Next Generation Networks

Dominik Seidel
Betreuer: Heiko Niedermayer
Seminar Future Internet SS2010
Lehrstuhl Netzarchitekturen und Netzdienste
Fakultät für Informatik, Technische Universität München
Email: seideldo@in.tum.de

KURZFASSUNG

Das Internet hat sich heutzutage zu einer etablierten Instanz entwickelt, im privaten sowie im geschäftlichen Umfeld. Jedoch ist aus Benutzersicht ein Wandel der Bedürfnisse entstanden: Weg von klassischen host-to-host Anwendungen hin zu einer auf Namen (des Contents) basierenden Kommunikation. Dem zu Grunde liegt die Tatsache, dass es den Benutzer nicht darum geht WIE oder von WO er den Content bezieht, sondern schlichtweg, dass er den gewünschten Content bekommt. Ein Lösungsansatz für dieses Problem sind die sogenannten Next Generation Networks(NGNs). Diese stellen das oben genannte "content-based" Networking in den Vordergrund, hauptsächlich durch das Verwenden gewisser Naming und Name Resolution Dienste. Für die Zukunft ist noch offen, ob sich diese neuartigen Netzwerke durchsetzen. Sie bieten aber auf jeden Fall gewisse Vorteile und Potentiale, auf die in der Arbeit eingegangen wird!

Schlüsselworte

CCN, DONA, Name Resolution, Naming, NGN

1. EINLEITUNG

Um das Internet als Technologie realisierbar zu machen, war das Domain Name System (DNS) eine der auschlaggebenden Entwicklungen. DNS wurde relativ früh , nämlich in den 1980ern entwickelt, als das Internet noch relativ jung, jedoch bereits schon an Leistungsgrenzen gestoßen war. DNS wurde daraufhin entwickelt, um eine strikte Übersetzung von Host Namen in IP Adressen zu gewährleisten. Ausserdem wurde DNS natürlich von Zeit zu Zeit angepasst und geupdated. Man kann es sich vereinfacht als einen hierarchischen und autonomem Verzeichnisdienst vorstellen, der den kompletten Namensraum verwaltet und in Zonen aufteilt. [4]
Das Problem bei dieser Enwicklung war, dass bereits andere Dienste verankert waren, wie zum Beispiel, dass TCP Sessions an IP Adressen gebunden waren, jedoch nicht an Namen. Somit waren die Möglichkeiten, DNS Namen wirkungsvoll einzusetzen schon im Vornherein durch die gegebene Architektur beschränkt. Man könnte auch sagen, dass sich das Internet auf dieser host-to-host Kommunikation basierend entwickelt hat. Zu dieser Zeit tat das den Ansprüchen auch genüge, nämlich der Kommunikation zwischen einer überschaubaren Anzahl von Hosts. Heute umfasst der Schwerpunkt der Internetnutzung jedoch die Datenübertragung sowie Zugang zu Diensten, wobei dem Benutzer lediglich der Content wichtig ist, und nicht wo dieser liegt bzw. erreicht wird. Letztendlich ist DNS nicht mehr ideal einsetzbar um die Bedürfnisse der Nutzer zu befriedigen und es Bedarf einem Neuansatz auf den hier eingegangen wird. Im Vordergrund der Arbeit steht die sogenannte Data-Oriented Network Architecture (DONA), welche einen Name Resolution Dienst "über" der IP Layer bereitstellen soll. Somit könnten die Mankos der bisherigen Datenübertragung beseitigt und der Zugrif auf Webservices erleichtert werden, sowie Verbesserungen im Bereich der Persistenz, Verfügbarkeit und Authenifikation erreicht werden. Ein Beispiel im Bereich Persistenz wäre hier, dass die sogenannten Broken Links die heutzutage gelegentlich auftreten, dann der Vergangenheit angehören sollten. [3]
In der Arbeit geht es hauptsächlich, um dies noch einmal zu betonen, um die DONA Architektur. Diese ist eine Variante eines NGNs, wobei es natürlich auch noch andere NGN Architekturen gibt, auf die hier nicht näher eingegangen wird. Ferner muss man wissen, dass die VoCCN Implementierung nicht im Zusammenhang mit DONA steht, und eigenständig zu betrachten ist. Deutlich wird dieser Neuansatz auch im Hinblick auf die Architektur, die Gegenstand des nächsten Kapitels ist.

