Promotionskolloquium

„Integration von isochronen Diensten, Sicherheitsmechanismen und Help-Desk-Lösungen in dreidimensionale E-Commerce Systeme unter Berücksichtigung der Benutzeranforderungen“

Dipl.-Ing. Kai-Oliver Detken
Universität Bremen, MZH 4180
Bremen, 29. November 2002

Inhalt

- Einordnung der Thematik und Eingangsthesen
- Lösungsansätze
- Messungen
- Eigene Lösungen
- Wissenschaftlicher Fortschritt
- Offene Diskussionsrunde

Einordnung in die Thematik und Eingangsthesen

Themen der Dissertation

- Die Promotion beschäftigte sich im Rahmen des EU-Forschungsprojektes INTELLECT (IST-10375) mit der Evaluierung und Integration von sicheren Echtzeitplattformen für das Internet
- Gerade eCommerce-Anwendungen leiden unter fehlender Akzeptanz von Security sowie QoS und besitzen Standardisierungslücken
- Viele benötigte Mechanismen sind ebenfalls noch nicht einsatzfähig bzw. verhalten sich in der Realität anders als in der Theorie
- Standardisierung der Grundlagen solcher Mechanismen und Dienste ist abhängig von prototypischen Messungen, Analysen und Implementierungen, die die Anforderungen aufzeigen
- Die Dissertation gibt einen Überblick der Entwicklung der letzten Jahre und nimmt durch Messungen eine Bewertung vor, die nach Feststellung der Engpässe in eigene Lösungsvorschläge mündet

