und k2 setzt sich nun aus den Latenzen zwischen den Präfix-Tracer-Knoten von k1 und k2 und der Latenz zwischen den beiden Tracer-Knoten zusammen [...]. Die Genauigkeit der Latenzschätzung hängt dabei von der Anzahl der Tracer-Knoten ab.

Global Network Positioning (GNP): Dieser Ansatz beruht auf einer kleinen Anzahl so genannter Landmark-Knoten, deren Koordinaten allen Knoten bekannt sein müssen. Andere Knoten müssen nun die Knoten-zu-Landmark-Latenzen via ICMP-Nachrichten messen. Basierend auf diesen Messungen werden nun die eigenen Koordinaten relativ zu denen der Landmark-Knoten errechnet [...].

Binning: Dieser Ansatz [...] basiert auf einer Klassifizierung der Knoten. Liegen zwei Knoten in der gleichen Klasse, sind sie im Bezug auf die Netzwerk-Latenz relativ nahe beieinander. Dieser Ansatz basiert auf der Annahme, dass es nicht notwendig ist, den jeweils optimalen Knoten zu finden, sondern dass auch eine grobe Abschätzungen durch Einteilung in verschiedene Distanzklassen ausreicht, um messbare Performanzsteigerungen zu erzielen.

Vivaldi: Hierbei handelt es sich um einen vollständig verteilten Algorithmus. Vivaldi [...] kommt ohne zentrale Infrastruktur aus und ist deshalb für viele P2P-Anwendungen besonders interessant. Jeder Knoten berechnet seine eigenen synthetischen Koordinaten in einem d-dimensionalen Koordinatensystem.

Dies erfolgt durch regelmäßigen Austausch der geschätzten Koordinaten und einer gemessenen Latenz zwischen zwei Knoten. Anhand dieser Daten kann nun jeder Knoten berechnen, inwiefern er die eigenen Koordinaten verschieben müsste, um die gemessene Latenz mit der geschätzten Latenz in Einklang zu bringen. Dabei wird angenommen, dass alle Knoten regelmäßig mit anderen Knoten kommunizieren und so die benötigten Daten austauschen können. Es ist ausreichend, wenn dabei jeder Knoten mit nur einem Bruchteil aller Knoten kommuniziert. Die Latenz lässt sich durch Berechnung der euklidischen Distanz der Koordinaten zweier Knoten schätzen. Hinsichtlich der Genauigkeit ist Vivaldi mit GNP vergleichbar. Die Qualität der Prognose hängt hier von der Anzahl der Kommunikationspartner und der Dimensionalität der Koordinaten ab[…]." [3]

2.4 Bittorrent

„BitTorrent […] ist ein kollaboratives Filesharing-Protokoll, das sich besonders für die schnelle Verteilung großer Datenmengen eignet. Technisch ist das Protokoll der OSI-Schicht 7, also der Anwendungsschicht, zuzuordnen und setzt auf das TCP/IP-Referenzmodell. Die Referenzimplementierung durch den Erfinder Bram Cohen erfolgte in der Programmiersprache Python. Mittlerweile steht eine Reihe alternativer Programme zur Verfügung, die das BitTorrent-Protokoll implementieren.

Im Vergleich zum herkömmlichen Herunterladen einer Datei mittels HTTP oder FTP werden bei der BitTorrent-Technik die (ansonsten ungenutzten) Upload-Kapazitäten der Downloader mitgenutzt, auch wenn sie die Datei erst unvollständig heruntergeladen haben. Dateien werden also nicht nur von einem Server verteilt, sondern auch vom Nutzer zum Nutzer (Peer-to-Peer). Das belastet den Server weniger und der Anbieter spart Kosten. Insgesamt ist die Downloadlast nicht geringer, sie wird lediglich auf die einzelnen Nutzer verlagert. Bei populären Dateien verhindert diese Technik das Zusammenbrechen des Netzes infolge des Überschreitens der Kapazitätsgrenzen des Anbieters.

Tracker

BitTorrent besteht aus zwei Teilen: Dem Server-Programm, genannt Tracker […] und dem Client, der auf dem PC des

Anwenders als Gegenstelle fungiert. Der Tracker verwaltet Informationen zu einer oder mehreren Dateien. Der herunterladende Client erfährt vom Tracker, wer sonst noch die Datei herunterlädt und verteilt. Sobald ein Client ein Segment […] der Datei erhalten und die Prüfsumme verifiziert hat, meldet er dies dem Tracker und kann dieses Dateistück nun schon an die anderen Clients weitergeben. Die Menge aller Clients, die am gleichen Torrent interessiert sind, nennt man Schwarm. Clients, die im Besitz des kompletten Inhalts des Torrents sind, und somit nichts von anderen Clients herunterladen, sondern lediglich Daten verteilen, nennt man Seeder […]. Clients, die nur im Besitz einiger Teile des Torrents sind, und sowohl Daten verteilen als auch beziehen, nennt man Peers […]. Clients, die nur herunterladen, ohne selber zu verteilen, nennt man Leecher […].

Um eine Datei herunterladen zu können, benötigt der Client eine Torrent-Datei […]. In dieser befindet sich die IP-Adresse (bzw. der Hostname) des Trackers sowie Dateiname, Größe und Prüfsummen der herunterzuladenden Datei. Eine Torrent-Datei kann auch Informationen über mehrere Dateien beinhalten. Torrent-Dateien sind wenige Kilobytes groß und werden üblicherweise auf der Website des Anbieters zum Herunterladen bereitgestellt. Löscht der Anbieter den Torrent aus dem Tracker

oder geht der Kontakt zum Tracker verloren, können die Clients keinen neuen Kontakt zu anderen Clients mehr aufbauen, der Austausch zwischen schon bekannten Clients kann aber fortgeführt werden. Um trotzdem die Kontaktaufnahme zu anderen Clients zu ermöglichen, wird in neueren Clients zusätzlich der trackerlose Betrieb verwendet.

Im Gegensatz zu anderen bekannten File-Sharing-Systemen werden nicht beliebige Dateien aus den Beständen der Teilnehmer ausgetauscht. Vielmehr verteilt jeder Schwarm nur die Dateien, welche der Autor der Torrent-Datei explizit zum Herunterladen vorgesehen hat. Auch der Betreiber des Trackers bestimmt selbst, welche Downloads von diesem verwaltet werden sollen. Die einzelnen Tracker stehen nicht in Verbindung zueinander, es existiert daher kein gemeinsames Netz. Jede Tracker-Datei erschafft somit ihr eigenes temporäres Netz aus beteiligten Clients. Anbieter können sich so von fremden, möglicherweise illegalen Inhalten leichter distanzieren. [4]

3. Taming the Torrent mit Ono

Nachdem die Grundlagen zum Verständnis von Ono geschaffen wurden kann nun genauer auf die Funktion und den Nutzen eingegangen werden.

Das Kapitel ist in 3 Teile untergliedert. Im ersten Teil wird genauer auf das Suchverfahren eingegangen, dass mit Hilfe von CDN anfragen versucht Peers innerhalb des ISP Netzes zu identifizieren und gezielt Verbindung zu diesen Peers aufzubauen.