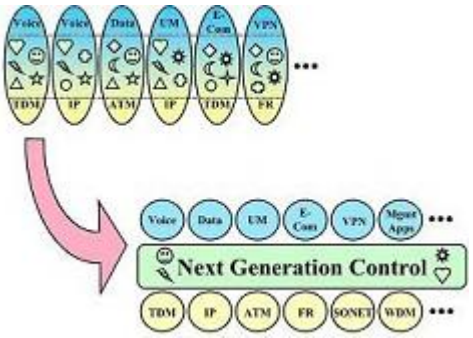

Abbildung 1: Next Generation Control [1]

2. ARCHITEKTUR

Bei der Architektur von Applikationen verwendet man meist je nach Vorliebe/Know-How etc. eine bestimmte zu Grunde liegende Technik, zum Beispiel wird eine Voice Applikation per IP-Protokoll realisiert. Bei einer zugeschnittenen Plattform auf der diese Applikation bzw. eine überschaubare Anzahl an Applikationen läuft, mag dieser Ansatz auch gut funktionieren, wird die Anzahl der Applikationen jedoch höher, kann es hier schnell zu Leistungsdefiziten kommen. Verantwortlich hierfür ist das Problem, dass bei vielen heterogenen Applikationen auch mehrere Verbindungen in ver-

schiedenen Protokollen aufgebaut werden müssen. Allgemein formuliert kommt es zu Einbußen, wenn zu viele unterschiedliche Systeme parallel arbeiten. Genau hier greift auch der Ansatz von Next Generation Networks, nämlich das Bereitstellen einer einheitlichen und flexiblen Steuereinheit (siehe Abbildung 1), die all diese verschiedenen Applikationen und Transportarten verbindet und verwaltet.[1]

2.1 VOIP Vergleich

Um aufzuzeigen, auf welche Punkte es bei der Architektur eines solchen Systems ankommt, kann man sich dies anhand einer einfachen Verbindung per VoiP zwischen zwei Benutzern klar machen. Abbildung 2 zeigt eine standardmäßige VoiP Verbindung:
Wenn Alice und Bob telefonieren wollen, wird zunächst eine Audio Verbindung per Session Initialiation Protocol (SIP) hergestellt. Durchgeführt wird dies typischerweise über einen Proxyserver, da die VoiP Endpunkte oft mobil sind oder dynamische IP Adressen verwenden. Akzeptiert Bob die Gesprächseinladung, so wird ein direkter "media path" zwischen beiden Endpunkten hergestellt. Hier zeigt sich das Dilemma, das in Verbindung mit der "alten" Architektur steht: Alice hat das simple Ziel direkt mit Bob zu reden. Das Netzwerk braucht um eine Verbindung herzustellen, jedoch den komplizierten Umweg aus Abbildung 2. Der Grund dafür ist, dass das Netwerk nicht direkt Bob's Telefon adressieren kann sondern lediglich seine IP Adresse!
Schaut man sich eine Realisierung der VoiP-Kommunikation

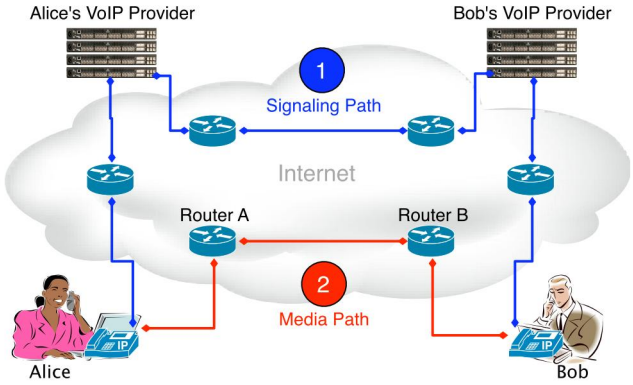

Abbildung 2: Klassische VoiP Kommunikation [2]