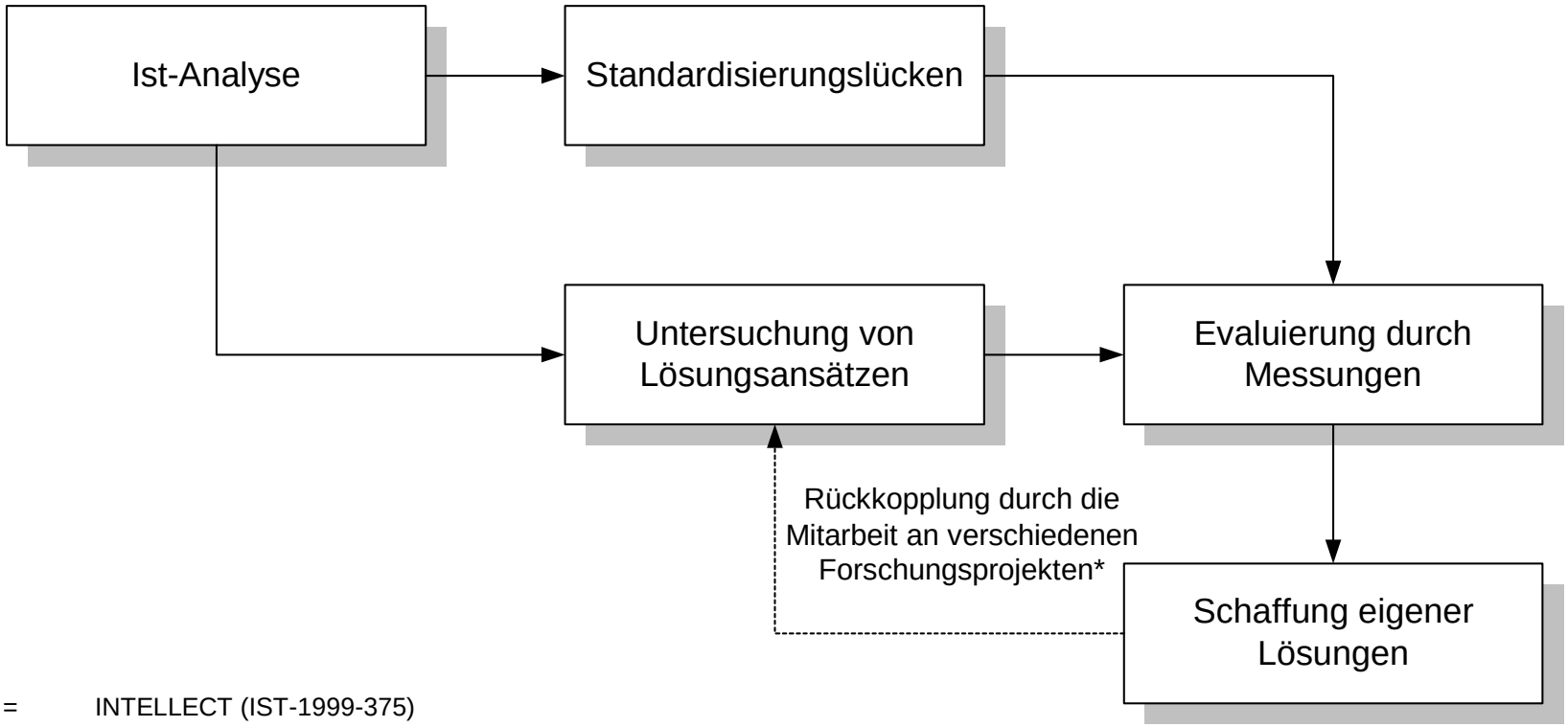
Eingangsthesen

1. Das Internet ist zum heutigen Zeitpunkt nicht **echtzeitfähig**. Das heißt, es können Audio-/Videoapplikationen sowie sensible Datenanwendungen nicht mit einer garantierten Qualität versehen werden
2. Heute sind **QoS-**Mechanismen grundsätzlich in heterogenen Netzen nicht anwendbar
3. Es fehlt an **Sicherheitsmechanismen**, die grundsätzlich ungenügend implementiert oder nicht in den Anwendungen verankert sind
4. Fehlende oder schlechte **Migrationsstrategien** zwischen Protokollen
5. Ineffizientes **Internetworking** zwischen TK- und Datennetzen
6. Es werden **Echtzeitdaten** über heutige Datennetze ineffizient übertragen bzw. es sind nicht die gleichen Qualitätsmaßstäbe erreichbar
7. Anwendungen und Dienste nehmen zu wenig Rücksicht auf die **Netzeigenschaften**

Unterteilung in die Hauptschwerpunkte

1. **Security:** Die Sicherheit spielt eine entscheidende Rolle bei der Akzeptanz heutiger IP-Lösungen. Ohne Vertraulichkeit, Integrität, Authentifizierung und Verschlüsselung werden eCommerce- und eBusiness- sowie Kommunikationslösungen im Internet sich nicht durchsetzen.
2. **Quality-of-Service (QoS):** Daten- und Echtzeitdienste müssen mit einer bestimmten Qualität angeboten werden können. Auf dem Weg in die Kommerzialisierung ist QoS eine wichtige Voraussetzung. Das Internet bietet bislang nur Best-effort.
3. **Traffic Engineering (TE):** Um eine Dienstgüte und damit Verzögerungszeiten garantieren zu können, sind TE-Mechanismen im Kernnetz notwendig. Bislang besitzt das Internet keinerlei solcher Mechanismen.
4. **Voice-over-IP (VoIP):** Echtzeitapplikationen setzen kurze und konstante Verzögerungszeiten voraus. Das Gleiche kann für Datenanwendungen gelten. VoIP (150 ms Latency, <50ms Jitter) ist eine sensible Anwendung, die eine hohe Qualität des Netzes voraussetzt.

Vorgehensweise



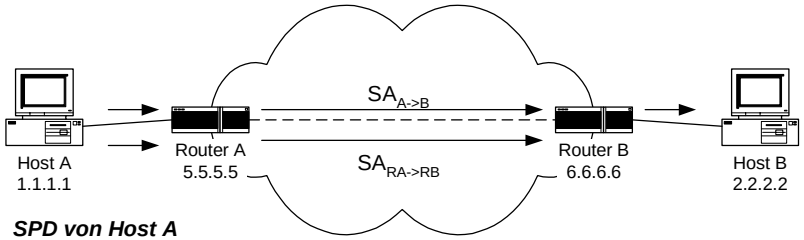
* = INTELLECT (IST-1999-375)
NGNI-SMONET (IST-2000-26418)
NGNI-VOIP (IST-2000-26418)
NOMAD (IST-2001-33292)

Lösungsansätze

Security für Echtzeitapplikationen

- Berücksichtigung der Anforderungen an eine Public Key Infrastructure (PKI)
- Aufbau einer sicheren VPN-Plattform mittels des Standards IPsec nach RFC-2401
- Integration eines einheitlichen Key-Management-Verfahrens (Internet Key Exchange – IKE) mittels Pre-Shared-Keys oder digitaler Signaturen
- Einsatz festgelegter und sicherer Verschlüsselungsverfahren (IPsec bietet drei Typen einer traditionelle Exponentiation über einen Primzahl-Modulus (MODP) sowie zwei Typen von elliptischen Kurven (EC2N))
- Einheitliche Implementierung der IPsec-Protokolle in den unterschiedlich teilnehmenden Geräten einer Security Association (SA)
- Implementierung sollte in Routern (keine Software-Clients) erfolgen, die allerdings zusätzliche Performance-Einbußen hinnehmen müssen

IPsec-Kommunikationsverarbeitung



SPD von Host A

Quelle	Senke	Protokoll	Port	Anwendungsstrategie
1.1.1.1	2.2.2.2	alle	alle	Transport-Modus AH mit HMAC-MD5

Ausgehende SADB von Host A

Ursprung	Ziel	Protokoll	SPI	SA-Eintrag	SA
1.1.1.1	2.2.2.2	AH	10	MD5-Schlüssel	$SA_{A->B}$