Im zweiten Teil werden dann konkrete Messergebnisse vorgestellt und interpretiert um den Vorteil des Plugins zu visualisieren.

3.1 Positionsbestimmung mittels CDN

Um zu identifizieren, welche Peers innerhalb des gleichen ISP Netzwerkes liegen nutzt Ono die Basis Funktionalität der Content-Distribution-Netzwerke. Diese beinhaltet Anfragen via DNS-Redirection zu den günstigsten Servern weiterzuleiten. Content Distribution Netzwerke haben üblicherweise in den meisten ISP Netzwerken Server, die miteinander verbunden sind.

Ono schickt einfache Anfragen an Inhalte (Content) (siehe Abbildung 1) an das Content Distribution Netzwerk und dieses leitet die Anfrage automatisch auf die performantesten Server für den jeweiligen Peer weiter.

Abbr.	DNS Name	CDN	Description
AA	e100.g.akamaiedge.net	Akamei	Air Asia (Southeast Pacific)
CN	a1921.g.akamai.net	Akamei	CNN.com (US news site)
LM	a245.g.akamai.net	Akamei	LeMonde.com (French news site)
FN	a20.g.akamai.net	Limelight	Fox News (US news site)
Abbr.	wdig.vo.llnwd.net	Akamei	ABC Streaming Video (US television)
PW	a1756.g.akamai.net	Akamei	Popular Web Site

Abbildung 1

Anhand der Server, auf die der Peer geleitet wurde, erstellt Ono ein Verzeichnis, indem die gelieferten Server gelistet sind. Ono ermittelt anschließend die Latenzzeiten zu diesen Servern. Der dadurch entstehende „Netzwerkplan“ wird dann mit den Plänen der anderen Peers ausgetauscht und verglichen.

Zeigt sich beim Vergleich der Server, dass die Netzwerkpläne sehr ähnlich sind, so wird mit dem gefundenen Peer bevorzugt eine Verbindung aufgebaut.

Die Priorisierung der Peers findet auf Basis der Ratio Maps statt, die anhand der vom CDN zurück gelieferten Server erstellt wurde.

3.2 Vorteile durch Ono

Internet Provider versuchen auf immer neuen Wegen P2P Verkehr so gut es geht einzugrenzen und zu limitieren. Vor allem der Verkehr fremder ISPs der nur durch die eigenen Netze geleitet wird ist den Providern ein Dorn im Auge, da er einen Großteil des Traffics verursacht. Verständlicher Weise wird dieser Traffic von den Providern im Netz nicht mit höchster Priorität behandelt. Deshalb ist die Latenzzeit und die Rate der Paketverluste in P2P Verbindungen über mehrer Netze hinweg oft sehr hoch. Die Qualität der Verbindung ist somit eher als schlecht anzusehen.

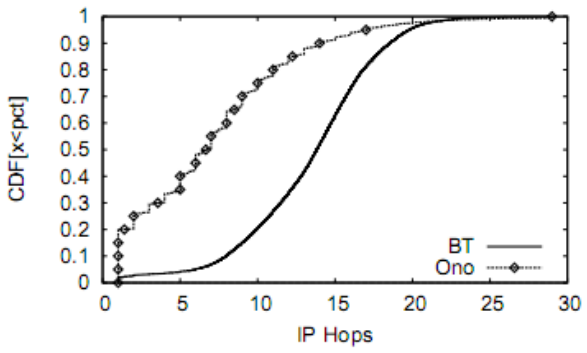


Abbildung 2 (Y- Achse Skala: 0 -100%)

Ono versucht durch den vergleich der erstellten Netzwerkkarten möglichst viel Traffic innerhalb des ISP internen Netzes abzuwickeln. Der Vorteil liegt auf der Hand. Für den Provider sind die Kosten geringer, da er nicht so viel Traffic in fremden

Netzen produziert. Zudem steigert die Qualität der Verbindung im eigenen Netz die Zufriedenheit seiner Kunden.

Für den Azureus Nutzer ist der Nutzen jedoch nicht weniger von Bedeutung. Die stabilere Verbindung führt zu weniger Paketkollisionen und somit zu weniger Verlusten. Die Latenzzeit zum gegenüberliegenden Peer ist um ein vielfaches geringer. Dadurch lassen sich bessere Downloadraten erzielen.

In Abbildung 2 ist zu sehen, dass durch das aktive Plugin die IP-Hops um einen großen Anteil reduziert werden können. Auf der

Um diese Zahl interpretieren zu können, ist es jedoch wichtig ein Verständnis für den Begriff IP-Hops zu bekommen.

Die Anzahl der IP Hops sagt aus, über wie viele verschiedne Server (IPS) ein Paket geroutet wird, bis es seinen Zielort erreicht. Eine niedrige Zahl an IP Hops ist somit ein Zeichen dafür, dass die Gegenstelle mit hoher Wahrscheinlichkeit im gleichen Netz liegt.

Im Zusammenhang mit den IP-Hops sollten auch die AS Hops erwähnt werden. Durch Ono bleibt mehr als 30% des Verkehrs innerhalb eines abgeschlossenen Systems, kurz AS genannt.

Dies zeigt deutlich, dass dadurch der Traffic stärker innerhalb des Netzes eines ISP gehalten werden kann als ohne das aktivierte Plugin.

Abbildung 3 zeigt wie viele AS (Autonome Systeme) durchlaufen werden bis der gegenüberliegende Peer erreicht wird. Es werden beide Methoden dargestellt. Sowohl das Standardt Verfahren über Bittorrent und Azureus, sowie das optimierte Verfahren mit Hilfe des Plugins Ono.

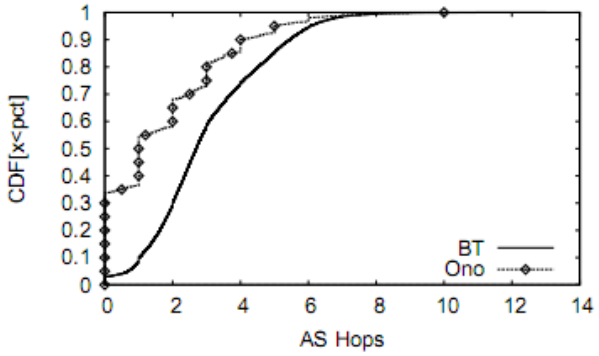


Abbildung 3 (Y- Achse Skala: 0 -100%)

Doch nicht nur die Anzahl der IP oder AS Hops sind ein Indikator für die Qualität der Verbindung.

Die Qualität hängt aber auch von der Latenzzeit ab. Die Latenzzeit, oder auch Verzögerungszeit genannt, spiegelt die Zeit wieder, die vergeht, bis eine Antwort von der Gegenstelle zurückkommt.

Die Latenzzeit wird von vielen Faktoren beeinflusst und ist ein wichtiger Wert in der Beurteilung von Netzwerkverbindungen.