per Content-Oriented Network Architektur in Abbildung 3 an, sieht man schnell dessen Vorzüge: Hier fällt nämlich die komplette Übersetzung durch eine Middleware weg, sprich der oben genannte Umweg über Proxies und ein SIP. Man hat hier nurnoch eine einsträngige Verbindung, die Signaling und Media path in Einem ist. Natürlich bedarf dieser Verbindungstyp noch zusätzlichen Anforderungen, zum Beispiel muss der Angerufene einen "service contact point" bereitstellen, sozusagen einen Andockpunkt für das Telefonat.
Zusätzlich sind noch andere Voraussetzungen nötig, auf die in dieser Arbeit jedoch nicht näher eingegangen wird.
Wichtig ist aber, dass man schon anhand der zwei Grafiken

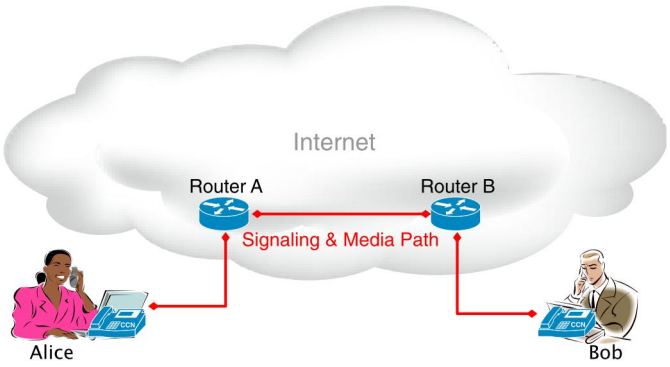

Abbildung 3: Voice-over-CCN Kommunikation[2]

sehen kann, dass hier eine Verringerung der Komplexität erzielt werden kann, abhängig von der verwendeten Architektur.[2]

2.2 Naming

Einen wichtigen Punkt der DONA Architektur ist die Zusammensetzung von Namen des Systems. Zentraler Punkt sind die sogenannten "principals", die Ersteller des Contents: Jeder principal lässt sich einem public-private Schlüsselpaar zuordnen, und jedes Datum, jeder Service oder Entity ist einem principal zugeordnet. Somit haben die Namen allgemein die Struktur P:L, wobei P der Hashwert des public keys (des principals) ist, und L die Benennung durch den principal. Als Resultat ergibt sich eine Namenstruktur die dafür sorgt, dass alle Namen einzigartig sind.
Ein weiterer Faktor ist das Besitzrecht der principals. Um genauer zu sein kann der principal festlegen, welchen Hosts er Zugriff auf die von ihm generierten Daten der Form P:L gewährt. Vorteil dabei ist auch, dass die Integrität der Daten damit auch garantiert/geprüft werden kann. Eine Verifizierung kann mithilfe der einzigartigen Namen und anhand der Kombination aus den eigentlich Daten, dem public key des principals und der Signatur erfolgen. Des weiteren kann durch einen Check, ob der public key denn wirklich zu P passt, Authentifikation garantiert werden. Persistenz wird dadurch erreicht, dass die Daten nicht an einen gewissen Ort gebunden sind, sondern überall bereitgestellt werden können.
Was man jedoch auch beachten muss, ist dass durch diese Technik auch Probleme entstehen, die es zu lösen gilt. Das zentrale Problem ist etwa, wie sich die Benutzer diese langen, komplex zusammengestellten Namen merken sollen. Hier schafft man sich Abhilfe, indem man externe Mechanismen verwendet, bei denen sich der Benutzer im Bezug auf die Sicherheit bzw. Vertrauenswürdigkeit seiner Daten sicher sein kann. Beispiel wäre eine Suchmaschine, die die komplexen Namen dann in menschenlesbare Namen umwandelt.[3]
Nun stellt sich die Frage, wie man sich sicher sein kann, dass diese Namen auch auf die gewünschten Ziele führen. Dies ist Aufgabe der Name Resolution, was im nächsten Abschnitt erläutert wird.