SPD von Router A

Quelle	Senke	Protokoll	Port	Anwendungsstrategie	Tunnelende
1.1.1/24	2.2.2/24	alle	alle	Tunnel-Modus ESP mit 3DES	6.6.6.6

Ausgehende SADB von Router A

Ursprung	Ziel	Protokoll	SPI	SA-Eintrag	SA
5.5.5.5	6.6.6.6	ESP-Tunnel	11	168-Bit 3DES-Schlüssel	$SA_{RA->RB}$

Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) Starting Site-to-Site IKE/IPSEC Client Mode Tunnel to 213.168.215.6
Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) Creating session 82360c to 213.168.215.6:500 via 217.225.152.183
Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) Begin Aggressive Mode Initiator to 213.168.215.6:500
Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) TO 213.168.215.6:500 284 Bytes, AG HDR SA KE NONCE ID (85caec)
Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) FROM 213.168.215.6:500 272 Bytes, AG HDR SA KE NONCE ID HASH (85caec)
Tue Oct 01 14:04:48 UTC IKE: Trace(*) (IKE) Accepting IKE = PreSharedKeys Modp1024 3DES HMAC-SHA
Tue Oct 01 14:04:51 UTC IKE: Trace(*) (IKE) Remote System/User Identity 213.168.215.6
Tue Oct 01 14:04:51 UTC IKE: Trace(*) (IKE) Aggressive Mode Initiator Done with 213.168.215.6:500; Lifetime=0S/0Kb

Tue Oct 01 14:05:21 UTC [52]: rcvd [LCP EchoRep id=0x3d magic=0x167589ee]
Tue Oct 01 14:05:26 UTC [52]: sent [LCP EchoReq id=0x3e magic=0x6fb99262]
Tue Oct 01 14:05:26 UTC IKE: Trace(*) (IKE) Session heartbeat timeout to 213.168.215.6:500. Destroying session.

Nov 30 05:33:08 Pluto[116]: "Bit" #1: initiating Aggressive Mode, state #1, connection "Bit"
Nov 30 05:33:15 Pluto[116]: "Bit" #1: STATE_AGGR_I2: sent AI2, ISAKMP SA established
Nov 30 05:33:15 Pluto[116]: "Bit" #2: initiating Quick Mode AGGRESSIVE+PSK+ENCRYPT+TUNNEL
Nov 30 05:33:17 syslog: setup: ...FreeS/WAN IPSEC started
Nov 30 05:33:17 Pluto[116]: "Bit" #2: STATE_QUICK_I2: sent QI2, IPsec SA established

IPsec: Standardisierungslücken*

- Wenn Preshared Key, Main Mode und dynamische IP-Adressen benutzt werden, muss der Preshared Key für alle IPsec-Clients identisch sein
- IPsec unterstützt nicht die traditionell im Remote Access eingesetzten unidirektionalen Authentisierungsmethoden RADIUS (PAP/CHAP), SecureID oder OTP
- Die IP-, DNS- und WINS-Adresszuweisung vom VPN-Gateway zum Client ist innerhalb des IKE-Protokolls nicht spezifiziert. Die privaten IP-, DNS- und WINS-Adressen müssten danach in jedem IPsec-Client fest konfiguriert werden
- Bei größeren Remote-Access-Netzen (>300 Teilnehmer) können Ressourceprobleme im VPN-Gateway auftreten. Die IPsec-Clients unterbrechen ihre Layer-2 PPP-Verbindung zum Provider immer wieder und löschen evtl. ihre eigenen SA, die nach wie vor im VPN-Gateway existieren
- Zwischen IPsec-Client und VPN-Gateway darf kein IP-NAT-Verfahren eingesetzt werden, da der IPsec-ESP-Header nicht über genügend Informationen verfügt. Setzt beispielsweise ein Filial-Router IP-NAT bei der Einwahl zum Provider ein, muss sich der Tunnelendpunkt im Router befinden. Das heißt, es findet nur eine Authentisierung des LAN statt, jedoch keine persönliche Authentisierung der Teilnehmer
- Bei großen Remote-Access-Installationen ist die Konfiguration und Administration der SPD-Einträge sowohl auf Client-Seite als auch im Zentralsystem sehr aufwendig

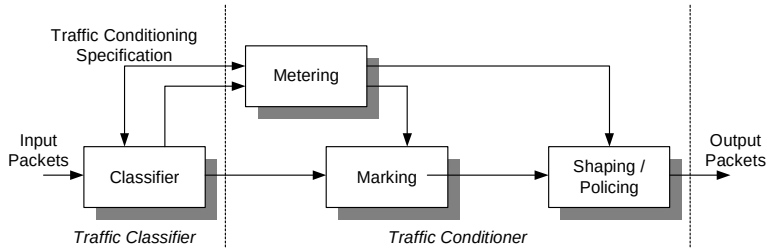
* u.a. bestätigt durch IETF: IPSEC Working Group