In Abbildung 4 ist die Latenzzeit im Verhältnis, mit aktiviertem Ono-Plugin und ohne dargestellt. Um die Grafik richtig beurteilen zu können bedarf es jedoch einer kleinen Anmerkung. Die meisten Systeme runden Werte unter einer ms auf den Wert 1. Dadurch ergibt sich ein Wert von über 35% der Verbindungen mit einer Latenzzeit unter 1 ms. Dieser Wert ist für Internetverbindungen, und besonders für P2P-Verbindungen außergewöhnlich gut.

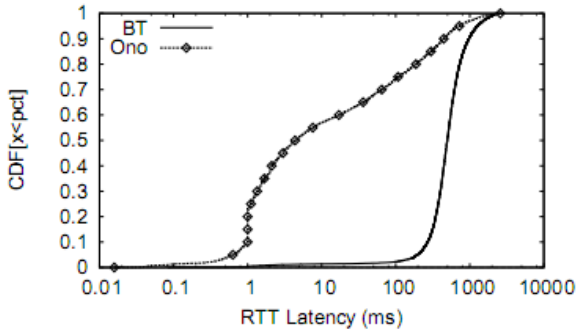


Abbildung 4 (Y- Achse Skala: 0 -100%)

Neben den IP und AS Hops, sowie der Latenzzeit bleibt ein weiteres Mittel, um die Effizienz und Qualität von TCP / IP Verbindungen beurteilen zu können.

Durch die Vielzahl an Paketen die durch das Internet geschickt werden und die große Flut an Paketdaten, die die Netzwerke und Netzwerkgeräte verarbeiten müssen, kommt es zu Paket-Kollisionen. Diese Kollisionen führen zum Verlust der kollidierten Pakete und können leider nicht verhindert werden, doch die Netzwerkprotokolle wissen damit umzugehen.

Durch Kontrollpakete wird erkannt ob es zu Kollisionen gekommen ist oder nicht. Wird also gerade eine Datei übertragen und das Kontrollpaket erkennt, dass es Kollisionen gegeben hat, so wird das verloren gegangene Paket erneut gesendet. Dies führt zu erhöhtem Traffic, ohne einen wirklichen Vorteil bei der Downloadgeschwindigkeit zu erreichen, da das verlorene Paket lediglich erneut gesendet wird.

Dieses Phänomen der Paketkollisionen wird durch die Loss rate, also die Verlustrate dargestellt. Je höher die Loss rate ist, desto mehr Pakete kollidieren irgendwo im Netz und gehen somit verloren. Diese müssen dann neu übertragen werden.

Abbildung 5 zeigt sehr deutlich, dass durch Ono die Verlustrate der Pakete um ein vielfaches geringer gehalten werden könnte.

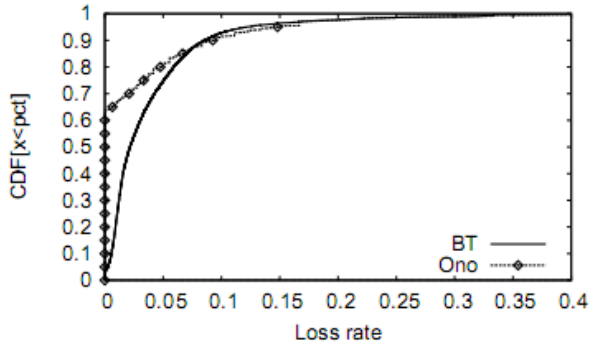


Abbildung 5 (Y- Achse Skala: 0 -100%)

Die Statistiken zeigen deutlich, dass durch Ono die Qualität der Verbindung um ein vielfaches gesteigert werden kann. Die Frage,

die sich jetzt stellt, ist, wie viele Systemressourcen benötigt werden, um dies zu erreichen.

3.3 Ressourcenverbrauch von Ono

Der Nutzen des Azureus Plugins wäre verschwindend gering, wenn durch die Auswahl der richtigen Peers schon ein Großteil der Bandbreite und Systemressourcen in Anspruch genommen werden würde. Daher stellt sich die Frage, wie aufwendig die gezielte Peeranalyse ist.
Der dieser Arbeit zugrunde liegende Artikel „Taming the Torrent“ stellte Messergebnisse vor, die zeigen, dass
sich die verfügbare Bandbreite durch die CDN Messverfahren nicht ausschlaggebend verändert.
Die Messungen werden schon von den CDN selbst übernommen, so dass sich für die Peers lediglich die DNS Anfragen zu den Content Distribution Networks ergeben. Diese Kommunikation ist jedoch mit dem Austausch einiger weniger Pakete schon erledigt. Somit entsteht durch die Anfragen an die CDNs eine so geringe Netzwerklast, dass diese vernachlässigt werden kann.
Zur Funktionalität von Ono gehört jedoch auch noch der Austausch und der Vergleich der erstellten Netzwerk Ratio Maps, also der angelegten Netzwerkkarten mit denen der gefundenen Peers.
Dieser Prozess ist aufwendiger, als die DNS Anfrage beim CDN, doch in Relation zu den heute verfügbaren Bandbreiten immer noch verschwindend gering. Messungen haben ergeben, dass Ono pro Tag mit einem Uploadvolumen von weniger als 50 KB auskommt. Das Downloadvolumen liegt bei unter 70 KB pro Tag und ist somit ebenfalls zu vernachlässigen.

Die Vorteile liegen also auf der Hand. Der Artikel zeigt, wie die Verbindung durch den Einsatz CDN basierter Messverfahren in einem P2P Netzwerk verbessert werden kann, ohne gewaltige Messverfahren und Bandbreiten in Anspruch nehmen zu müssen.
Lediglich durch verschicken kleiner DNS Anfragen wird die Auswahl der richtigen Peer ausgelöst und die Qualität der Verbindung somit deutlich gesteigert.

Ob sich diese Messergebnisse jedoch auch wirklich in Downloadraten zeigen und welche Probleme sich zusätzlich noch ergeben wird im folgenden Abschnitt erläutert.

4. Kritik

4.1 Nutzen

Nach einem ersten Blick auf die Messergebnisse ist man durchaus vom Nutzen des Plugins Ono überzeugt und sieht anhand der zahlreichen Grafiken relativ deutlich, wo der Vorteil gegenüber der normalen Peersuche liegt.

Betrachtet man die Thematik jedoch einmal genauer fängt man an sich über die Eine oder andere Problemstellung Gedanken zu machen. So stellen sich folgende Fragen:

Ist das Tool für jeden Inhalt den ich über Bittorrent downloaden möchte von Vorteil?

Findet Ono den Peer der mir am nahsten ist, oder den Peer der für mich am besten ist?

Gibt es einen Vorteil davon, den Traffic innerhalb meines ISP Netzes zu halten?