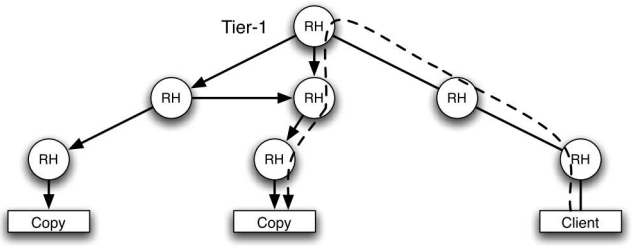

Abbildung 4: REGISTER = ganzen Pfeile und FIND = gestrichelten Pfeile[3]

2.3 Name Resolution

Nun gilt es zu erläutern wie DONA die Namen auf die gewünschten Hosts mappt. Ziel dieses Dienstes ist es eine hohe Verfügbarkeit der Daten zu erzielen, indem nahegelegene Kopien der Datei gefunden werden können, sowie die Fehleranfälligkeit minimal gehalten wird. DONA benutzt hierfür sogenannte Resolution handlers (RHs), die eine neue Art von Netzwerkentität darstellen. Jede Domain oder Verwaltungseinheit verfügt dann über eine spezifische RH. Denkbar ist auch das Untergliedern von sozialen Strukturen mit Hilfe der RHs. Zum Beispiel könnte an der TU München jeder Lehrstuhl einen eigenen RH haben, und dieser könnten dann für die Mitarbeiter am Lehrstuhl intern andere Rechte besitzen, im Gegensatz zu Lehrstuhl externen Personen. Realisiert wird diese Technik, indem jeder Client seinen lokalen RH kennt und diesem zugeordnet ist.

Desweiteren stellt DONA zwei Basisbefehle zur Verfügung, nämlich einmal FIND(P:L) und REGISTER(P:L), die in diesem Abschnitt erläutert werden. FIND hat den Zweck, solche Daten mit dem Namen P:L zu finden, und die Rhs sorgen dafür, dass eine nahegelegene Kopie davon gefunden wird. Im Gegensatz dazu sorgt REGISTER dafür, das Daten sozusagen für andere auffindbar sind, und somit durch FIND Anweisungen gefunden werden können.

Ein FIND(P:L) wir abgehandelt, wie in Abbildung 4 teils ersichtlich, indem zuerst geschaut wird ob unter den RHs eine den gesuchten Namen registriert hat. Ist dies der Fall, wird der FIND zum nächsten RH geleitet, so lange bis der Pfad bis zu dem RH mit der gewünschten Kopie erreicht wird. Im Bezug auf unser Beispiel wandert der FIND erst einmal bis zur Wurzel, und daraufhin bis zum Blatt mit der gewünschten Kopie. Zusätzlich initiieren die FIND Befehle auch den eigentlichen Austausch/Transport der Daten. Nur der Austausch der eigentlichen Datenpakete erfolgt per gewohnter IP Routing Technik. Das hat den Vorteil, dass an der bestehenden IP Infrastruktur keine Änderungen vorgenommen werden müssen.

Bei den REGISTER(P:L) Befehlen ist das Verhalten ein wenig anders: Wenn ein RH einen solchen erhält, wird dieser lediglich nach oben weitergereicht, wenn es eine nähere Kopie enthält (im Vergleich zur bestehenden "Struktur") oder wenn es sich um ein neues Objekt handelt. Ausserdem bedarf es einer Authentifikation, da garantiert werden muss, dass es sich bei P um eine vertrauenswürdige Quelle handelt. Der Client muss sich deshalb in der Regel mit P's private key anmelden.[3]

3. FUNKTIONEN

Die zu Grunde liegenden Architekturmerkmale wie die Name Resolution spiegeln sich natürlich auch in der Funktionalität von DONA wieder. Grundsätzlich lassen sich diese in vier Bereiche aufteilen, die in der Arbeit nun erläutert werden.