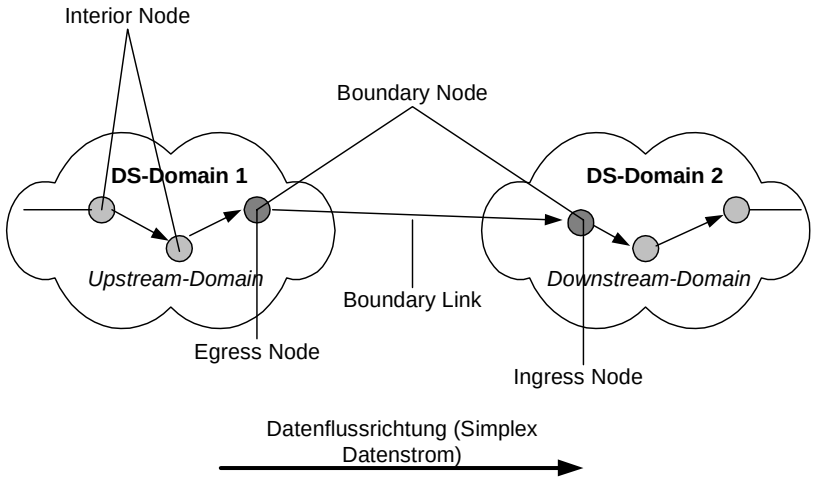
IP-basierte QoS-Ansätze (1)

- **IntServ-Ansatz**
  - Ressourcen müssen explizit verwaltet werden, um die Anforderungen der Anwendungen erfüllen zu können
  - Die Dienstgarantien für Echtzeitapplikationen können nicht ohne Reservierung von Ressourcen erfolgen
  - Die End-to-end-Verzögerungszeiten müssen begrenzt werden, um die dynamische Anpassung an sich ändernde Netzbedingungen gewährleisten zu können
  - Statistisches Aufteilen zwischen Echtzeit- und Datenapplikationen ist vorteilhaft, wenn man über eine gemeinsame Infrastruktur beide Anwendungen nutzen will
  - Nutzung von drei Klassen (Best-effort, Controlled Load Service, Guaranteed Service)
- **Evaluierung**
  - Vergleich des Verhaltens beim Best-effort-Verkehr mit dem CLNE-Service bei geringer Last
  - Vergleich auf einer stark belasteten Strecke. Dabei ist auf die Änderung der Gesamtverzögerung des CLNE-Verkehrs sowie die Änderung der Paketverlustrate zu achten
  - Erzeugen eines Datenstroms, der die angemeldete T_{SPEC} nicht einhält: Überschreitung der mittleren Datenrate sowie der Burstiness
  - Erzeugen einer Überlast mittels reservierter Datenströme ohne Best-effort-Verkehr

IP-basierte QoS-Ansätze (2)

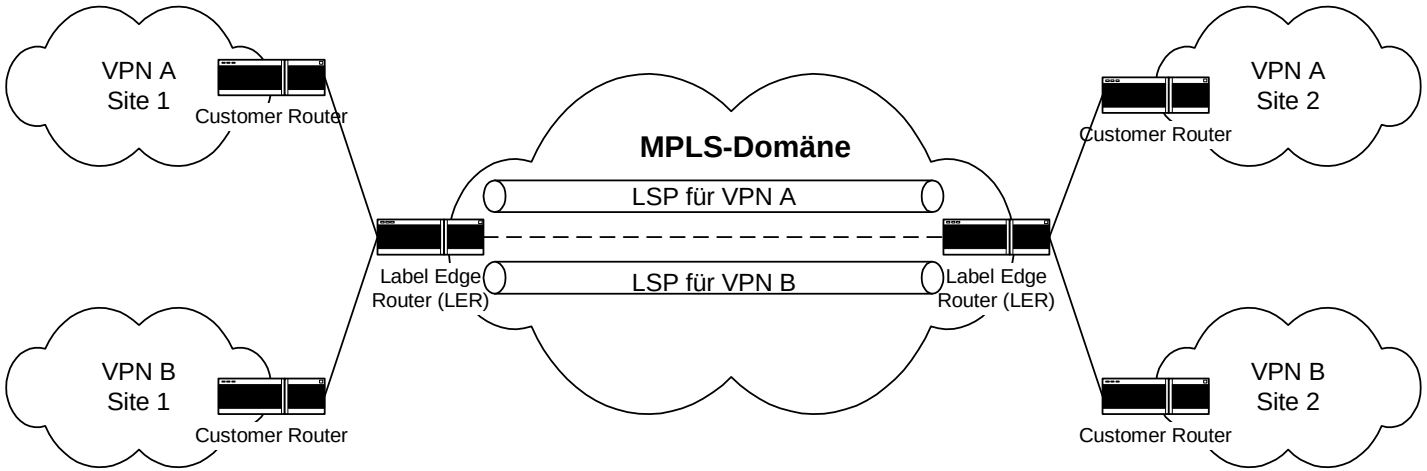
- **DiffServ-Ansatz**
  - Aggregate Flows
  - 2 oder mehr Flows in eine Kategorie
  - Ressourcen-Priorisierung
  - Keine Flow-State-Information
  - 3 Klassen (Best-effort, Assured Forwarding, Expedited Forwarding)
- **Evaluierung**
  - Im Mittelpunkt aller DiffServ-Betrachtungen steht eine administrative Einheit
  - Bildung von DiffServ-Clouds
  - Skalierbarkeit
  - Zuordnung von Priorisierungen für Applikationen
  - Sicherheitsimplementierung mittels IPsec (Authentifizierung im DSCP im Ingress Node)