Beginnen wir mit der letzten Frage. Der Vorteil der sich für einen Endbenutzer ergibt hängt leider nicht nur von dem Faktor ab, ob meine Peergegenstelle innerhalb meines Netzes liegt und ist auch direkt mit der zweiten Frage verknüpft, denn das Ziel ist es als Azureus-Nutzer sicher nicht den nahesten Peer zu finden. Das Ziel ist es eher, den Peer zu finden, der die gewünschten Dateifragmente am schnellsten und unkompliziertesten zur Verfügung stellt. Liegt dieser Peer in einem anderen ISP Netz, so hat dies für den Azureus-Nutzer keine Nachteile, sofern der Download möglichst schnell abgeschlossen werden kann.

Hieraus gelangt man wiederum zur ersten Frage. Bringt das Tool für jeden Inhalt einen Vorteil? Nehmen wir mal an der Filesharing User möchte eine nicht sehr weit verbreitete Datei herunterladen. Ono verbindet ihn zu den verfügbaren Peers die innerhalb seines ISPs liegen.

Die beste Qualität der Verbindung, also mit geringer Latenzzeit und wenig IP-Hops ergibt sich nun bei einem Peer innerhalb des gleichen ISPs. Dieser ist nun jedoch gerade dabei eine Musiksammlung via Bittorrent aus dem Internet herunter zuladen. Seine maximale Upload-Bandbreite ist somit erschöpft, er stellt für den diesen Peer jedoch trotzdem eine Verbindung zur Verfügung.

Sicherlich ergibt sich jetzt ein Vorteil für den Provider, da dieser den Traffic innerhalb seines Netzes halten kann. Der Azureus Nutzer hätte jedoch mit einem Peer, der ein paar Knoten weiter entfernt in einem Nachbarnetz liegt, vielleicht mehr Bandbreite zur Verfügung gestellt bekommen. Dadurch wäre auch der Download schneller zu einem erfolgreichen Ende gekommen.

Somit zeigt sich in einigen Messergebnissen deutlich, wie gering sich der Vorteil von Ono in einigen Fällen zeigt.

In Abbildung 6 sieht man die Download-Geschwindigkeit mit und ohne den Einsatz von Ono in einem normalen ISP Netz. Der Unterschied ist gering.

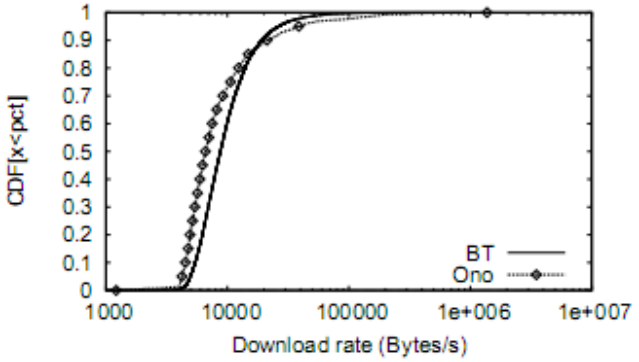


Abbildung 6 (Y- Achse Skala: 0 -100%)

Da die Anzahl der Nutzer stetig steigt und das Angebot immer größer wird ist es in 80%-90% der Fällen so, dass Azureus User ihren verfügbaren Down- und somit auch Upload fast immer zu 100% ausnutzen. Auch deshalb kann Ono hier keinen deutlichen Geschwindigkeitsvorteil erzielen.

Aus diesem Grund haben sich die Entwickler den Vorteil des Plugins mit Hilfe eines speziellen Tarifsystems eines fremden Providers gezeigt. Die Besonderheit des Tarifs liegt darin, dass für Verbindungen innerhalb des ISP Netzes mehr Bandbreite zur Verfügung gestellt wird als für Netzübergreifende Verbindungen. Abbildung 8 zeigt das Ergebnis dieser Messung.

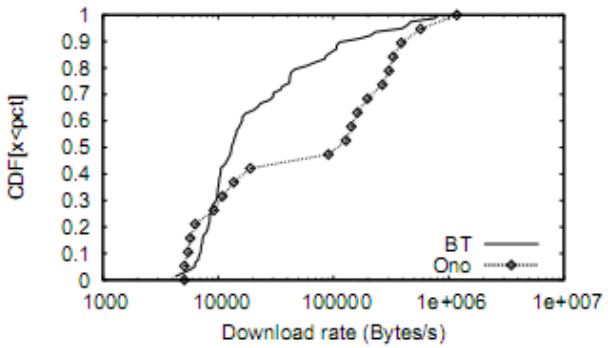


Abbildung 7 (Y- Achse Skala: 0 -100%)

Hier wird deutlich, dass die annähernde Vollauslastung der meisten Peers mit unter ein Grund für die schlechten Downloadvorteile von Ono ist.

Recherche in Online Foren hat ergeben, dass die Meinung über das Tool recht eindeutig ist. Die Aussagen könnten in folgendem Zitat zusammengefasst werden: „Ich konnte keinen Geschwindigkeitsvorteil erkennen!“

4.2 Resume

Abschließend könnte man sagen, die Idee des Plugins ist nicht neu. Es gibt viele verschiedene Plugins für Azureus oder andere Bittorrent Clients die von sich behaupten, die Downloadqualität und somit die Downloadraten zu steigern.

In der Praxis scheitern die meisten Tools jedoch nicht an den Messverfahren, sondern entweder an der zu geringen Verfügbarkeit der angefragten Dateien, oder an der allgemein überlasteten Bittorrent Peer to Peer Netzarchitektur. Denn die Grenzen setzen am Ende doch die Provider mit ihren Breitbandanschlüssen.

5. Literaturverzeichnis

[1]http://de.wikipedia.org/wiki/Peer-to-Peer

[2]http://www.net.t-labs.tu-berlin.de/teaching/ws0708/IR_seminar/ausarbeitungen/ thomas_guenther.pdf

[3]http://i30www.ira.uka.de/teaching/thesisdocuments/p2p/2004/tomanek_st_hybrid-overlay.pdf

[4]http://de.wikipedia.org/wiki/BitTorrent

[5] Alle Abbildungen wurden aus dieser Quelle entnommen: David R. Choffnes and Fabián E. Bustamante: Taming the Torrent, A Practical Approach to Reducing Cross-ISP Traffic in Peer-to-Peer Systems. USA

Chronicle Erkennungssysteme

Florian Bezold
Seminar Innovative Internet-Technologien und Mobilkommunikation, WS 2008/2009
Institut für Informatik, Lehrstuhl Netzarchitekturen und Netzdienste
Technische Universität München
florian.bezold@mytum.de

ABSTRACT
Das Ereigniserkennungssystem, oder Chronicle Recognition System (CRS), von dem dieser Artikel handelt, bekommt als Eingabe einen Kette von Ereignissen mit Zeitstempeln, versucht darauf auftretende vorgegebene Muster zu erkennen und generiert als Ausgabe Folge-Ereignisse oder führt Aktionen aus. Die Hauptaufgabe ist effizient komplexe Muster in Echtzeit zu erkennen.

Schlüsselwörter
Chronicle Recognition System (CRS), Situationserkennung, Ereigniserkennung.