Erster ist die Selektion des Servers (bzw. mehrerer Server). Will ein Server einen Service mit dem Namen P:L registrieren, so benötigt er erst einmal die Authorisierung von einem principal P. Ist das erfüllt, dann wird der Service einfach beim lokalen RH angemeldet, und ist ab sofort dann auch für FIND Anweisungen zu finden. Natürlich ist es so, dass immer der näheste Server gewählt wird bei identischen Services. Durch diesen Mechanismus wäre es auch kein Problem zum Beispiel P2P Applikationen wie BitTorrent zu realisieren. Hier würde man einfach einzigartige Namen verwenden, mit denen der Principal die jeweilige P2P Instanz identifizieren kann. Probleme würden auch nicht auftreten, wenn wie bei BitTorrent üblich, die Dateien in einzelne, kleine Teile gesplittet werden. In diesem Fall würde man sich einfach mit einem Index Abhilfe verschaffen, der die einzelnen Teilnamen dem Dateinamen zuordnet! Der Client könnte somit auch sehr simpel nur einzelne Teildateien bekommen.

Zweite wichtige Funktionalität ist der "mobile" Charakter. Dies soll heißen, dass ein Host sich einfach an einer Location abmelden kann und beim Wiederverbinden die neue Location anmeldet. Somit würden FINDs dann auch direkt an diese neue Location geleitet werden, sobald die Registrierung erfolgt ist. Ähnlich läuft dies beim sogenannten Multihoming ab, sprich wenn man mehrere ISPs besitzt: Automatisch wird dann ein REGISTER an alle Provider übermittelt. Gleichzeitig können FINDs dann auch multiple Bezugsquellen benutzen.

Nächster Punkt ist die "Session Initialition". Eine zentrale Rolle spielen hierbei das sogenannte Session Initiation Protocol (SIP), welches eng mit den grundlegenden DONA Funktionen (FIND und REGISTER) verknüpft ist. Typischerweise erfolgt dann der Aufbau einer Session zwischen zwei Partnern wie folgt: Zuerst schickt ein SIP Agent eine SIP INVITE Nachricht an den Sessionpartner. Das SIP Proxy leitet diese Nachricht dann an die Location des Zielagents weiter und wenn dieser antwortet, startet die Session. Um die Location auch immer auf dem aktuellen Stand zu halten, registrieren SIP Agents kontinuierlich ihre Position. Im Zusammenhang mit FIND/REGISTER kann man sagen, dass eine SIP Invite Nachricht vergleichbar mit einem FIND ist, sowie eine SIP und DONA REGISTER Nachrichten ziemlich ähnlich sind. Anzumerken ist noch, dass wie schon einmal erwähnt, der eigentliche Datentransfer standardgemäß per IP Protokoll erfolgt.

Der letzte Punkt ist das "Multicast State Establishment". Diese Funktion beinhaltet Die Möglichkeit über Domaingrenzen hinaus anderen Gruppen ohne Problemen beizutreten. Zum besseren Verständnis nehme man an, es gebe einen Domainrouter mit lokalen Mitgliedern in einer Gruppe X. Nun muss aber gewährleistet sein, dass auch Gruppenmitgliedern die aus anderen Domains stammen, der Zugriff möglich ist bzw. mit diesen kommuniziert werden kann. Dies ist Sinn und Zweck dieser Funktion, die per "intradomain multicast protocol" solch ein Überschreiten der lokalen Domaingrenzen ermöglicht. Auch im Hinblick auf die Gruppennnamen erkennt man wieder die typische Struktur: Gruppen haben Namen der Form P:G, wobei P wieder der principal ist,

also der Gruppengründer und G der eigentliche Gruppenname. Damit ist ebenfalls wieder die Einzigartigkeit der Gruppennnamen garantiert.[3]
Dies waren die Grundfunktionalitäten, wobei es noch zahlreiche spezifischere Erweiterungen für DONA gibt bzw. in Arbeit sind. Auf diese wird im Rahmen dieser Arbeit jedoch nicht eingegangen, da dies eher Feinheiten sind die für ein Grundverständnis für NGN's/DONA nicht noetig sind. Wichtig ist aber der nächste Punkt, nämlich der Sicherheitsaspekt in einem solchen System.