Traffic Engineering (TE)

- Verbindung von VPN-Plattformen zu einem Gesamtnetz
- Garantie der Verzögerungszeiten, da der Routing-Pfad einmal festgelegt und eingehalten wird
- Sender bestimmt den Weg durch das Netz (Source Routing)
- Die Festlegung der Route im MPLS wird nicht pro Paket, sondern pro LSP getroffen!
- MPLS-Umsetzung ist u.a. mittels POS und ATM auf Layer-2-Schicht möglich
- Label Switching kann über mehre LSP gleichzeitig erfolgen



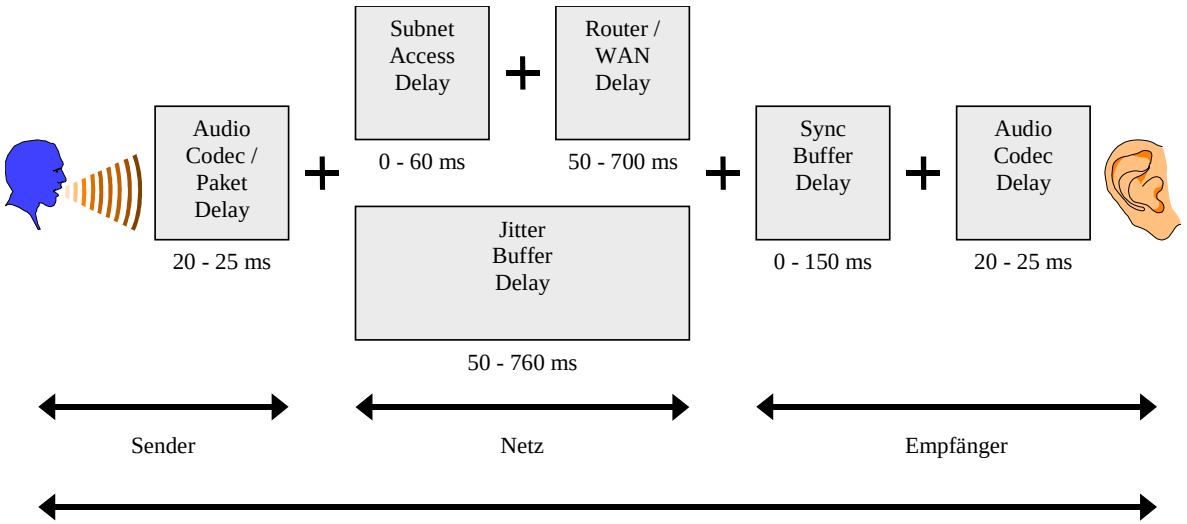
QoS/TE: Standardisierungslücken*

- **Keine End-to-end-Dienstgüte vorhanden:** Es gibt unterschiedliche Ansätze für unterschiedliche Technologien (Mapping notwendig!); proprietäre Verfahren der Hersteller; heterogenes Umfeld im Internet; IPv4 besitzt keine QoS-Mechanismen
- **Applikationen kennen keinen QoS/CoS:** Betriebssysteme unterstützen kaum QoS/CoS; Echtzeitapplikationen nutzen keine Stack-Implementierungen
- **Datennetze kennen keinen QoS/CoS:** Datennetze sind nicht auf Echtzeitanwendungen ausgelegt
- **Policies sind notwendig:** QoS kann nur über Policies verwaltet werden; diese sind kaum implementiert
- **TE-Signalisierungsprotokolle:** Unterschiedliche Protokolle zur Signalisierung (RSVP-TE, CR-LDP, MPLS-LDP) von QoS und Label Switching sind im Einsatz, die sich inkompatibel zueinander verhalten; zusätzlich verhalten diese sich auch unterschiedlich in Routing- oder LSP-Setup-Fragen bzw. sind unterschiedlich von den Herstellern implementiert
- **MPLS-Interoperabilität:** eine Fülle an Drafts (über 100) und RFCs (17) liegen vor, die eine hohe Komplexität und somit Interoperabilitätsprobleme bedeuten!