1. Einführung – Situations Erkennung
Das Chronicle Recognition System baut grundlegend auf die Situationserkennung, die sogenannte Situation Recognition[1], auf.

Hierbei werden Situationen betrachtet, die nicht auf festen Zuständen, sondern auf Entwicklungen in sich verändernden Umgebungen basieren. Um ein solches System lauffähig zu halten, muss versucht werden, durch Quasi-Vorhersehung eine gute Interpretation davon geben können, was in einem dynamischen System passiert. Solche Aufgaben ergeben sich zum Beispiel in Umgebungs-, oder Prozessüberwachung in Netzwerken

Die Entwicklung dieser Situationserkennung orientierte sich an einer Multi-Sensor Maschine. Diese besaß mobile und feste Kameras, Laser zur Entfernungsbestimmung, Lichtschranken und Schallmesser. Sie wurde benutzt um eine geschlossene Umgebung zu überwachen.

Wenn nun also ein Sensor eine Veränderung feststellt, sei es ein Durchschreiten einer Lichtschranke oder eine Bewegungserkennung, so wird dieser Vorfall interpretiert und als Ereignis festgehalten. Eine Menge von Ereignissen, die innerhalb eines Zeitrahmens auftreten bezeichnet man als Situation. Die Feststellung einer solchen Situation, kann wiederum Ereignisse generieren oder Alarme auslösen, Nachrichten verschicken oder weitere Aktionen auslösen.

Der Entwickler eines solchen Systems stellt am Anfang Modelle von Situationen zur Verfügung, von möglichen Entwicklungen, die in der beobachteten Umgebung auftreten können. Jedes solches Situationsmodell besteht aus einer Menge von Ereignismustern und zeitlichen Beschränkungen zwischen ihnen. Wenn nun also Ereignisse beobachtet werden, die einem Muster entsprechen und auch die zeitlichen Schranken eingehalten werden, spricht man von dem Auftreten einer Instanz dieser Situation. Es kann auch angegeben werden, welche Folgeereignisse oder Aktionen beim Auftreten einer bestimmten Situation ausgelöst werden sollen. Ein solches Folgeereignis kann wiederum als Eingabe einer anderen Situation dienen, womit auch rekursive Ketten erzeugt werden können.

Die Hauptaufgabe eines solchen Systems ist es komplexe, temporale Muster zeitnah zu Erkennen, während diese auftreten.

2. Chronicle Erkennung
Die Chronicles im CRS beschreiben also solche Situationsmodelle, mit ihren Mustern von Ereignissen und deren zeitlichen Beschränkungen.

Das CRS bekommt als Eingabe eine Kette von Ereignissen, die mit Zeitstempeln versehen sind. Es erkennt Instanzen von auftretenden Chronicles, während diese sich entwickeln und produziert als Ausgabe Folgeereignisse, oder löst Aktionen aus.

Es ist grundsätzliche auf eine schnelle und effiziente Erkennung von komplexen Mustern ausgelegt.

2.1 Darstellung im CRS

2.1.1 Zeitdarstellung
Die Zeitdarstellung ist der Einfachheit halber linear gewählt, mit genügend kleinen Einheiten, um jedes Auftreten eines Ereignisses einem bestimmten Zeitpunkt zuordnen zu können.

Intervalle und Relationen können genauso benutzt werden, wie zeitliche Zusammenhänge der Zeitpunkt-Algebra (z.B. b*efore, simultaneous, after*)

2.1.2 Domänenattribute
Die Umgebung, die benutzt wird, ist durch Domänenattribute beschrieben. Ein solches Attribut besteht aus einem Paar P:v, wobei P der Attributname und v sein Wert ist.

2.1.3 Aussagen, Ereignisse
Eine Menge von Domänen Attributes P:v ist zeitlich bedingt durch Prädikate wie z.B. *event* und *hold.* Zu jedem möglichen Zeitpunkt t, hat jedes Domänenattribut nur einen einzigen Wert aus seiner Wertemenge (Abbildung 1).

Aussagen beschreiben das Bestehen des Wertes eines Attributs P über ein Zeitintervall [t1,t2], ohne exaktes Wissen darüber, wann dieser Wert erreicht wurde.

$$hold(P{:}v, (t1,t2))$$

Als Ereignismuster wird die Veränderung des Wertes eines Attributs bezeichnet. Das Ereignis selbst ist eine Instanz eines Musters, mit einem Zeitstempel, ohne bestimmte Dauer. Es wird ausgedrückt durch das Prädikat *event*.

$$event(P{:}(v_1,v_2), t)$$

Zusätzlich wird ein *forbidden event* (P, (t,t')) definiert. Das bedeutet, das Chronicle wird nicht erkannt, falls eine Veränderung des Wertes v innerhalb von t und t' auftritt.

noevent(P, (t,t'))

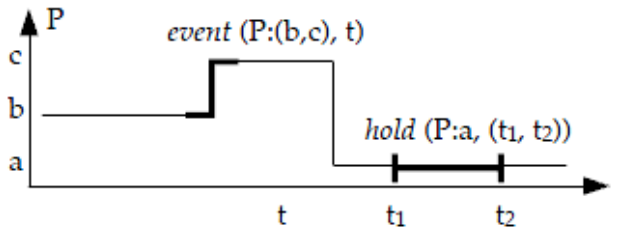


Abbildung 1: Die event *und* hold *Prädikate*

Quelle [3]

Diese drei Prädikate bilden die Basis der eigens entwickelten Sprache, zur Beschreibung der Chronicles, zu finden unter [8].

2.1.4 Ereignisverarbeitungsverzögerungen

Durch Sensorverarbeitung oder Datenübertragung wird ein Ereignis e immer erst mit einer gewissen Verzögerung empfangen. Diese Verzögerung wird begrenzt durch ein Intervall $\Delta(e)$, welches durch den Benutzer vorgegeben wird. Wenn d(e) der Zeitpunkt des Auftretens ist und r(e) der Zeitpunkt des Eingangs, so hat man immer $r(e) \in \Delta(e)+d(e)$

Diese Verzögerungen erlauben dem System Eingänge zu verarbeiten, die nicht in chronologischer Reihenfolge ankommen.

Zur Vereinfachung bezeichnet e den Namen des Ereignisses, sowie den Zeitpunkt des Auftretens d(e).

2.1.5 Chroniclemodell

Ein Chroniclemodell beschreibt einen Teil der Entwicklung der Umgebung. Es Besteht aus 4 Teilen:

- eine Menge von Ereignissen, die die relevanten Veränderungen der Umgebung für das jeweilige Chronicle repräsentieren,
- eine Menge von Aussagen, welche das Auftreten der Ereignisse des Chronicles beschreiben,
- eine Menge von zeitlichen Grenzen, und
- eine Menge von Aktionen, die durchgeführt werden, sollte das Chronicle erkannt werden.