4. SICHERHEIT

4.1 Beispiel VoCCN Sicherheit

Um ein Beispiel mit Praxisbezug zu haben, kommen wir zunächst noch einmal auf das VoCCN Beispiel (Abbildung 3) zurück. Hier wurde Media Path und Signaling Path seperat gesichert.
Bei Beiden benutzt man die Authentifizierung der User per Key-paar. Eine Verbindung kommt also nur zu Stande wenn die entsprechenden Keys verifiziert werden. Verantwortlich für die Verteilung dieser ist ebenfalls das CCN Netzwerk. Zusätzlich wurden im Media Path alle Gespräche per Secure Real-Time Transportation Protokoll geführt. Dieses sorgt grob gesagt für eine Verschlüsselung, Authentifizierung und Integrität des Nachrichtenaustauschs.
Die Sicherheit des Signaling Path wird garantiert, indem ein einfaches Verschlüsselungs- und Authentifizierungsschema implementiert ist. Genauer gesagt wird die SIP Invite Message verschlüsselt und benutzt einen zufällig generierten symmetrischen Schlüssel auf Seiten des Angerufenen. Der Anrufer würde diesen symmetrischen Key dann seinerseits per public key seines Gegenübers entschlüsseln.
Als Resultat hat man laut den Entwicklern dann eine deutlich sicherere Variante im Vergleich zur herkömmlichen VoiP Kommunikation. Diese ist nämlich oft komplett unverschlüsselt und ohne Authentifizierung![2]

4.2 Sicherheitsbelange

Der erste kritische Punkt sind denial-of-service Attacken die RHs, Server oder Clients "überfluten" könnten, was jedoch per IP-level Mechanismen in der Regel verhindert wird. Größeres Problem stellen schon Angriffe dar, die zum Ziel haben, die Ressourcen einer RH zu limitieren/einzuschränken. Hier müsste man sich dann mit den ISPs verständigen. Genauer gesagt müssten diese ein Limit an FINDs und REGISTERs pro Zeitperiode festsetzen.
Betrachtet man den Austausch der RHs untereinander, ist man schon zum Großteil dadurch abgesichert, dass die RHs untereinander ihre public keys austauschen und verifizieren. Dadurch kann garantiert werden, dass das gesendete Paket auch von der gewünschten RH stammt. Eine böshafte RH ist dadurch jedoch noch nicht abgewehrt. Diese könnte zum Beispiel erhaltene REGISTER Befehle schlicht und einfach verweigern, insbesondere das Weiterleiten an die nächsten RHs. Noch schlimmer wäre der Fall, wenn eine bösartige RH die REGISTERs anderer RHs abhört und dessen Daten sammelt. Um dies zu verhindern wird weitergeleiteten REGISTERs stehts der public key der nächsten RH angefügt. Ein Dilemma ware natürlich auch das Verweigern des Service einer RH an den oder die Clients. Dieses Problem kann aber behoben werden, indem der Client diese RH als fehlerhaft erkennt und automatisch eine andere Kopie der Daten ansteuert.
Im Allgemeinen kann man sagen, dass die RHs in der Regel von den Internet Service Providern (ISPs) verwaltet werden und Kunden dafür zahlen. Folglich wird es im Interesse aller liegen, wenn bei Problemen so schnell wie möglich eine Lösung gefunden wird. Vor allem die ISP werden es sich nicht leisten können, Sicherheitslücken zuzulassen.[3]

5. IMPLEMENTIERUNG

Im Bereich der Implementierung gilt anzumerken, dass die meisten Testsysteme nur im kleinen Umfang gearbeitet haben. Dies liegt daran, dass meist nur Prototypen implementiert wurden und logischerweise ein System im Umfang des eines ISPs umzusetzen, im Rahmen von Tests nicht möglich war. Man sieht jedoch gut, wie die einzelnen Module eines solchen Systems agieren.
Als erstes benötigt man ein Router Modul. Dieses handelt empfangene REGISTERs und FINDs ab und leitet diese bei Bedarf weiter. Es beinhaltet auch das Registration Table, welches eben FINDs gezielt an andere RHs weiterleiten kann. Aussderdem überwacht das Router Modul den Status der angeschlossenen Nachbarn. Mit diesen Nachbar-RHs werden auch REGISTER Daten ausgetauscht, sodass jede RH immer auf dem neusten Stand ist.
Ein weiteres Modul ist das Caching Modul. Dieses ermöglicht den Zugang zu dem lokalen Cache von Daten. Darauf können dann gewisse Applikationsmodule zugreifen, zum Beispiel eine P2P Applikation. Wird der Cache abgefragt erfolgt dies in zwei Schritten: Erst wird ermittelt ob das gesuchte Objekt im Cache ist. Ist dieser Vorgang erfolgreich, wird auf das Objekt regulär per Dateisystem zugegriffen.
Letztes Modul sind die Application Modules, die oben schon genannt wurden. Sehr üblich sind Dinge wie HTTP, SIP,P2P Applikationen oder RSS.
Allgemein hatte man bei dem implementierten DONA Prototyp an die 17.000 Zeilen Programmcode. Aber wie gesagt konnte man noch keine Aussagen über die Skalierbarkeit machen, da dafür das System noch zu wenig Umfang hatte.[3]