*u.a. bestätigt aus Arbeiten in NGNI-QOS: Infrastructure Working Group sowie IETF: MPLS Working Group und DiffServ Working Group

Voice-over-IP

- Hohe Verzögerung durch hohen Overhead (76% bei 20 Byte Nutzdaten) und Übertragungsverzögerung auf der Gesamtstrecke
- Unterschiedliche Paketgrößen und Laufzeiten verursachen Jitter- und Latenzeffekte
- Komprimierung der Sprache sorgt für geringere Anforderung an die Übertragungsrate
- Paketlängen von 300 Bit (ca. 38 Byte) bis 700 Bit (ca. 87 Byte) liefern optimale Ergebnisse für die Gesamtlaufzeit, da bei dieser Bitanzahl ein Minimum der Paketierzeit, unter Berücksichtigung kleiner Netzknotenverzögerungen durch weniger Paketaufkommen, erreicht werden kann
- Implementierung muss sorgfältig mit dem Netz abgestimmt werden



Gegenmaßnahmen:

- Pufferung
- Flusssteuerung
- Wahl des Vermittlungssystems
- Kanalauslastung und -kapazität
- Zusammenfassung der Sprach- und Datenströme

VoIP: Standardisierungslücken*

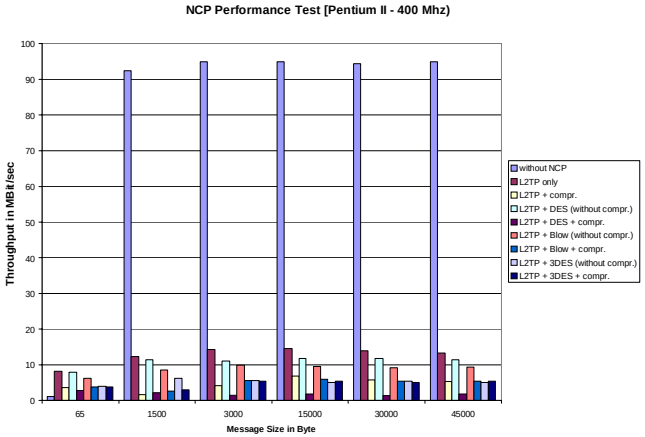
- Direkte Kommunikation bei eCommerce-Lösung über das Internet bislang kaum vorhanden, da geringe Möglichkeiten vorhanden sind und **kein QoS** genutzt wird
- Bislang ist die Sprach- und Bildqualität im Internet von der Tageszeit abhängig, da **unterschiedliche Lasten** auftreten. Diese Lasten lassen sich in einem heterogenen Umfeld nicht oder nur schwer beherrschen
- Videokonferenzsysteme als eine Möglichkeit Echtzeitdaten zu übertragen haben sich nicht durchsetzen können, da die **technische Implementierung unzureichend**** war, die Handhabung sich unfreundlich gestaltete sowie Multicast zur Gruppenkommunikation ein komplexes Themengebiet darstellt.
- **Sicherheitsprobleme:** Eine Verschlüsselung der Sprachdaten wird bislang weder verwendet noch von den Herstellern/Standards angeboten sowie fehlt eine Authentifizierung oder ist unzureichend (z.B. über SIP Security)
- Unterschiedliche **Signalisierungsprotokolle** für Endstationen (H.323 und SIP) sowie Gateways (MEGACO/H.248 und MGCP) sind vorhanden

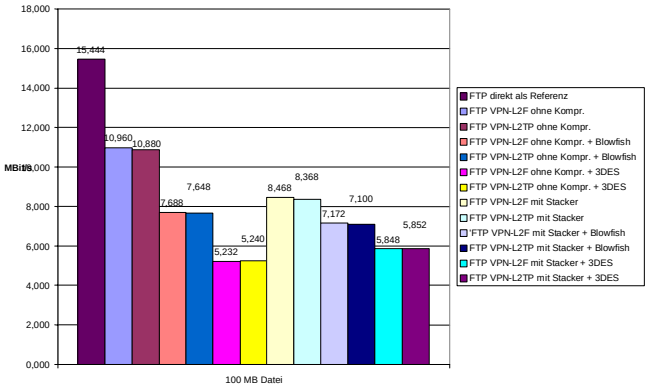
** z.B. Einsatz von 10-MBit/s-Ethernet, mit einer minimalen Paketierverzögerung von 1,2 ms