Als Beispiel betrachten wir ein Chronicle, welches ein Problem mit einem Switch beschreibt (Abbildung 2).

```
chronicle SwitchProblem
{
    //- forthcoming chronicle events
    event (Transmission:(on,off),e1);
    event (Transmission:(off,on),e2);
    event (Component1:(?,ok),e3);
    event (Component2:(?,ok),e4);
    event (Component3:(?,ok),e5);

    //- assertions (context)
    hold (Traffic:normal, (e1,e6));

    //- temporal constraints
    e1 < e2 < e3 < e6;
    e2 < e4 < e6;
    e2 < e5 < e6;
    (e2 - e1) in [0, 180];
    (e6 - e2) in [60, 120];

    when recognised {
        report "Switch pb detection";
    }
}
```

Abbildung 2: Chronicle zur Erkennung eines Verbindungsproblems zu einem Switch

Quelle [3]

Der Block „forthcoming chronicle events“ beschreibt die Ereignisse, die der Reihe nach erwartet werden: Die Verbindung zu dem beobachteten Switch wird zum Zeitpunkt e1 unterbrochen. Nach automatischem Neustart des Gerätes, wird die Verbindung zum Zeitpunkt e2 wiederhergestellt. Danach senden die 3 Geräte, die an dem Switch angeschlossen sind, eine Statusnachricht zu den Zeitpunkten e3, e4 und e5. Das Beispiel verwendet eine einzige Aussage („assertions (context)“). Diese besagt, dass zwischen Anfang (e1) und Ende (e6) der überwachten Zeit, kein messbarer Anstieg des Datenverkehrs zu verzeichnen ist, da sonst die Reihenfolge der eingehenden Ereignisse beeinflusst werden könnte. Im Block „temporal constraints“ werden die zeitlichen Zusammenhänge definiert. Der letzte Teil des Chonicles beschreibt die Aktion, die ausgeführt wird bei vollständiger Erkennung. Fragezeichen werden an Stellen verwendet, an denen der genaue Wert nicht von Interesse ist.

2.2 Chronicle Entwicklung

Das Cronicle Recognition System muss im laufenden Betrieb alle eingehenden Daten auf Übereinstimmung mit Chronicle Modellen überprüfen. Wenn nur ein Teil der Ereignisse mit einer Untermenge von Chronicleereignissen übereinstimmen, spricht man von einer partiellen Instanz des Chroniclemodells. Bei einer kompletten Übereinstimmung (alle Zeitschranken und Aussagen werden eingehalten) spricht man davon, dass die Instanz erkannt wird.

Das System kann nur Ereignisse als Eingabe bekommen und verarbeiten. Da ein kontinuierlicher Strom von Ereignissen

angenommen wird, werden Aussagen durch Auftreten und Nicht-Auftreten dieser Ereignisse behandelt. Um z.B. die Aussage hold(P:v,(t1,t2) zu verarbeiten, überprüft das System, ob ein Ereignis *event*(P:(?,v),t) mit t<t1 stattgefunden hat, mit der Bedingung, dass keine Veränderung des Wertes von P vorgefallen ist in [t,t2[.

2.2.1 On Line Chronicle Management

Es gibt im Grunde 2 Arten, um eine Chronicleinstanz zu verändern. Entweder ein neues Ereignis tritt ein und wird in die Instanz integriert, oder Zeitlimit wird verletzt und des Instanz geschlossen.

Zur Verdeutlichung ein einfaches Szenario am eben genannten Beispiel. Die Verbindung zum Switch bricht zum Zeitpunkt 10' ab (e1), zum Zeitpunkt 12' (e2) ist der Neustart beendet. Wenn das System das Auftreten von Ereignis e1 feststellt, erkennt es eine mögliche Instanz des Chroniclemodells Es erwartet nun die Ereignisse e2, e3, e4, e5 in ihren jeweiligen Zeitfenstern (Abbildung 3 links). Ereignis e2 tritt innerhalb des Zeitfensters auf, also wird mit der Erkennung der Instanz normal fortgefahren. Wenn aber nun Ereignis e3, e4 oder e5 zum Zeitpunkt 14' nicht aufgetreten ist, wird das Zeitfenster verletzt und die Instanz geschlossen.

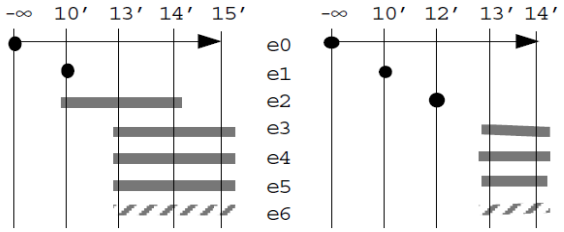


Abbildung 3: Verarbeitung von auftretenden Ereignisses innerhalb eines Chronicles

Quelle: [3]

2.2.2 Zeitlinien

Für jede Chronical Instanz werden 2 Zeitlinien geführt:

- Die *deadline* eines Chronicles ist der späteste Zeitpunkt für ein nicht auftretendes Ereignis, unter Rücksichtnahme auf evtl. anfallende Verzögerungen.
- Ähnlich berechnen wir die *non occurence line* einer Instanz, als letzter Zeitpunkt an dem man sicher sein kann, dass die letzte Aussage gilt.

Diese Zeitlinien werden benutzt, um festzustellen, ob etwas unternommen werden muss, obwohl kein Ereignis eingetreten ist. Wenn die jetzige Zeit die *deadline* überschreitet, „stirbt“ die Instanz, weil ein Folgeereignis nicht eingetreten ist. Bei Überschreitung der *non occurence line* gilt eine Aussage als veraltet (Falls es die letzte war, gilt das Chronicle als erkannt).

2.2.3 Duplikate von Chronicles

Um eine Erkennung bei jedem Auftreten zu gewährleisten, ist es von Nöten, Duplikate von Chronicles erzeugt zu erzeugen. Ohne Duplikate kann es passieren, dass Ereignisse verpasst werden, wenn sich mehrere Instanzen eines Chronicles zeitweise überlappen.

Betrachten man z.B. das folgende Modell:

e1 before e2 before e3 with (e3-e2) ≤ 3'

Als Eingabe dient die Ereigniskette:

e1 zu Zeit 0', e2 bei 3', e2 bei 8', e3 bei 11'

Das erste Auftreten von e2 wird also das Chronicle bei Zeit 6' „killen“, da kein Auftreten von Ereignis e3 innerhalb von 3' stattfindet. Das zweite Auftreten von e2 wird dadurch nicht erkannt. Diesem Problem kann entgegen gewirkt werden, indem zum Zeitpunkt 3'(erstes Auftreten von e2) eine zweite Instanz des Chronicles erzeugt wird. Eine Verdeutlichung der Erkennung von e2 zeigt Abbildung 4.

Die Hauptquelle der Komplexität des CRS ist die Anzahl der erzeugten Instanzen. Entsprechend muss versucht werden die Duplizierung zu begrenzen.

Die Erste Möglichkeit dies zu bewerkstelligen ist, jedem Chronicle eine maximale Lebensdauer zu geben. Trotz dieser Begrenzung kann immer noch eine große Anzahl von Duplikaten generiert werden.