5.1 Performance

Dier Performance konnte man jedoch schon ganz gut bei der VoCCN Implementierung vergleichen. Zunächst war die Gesprächsqualität laut der Verfasser recht gut. Getestet wurde zwischen zwei schnellen Workstations mit 100 Mb/s oder im Wechsel mit 1Gb/s.
Die genauen Testergebnisse lassen sich ganz gut in Abbildung 5 ablesen. Hier wurde jeweils ein 10 minütiges Gespräch getestet, einmal per VoCNN Implementierung (gestrichelte Linie) und einmal über die "'normale"' Variante mittels UDP. Abweichend von den erwarteten Resultaten hat VoCCN deutlich weniger Pakete unter oder bei der erwarteten Intervallzeit. Im hohen Intervallbereich sind lediglich wenige Pakete transferiert worden. Alles in allem sind bei keiner der beiden Varianten Pakete verloren gegangen![2]

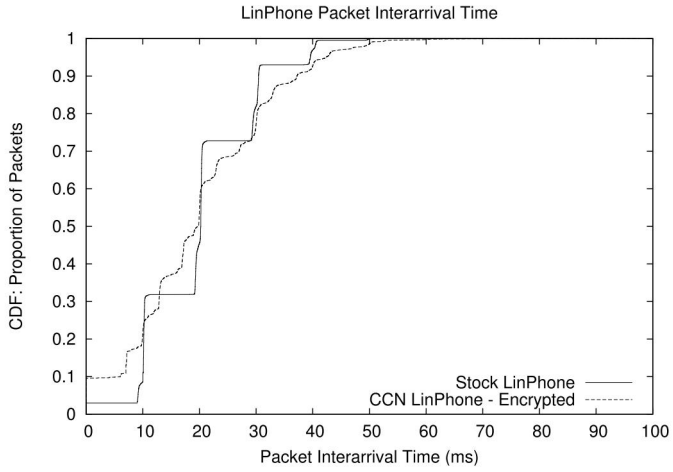

Abbildung 5: Verteilung der Paketintervalle (kumultativ) bei einem 10 min Anruf[2]

6. REALISIERBARKEIT

Da nun alle wesentlichen Fakten im Bezug auf DONA geklärt sind, stellt sich die Frage ob in der Realität ein solches System überhaupt umsetzbar ist oder ob vieleicht die Kosten ein solches zu betreiben den wirtschaftlichen Rahmen eines ISPs sprengen wuerden. Angemerkt sei, dass im Bereich einer "'durchschnittlichen"' RH keine grossen Leistungsengpässe zu erwarten sind. Grund dafür ist, dass eine RH ja wie gesagt baumartig arrangiert ist, sprich sie muss nur FINDs und REGISTERs von RHs auf gleicher Ebene oder darunter abhandeln. Lediglich die Wurzel im Baum, daher die Tier-1 RH, welche seitens des Providers verwaltet wird muss sogesehen das volle Spektrum der registrierten Datennamen wissen

Schätzt man zunächst den Platz der zum Speichern aller (öffentlichen) Webseitennamen nötig wäre, würden wir auf ca. 10 hoch 11 Seiten kommen. Geht man nun von 42 byte pro Eintrag aus, kommt man auf lediglich 4 TB benötigten Speicher. Da heutzutage die Preise für Speicher dieses Umfangs sehr erschwinglich sind, stellt dies kein Problem dar.