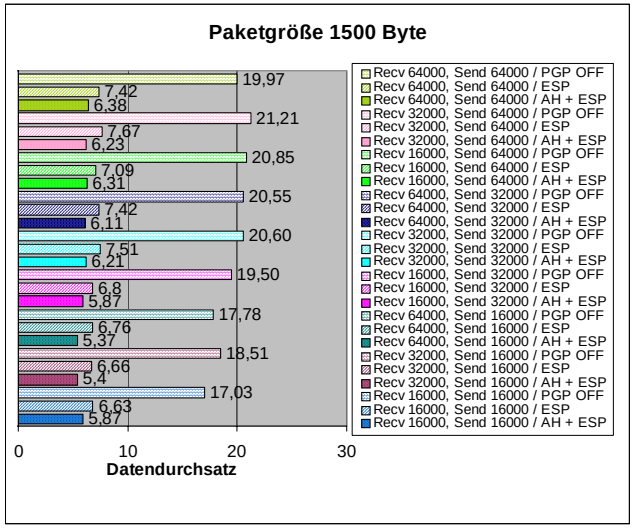
*u.a. bestätigt durch Arbeiten in NGNI-VOIP: New Services Working Group und INTELLECT sowie IETF: IPTEL Working Group

Messungen (Ausschnitte)

Security-Messungen





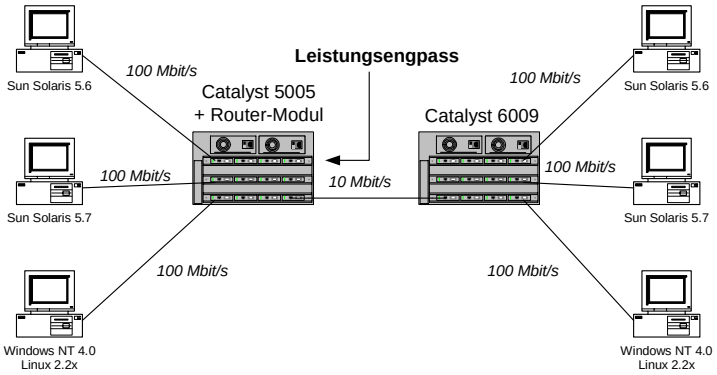


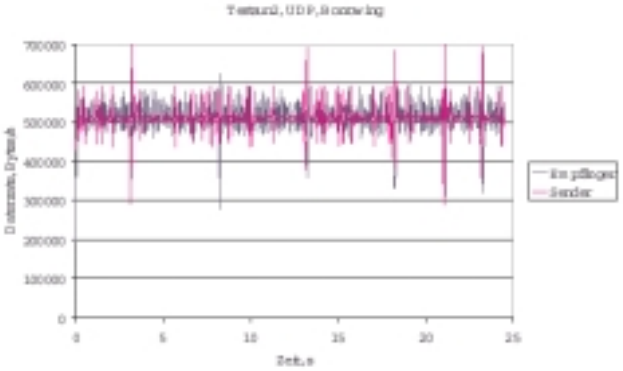
Performance	
ohne IPsec	315 Kbyte/s
IPsec ohne AH und ESP	47 Kbyte/s
IPsec mit AH	26 Kbyte/s
IPsec mit ESP (Transport- oder Tunnelmodus)	26 Kbyte/s
IPsec mit AH und ESP (Transportmodus)	20 Kbyte/s
IPsec mit AH und ESP (Tunnelmodus)	18 Kbyte/s