Es kann beispielsweise vorkommen, dass Chronicles existieren, die keine zwei überlappenden Instanzen dulden. Oder der Benutzer könnte daran interessiert sein, nur eine Instanz gleichzeitig zu erkennen. In beiden Fällen wird, sobald eine Instanz erkannt wurde, alle offenen Instanzen des selben Chronicles entfernt.

2.2.4 Laufzeit

Jede elementare Operation in diesem System läuft in $O(m^2)$, wobei m die Anzahl der anstehenden Ereignisse in einer Chronicle Instanz ist. $m \leq n$ die ins gesamte Anzahl von Ereignissen im Chroniclemodell.

Für K Instanzen von Modellen mit jeweils n Ereignissen, verarbeitet der Algorithmus neue Ereignisse mit einer Komplexität von $O(Kn^2)$. K ist größer als M, die Anzahl von Chroniclemodellen. Wenn es schafft wird, dass K von der selben Ordnung wie M ist, bleibt die Gesamtkomplexität im Rahmen.

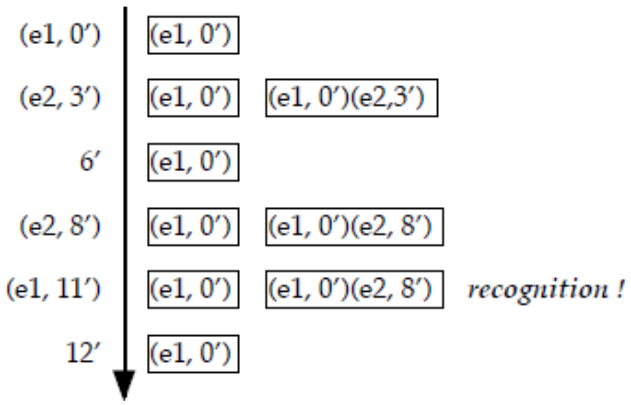


Abbildung 4: Duplikation während des Erkennungsprozesses

Quelle: [3]

2.2.5 *Frühe Anwendungen*

Zu Testzwecken wurde dieses System 1996 zur Überwachung des größten französischen Paketswitchnetzwerks eingesetzt [3]. Es wurden um die 100 verschiedene Chroniclemodelle benötigt um die selbe Arbeit zu verrichten wie das damals aktuell laufende System, welches an die 300 verschiedenen Regeln brauchte. Außerdem ist die Darstellung mit Chronicles für den Operator natürlicher, als Regeln für die Beschreibung von Abfolgen von Alarmen.

Ferner wurde es z.B. im AUSTRAL Projekt benutzt, um Alarmsequenzen zu analysieren, die von Unterstationen in einem mittleren französischen Stromverteilernetzwerk ausgesendet wurden [9].

3. Verbesserungen

Im Laufe der Zeit wurden einige Verbesserungen vorgenommen, um die Performance des Systems zu steigern. Ein paar von ihnen werden im Folgenden kurz dargestellt

3.1. Ereigniszähler

Die Hauptmotivation für diese Erweiterung entstammt der Alarmverarbeitung. Viele Alarme können von ein und demselben bestehenden Problem ausgelöst werden und das Zählen kann dabei hilfreich sein, die Gewichtung des Problems besser zu bestimmen. Außerdem konnten einige Fehler ausschließlich durch Zählen von Alarmen festgestellt werden.

So zum Beispiel im französischen Telekommunikationsnetzwerk. Die zwei folgenden Fehler sind bekannt:

- F1: ein technisches Zentrum (welches viele kleinere Unterzentren beherbergt) fährt herunter und startet neu. Es sendet der Überwachung ein „reboot“ Ereignis und jede der Unterstationen sendet ein „OK status“ Ereignis,
- F2: wenn eine Unterstation neu startet, sendet es ebenfalls ein „OK status“ Ereignis an die Überwachung.

Wenn nun viele dieser Fehler auf einmal auftreten, kann es passieren, dass einige der „reboot“ und „OK status“ Nachrichten von F1 verloren gehen und man den Ausfall von einem wichtigen Technischen Zentrum, nicht vom Ausfall von mehreren kleinen Unterstationen unterscheiden kann.

In der bisherigen Sprache sind solche Zähler schwer zu implementieren. Um diesem Problem entgegen zu wirken, wurde zusätzlich zu den bestehenden drei Prädikaten, *event*, *noevent* und *hold*, ein neues Prädikat *occurs* eingeführt:

$$occurs((n_1,n_2),a,(t1,t2), \text{ mit } 0 \leq n_1 \leq n_2$$

Dieses Prädikat beschreibt, zwischen Zeitpunkt n_1 und n_2 gibt es genau N Auftreten von Ereignissen, die dem Muster a entsprechen. Mit $n_1 \leq N \leq n_2$.

Unser neues Prädikat kann als Vereinheitlichung unser Chroniclesprache dienen, da man alle alten Prädikate durch das Neue ausdrücken kann. Das *noevent* Prädikat würde beispielsweise so aussehen:

$$noevent(a,(t1,t2)) \equiv occurs((0,0),a,(t1,t2))$$

Das Prädikat *event* sagt aus, dass ein Ereignis mindestens ein mal auftreten muss, es lautet also folgendermaßen:

$$event(a,t) \equiv occurs((1,+\infty),a,(t,t+1))$$

Diese neue Repräsentation von Zählern ist effizienter, weil es Chronicles in einer präziseren Art und Weise beschreibt. Wenn man nur mit den alten Prädikaten versucht Zähler zu implementieren, vergrößert sich die Anzahl von Chronicle Modellen, die das Erkennungssystem benutzt.

Dadurch, dass die Performance des Systems direkt mit der Anzahl der Chroniceinstanzen (oder Hypothesen) die erstellt werden zusammenhängt, vergleicht der folgende empirische Test wie viele davon mit dem alten und dem neuen System bei der Erkennung erstellt werden.

Es wurden zehn zufällige Chronicles mit Ereigniszählern, mit einer Obergrenze von weniger als 6 Ereignissen, erstellt. Außerdem wurden die entsprechenden Chronicles zusätzlich mit der alten Methode geschrieben, ohne *occurs* Prädikate (entsprach einer Anzahl von ca. 40 Chronicles). Beide Systeme bekamen als Eingabe log-Dateien, mit um die 1000 Ereignissen. Die folgende Tabelle zeigt die Anzahl der Chronicleinstanzen, die generiert wurden mit beiden Systemen.

Δ	Reco	old CRS	new CRS	ratio
4.418	165	35033	4712	7.43
7.025	105	32456	4722	6.87
9.64	71	26139	4411	5.93
12.1	58	23056	4348	5.3

Tabelle 1: Vergleich: altes und neues System

Quelle: [4]

Die Anzahl der Erkennungen (Reco) bleibt offensichtlich gleich, da sich an der Logik dahinter nichts verändert hat. Der Parameter Δ beschreibt die Verzögerung zwischen zwei aufeinander folgenden Ereignissen.

Bei einer Anwendung die z.B. das Zählen von Alarmen beinhaltet, kann mit dieser neuen Methode die Performance um ein vielfaches verbessert werden.