Geht man ferner von einer durchschnittlichen Lebensdauer von zwei Wochen pro REGISTER aus, muss eine Tier-1 RH ungefähr 83000 Registrierungsnachrichten pro Sekunde verarbeiten können. Umgerechnet entspricht dies 680 Mbps, was für ein RH eines ISPs kein Problem darstellen darf. Was jedoch weitaus aufwendiger ist als der blose Register-Befehl, sind die komplizierten kryptograpischen Prozesse die gleichzeitig anfallen. Hochgerechnet wäre für diese Arbeit die Leistung von ca. 40 3 GHz CPUs nötig. Trotz allem ist diese Anzahl doch durchaus im Rahmen und stellt für einen ISP kein Problem dar.

Auf Seiten der FINDs kann man davon ausgehen, dass diese ungefähr die selbe Last verursachen wie eine gewöhnliche HTTP Anfrage. Man kann annehmen, dass eine voll ausgelastete 1 Gbps verbindung etwa 20.000 FINDs pro Sekunde verursacht. Rechnet man bei einem FIND 150 byte Volumen, wären das nur 24Mbps die nötig sind, sprich 2,4 Prozent der Leitung. Nehme man nun des weiteren an, dass ein CPU um die 40.0000 FINDs pro Sekunde verarbeiten kann, so können 500 PCs mit jeweils 8 GB RAM summa summarum eine Last von 1 Tbps verkraften. Dies wäre schätzungweise auch ein sicherer Wert der noch Puffer zulässt. Wohlgemerkt wäre mit diesen 500 PCs die komplette Tier-1 eines ISPs abgedeckt, was durchaus im Rahmen einer wirtschaftlichen Tätigkeit liegt.[3]

7. ZUSAMMENFASSUNG

Betrachtet man nun diese NGNs im Hinblick auf die Zukunft, haben diese durchaus Potential Nachfolger der herkömmlichen Netwerke mit DNS etc. zu werden, da sie das was den Kunden und Benutzer am meisten interessiert, den Content, in den Mittelpunkt stellen. Dieser kann dann durchaus effizient und skalierbar bereitgestellt werden. Auch klassische Dienste wie telefonieren, e-mailen oder surfen kann per CCN effektiv bereitgestellt werden. Das Ergebnis muss nicht ungedingt besser sein als das der herkömmlichen IP-Struktur, entscheidender Faktor ist aber: Die Vereinfachung in der Architektur, Implementierung und Konfiguration ist der ausschlaggebende Wegweiser. Darüber hinaus spielt der Sicherheitsaspekt, im Hinblick auf Datenklau Skandale etc., zur heutigen Zeit eine immer wichtigere Rolle. Da dieser bekanntlich (siehe Abschnitt 3) wesentlich höher als bei den jetzigen Netzwerken ist, besteht auch hier hohes Potential. Vor allem werden keine Zwischenstationen mehr, wie z.B. Proxies benötigt.[2]

Allerdings ist ein wesentlicher Faktor, der wahrscheinlich für den Erfolg oder Misserfolg verantwortlich sein wird, noch fraglich: Nämlich ob diese Technologie die Aufmerksamkeit der Telekommunikationsbranche und deren Unternehmen auf sich ziehen kann. Denn ohne eine weitflächige NGN Umwelt wäre dessen Durchsetzung auf dem Markt wohl eher unwahrscheinlich. Zusätzlich werden erst durch eine breite Wahrnehmung/Entwicklung der Industrie die wirklichen Voraussetzungen für NGNs bekannt werden. Unternehmen sollten deswegen die NGN Technologie als Differenzierungsstrategie sehen um sich von der Konkurrenz abzuheben![1]

8. LITERATUR

[1] J. Crimi. Next generation network (ngn) services. *Telcordia Technologies*, 2009.

[2] V. Jacobson, D. Smetters, and N. Briggs. Acm rearch '09. In *VoCCN: Voice Over Content-Centric Networks*, Dezember 2009.

[3] T. Koponen, M. Chawla, and B.Chun. Sigcomm'07. In *A Data-Oriented (and Beyond) Network Architecture*, August 2007.

[4] P. Vixie. Dns complexity. *Queue*, 5(3):24–29, April 2007.