3.2. Temporäre Fokussierung

In manchen Fällen ist es so, dass verschiedene Ereignisse unterschiedlich oft auftreten. Angenommen, Ereignis f tritt sehr häufig auf und Folgeereignis e nur sehr selten. Wegen diesem Unterschied, könnte die Erkennung des Chronicles sehr schwierig sein. Für jedes Auftreten von f wird eine Instanz des Chronicles erstellt, die auf ein Eintreten von e wartet. Wenn also zwischen Zeitpunkt 0 und 1, tausendmal das Ereignis f auftritt, werden auch tausend Instanzen erstellt. Da aber e nur sehr selten auftritt, werden die meisten dieser Instanzen letztendlich wieder zerstört.

Um die Performance zu steigern und die Erstellung vieler unnötiger Instanzen zu verhindern, wird der Fokus auf das Ereignis e gelegt. Das geschieht folgendermaßen: wenn ein Ereignis f eintritt, wird dieses Ereignis in einem Kollektor gespeichert und eine neue Instanzen nur bei einem Auftreten von e erstellt. Wird ein Ereignis e festgestellt, prüft das System, welches vorhergehende Ereignis f im Kollektor das zu e gehörende ist, und erstellt die Instanz des Chronicles. Damit senkt man die effektive Anzahl der Instanzen die zur Erkennung von Mustern erstellt werden müssen.

Um nicht auf unwahrscheinliche Ereignisse beschränkt zu sein, wird ein *Level* für jeden Typ von Ereignissen eingeführt. Die temporäre Fokussierung sieht also so aus:

Fange mit einer Integration von Ereignis mit Level n+1 *nur an, wenn eine Instanz existiert, so dass alle Ereignisse mit Level zwischen* 1 *und* n *bereits integriert wurden.*

4. Anwendungen

Mit chroniclebasierenden Ansätzen wurde bisher in vielen Domänen experimentiert.

Das TIGER Projekt (Aguilar *et al*., 1994 [6]) entwickelte ein Softwaresystem, das die selbe durchgehende Überwachung einer Gasturbine gewährleisten sollte, die auch ein gut ausgebildeter Ingenieur bieten würde. Es bestand aus zwei Anwendungen. Einerseits wurde die 28MW General Electric Gasturbine des Fife Ethylene Kraftwerks in Schottland überwacht. Diese Turbine war ein vitaler Bestandteil des Kraftwerks, falls sie ausfiel hätte das komplette Kraftwerk abgeschaltet werden müssen. Andererseits kam das System in einigen Dassault Aviation Flugzeugen zum Einsatz und überwachte dort die „auxiliary power unit", eine kleinere Turbine. Das vom TIGER Projekt zur Verfügung gestellte System basierte teilweise auf der Chronicle Erkennung und konnte Ereignisse beschreiben, die für ein klassisches System zu komplex gewesen wären.

Weitere Anwendungen finden sich in der Medizin, z.B. zur Erkennung von Hepatitis Symptomen oder zur Intelligenten Patientenüberwachung. Das CALICOT System (Carrault et al., 1999 [7]) beschäftigt sich mit EKG Auswertung und der Erkennung von Herzrhythmusstörungen. Die On-line Analyse der EKG Daten wird von einem Chronicle Erkennungs-System durchgeführt. Es erkennt pathologische Situationen, indem es die symbolischen Beschreibungen der Signale mit zeitlichen Mustern vergleicht. In [10] werden erste Ansätze für die Anwendung von CRS für für Mobilitätsentscheidung in 3G&Beyond-Netzen beschrieben. Attribute für die Ereignisse sind in dem Fall beispielsweise Signalstärke und Paketverlustrate bei den aktuell verfügbaren Links (z.B. WLAN und UMTS).

5. Fazit

Chronicle Erkennung ist ein relativ neues System zur effizienten Erkennung von vorgegebenen Mustern innerhalb einer eingehenden Kette von Ereignissen. Im Gegensatz zu zur Zeit schon eingesetzten System, z.B. zur Analyse von Komponenten im Netzwerk, setzt es nicht auf Regeln zur Handhabung von Abfolgen von empfangenen Alarmen, sondern bietet dem Benutzer die Möglichkeit, in einer eigenen Programmiersprache sogenannte Chronicles zu definieren. Diese bestehen aus Ereignissen die im überwachten System vorkommen, verknüpft mit möglichen Nachfolgeereignissen die an feste Zeitfenster gebunden sind. Der Vorteil gegenüber konventionellen Systemen ist einerseits der hohe Grad an Formalität, mit dem die Modelle der Muster, welche überwachen werden sollen, beschrieben werden können. Anderseits die Effizienz der Erkennung, die einen Einsatz bei Echtzeit-Überwachung möglich macht.

6. Literatur

[1] C. Dousson, P. Gaborit, M. Ghallab. Situation recognition: representation and algorithms. *Thirteenth International Join Conference on Artificial Intelligence* (1993), Chambéry, pp. 166-172.

[2] C. Dousson, P. Le Magait. Improvement of chronicle-based monitoring using temporal focalization and hierarchization. *In proc. of the 17th International Workshop on Principles of Diagnosis (DX), pp. 257-261. Peñaranda de Duero, Burgos, Spain, June 2006.*

[3] C. Dousson. Alarm driven supervision for telecommunication networks: II- On-line chronicle recognition. *Annals of Telecommunications n° 9/10 (tome 51). September/October 1996, pp. 501-508*

[4] C. Dousson. Extending and unifying chronicle representation with event counters. *In proc. of the 15th ECAI, F. van Harmelen (ed.), IOS Press. pp. 257-261 Lyon, France, July 2002.*

[5] M.-O. Cordier, C. Dousson. Alarm driven monitoring based on chronicles. *In proc. of the 4th Sumposium on Fault Detection Supervision and Safety for Technical Processes (SafeProcess), pp. 286-291 Budapest, Hungary, June 2000*

[6] J. Aguilar et. al., TIGER: real-time situation assessment of dynamic systems. *Intelligent Systems Engineering pp. 103-124, 1994*

[7] G. Carrault et. al., A model-based approach for learning to identify cardiac arrhythmias. *In: AIMDM'99: Artificial Intelligence on Medicine and Medical Decision Making (W. Horn et al., Ed.). Vol. 1620 of LNAI. Springer Verlag Aalborg, Denmark.*

[8] C. Dousson. Chronicle's Language *http://crs.elibel.tm.fr/docs/language/index.html*

[9] P. Laborie, J.-P. Krivine, Automatic generation of chronicles and its application to alarm processing in power distribution systems. *8th international workshop of diagnosis (DX'97). Mont Saint-Michel, France, 1997*

[10] C. Dousson, *K.* Pentikousis, Chronicle Recognition for Mobility Management Triggers. *In: IEEE Symposium on Computers and Communications (ISCC'07), Portugal, 2007.*

ISBN 3-937201-04-1

ISSN 1868-2634 (print)
ISSN 1868-2642 (electronic)