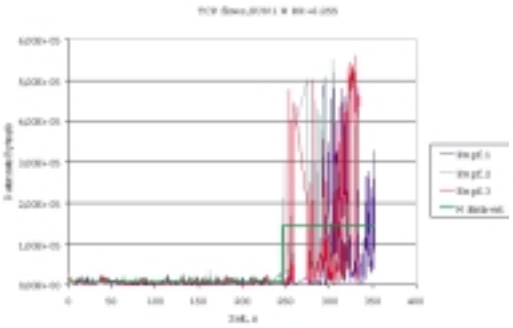
IPsec-Messungen: Ergebnisse

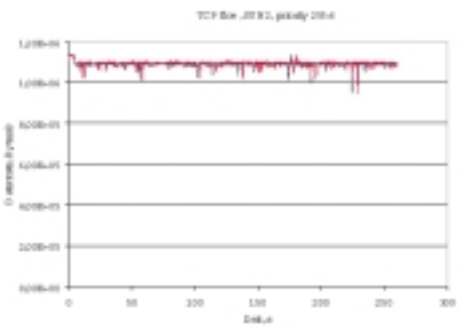
- Der **Transportmodus** von IPsec schützt vorrangig höhere Schichtenprotokolle, was wiederum für den Einsatz von End-to-end-Kommunikation spricht
- Um eine höchstmögliche Sicherheit bezüglich der eingesetzten Anwendungen erhalten zu können, muss heute AH und ESP im **Tunnelmodus** eingesetzt werden
- Nachteilig wirkten sich die erheblichen **Performance-Verlust** aus, die u.a. durch die Router-Verschlüsselung verursacht wurden
- Obwohl im Back-to-back-Betrieb getestet wurde, lag der Leistungsverlust bei einer **FTP-Anwendung** bei 500 kBit/s bei nur 6%, stieg dann aber auf 28% bei 1 MBit/s bzw. 62,9% bei 2 MBit/s an
- Bei der Übertragung einer **Dialoganwendung**, das heißt bei einem Austausch von Paketen geringer Größe, lag der Leistungsverlust bei 64 kBit/s bereits bei 22,2% und stieg dann kontinuierlich auf 63,7% bei 2 MBit/s an
- So kam es zu einer durchschnittlichen, minimalen **Verzögerung** von 12,57 % bei 64 KBit/s und bis zu 175,9 % bei einem Datendurchsatz von 2 MBit/s
- **Fazit**: IPsec ist bei heutigen Software-Implementierungen nur bedingt für Echtzeitanwendungen geeignet

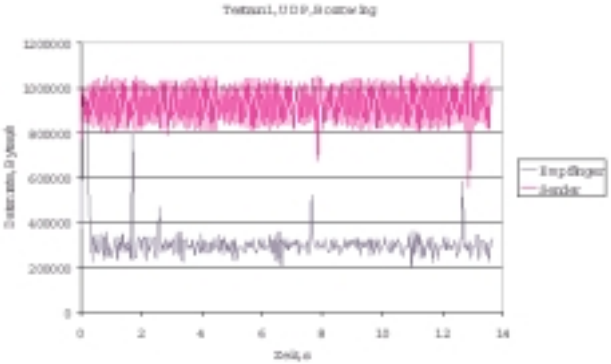
DiffServ-Messungen







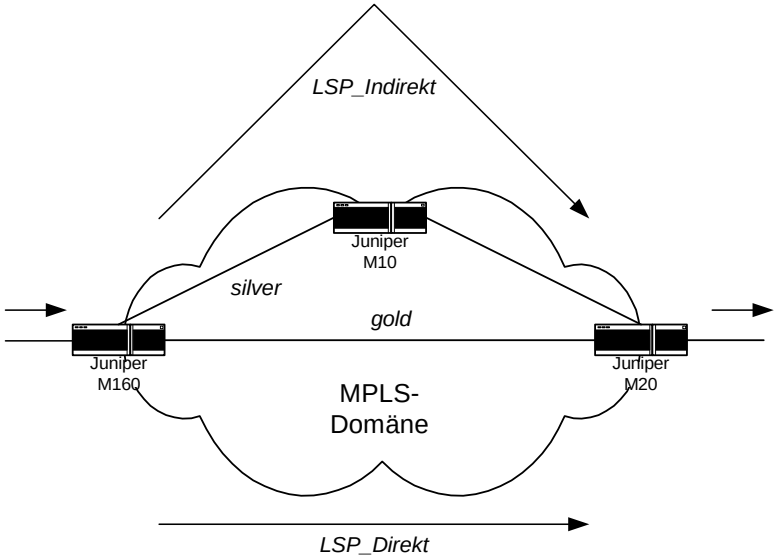




DiffServ-Messungen: Ergebnisse

- Wird die Queue über eine längere Zeitspanne überlastet, so ist der mittlere Füllgrad unabhängig von der Queue Size permanent knapp unter 100%. Somit ist die Wahrscheinlichkeit für das **Droppen** der Frames bei **größeren Bursts** (gilt auch bei größeren Paketen) höher als bei kurzen Datenpaketen
- Bei diesem Effekt kommt es im Wesentlichen auf das Verhältnis der **Burst Size** zur **Queue Size** an. Dieses Verhältnis wird mit der Teilung des gesamten Pufferplatzes in mehrere Queues immer größer
- Die **Verdrängung der Datenströme** war z.B. wesentlich schwächer ausgeprägt, wenn QoS auf dem Switch abgeschaltet war. Die Ursache dafür ist, dass in diesem Fall insgesamt eine Queue pro Port vorhanden war, die dann entsprechend größer wird
- Wenn keine **Trennung der Queues** möglich sind, ist keine QoS-Realisierung für multimediale Dienste realisierbar
- Verbindungen von **10 MBit/s** besitzen für Echtzeitapplikationen wie VoIP zu große Gesamtverzögerungen (minimale Paketierverzögerung von 1,2 ms)
- Für jeden **QoS-Dienst** wird je eine **zusätzliche Queue** benötigt, die bei Echtzeitapplikationen mit ausreichenden Reserven belegt werden sollte
- Das bedeutet, dass man das **Burstiness-Problem** mit der Einführung von QoS verschärft!

MPLS-Messungen




```
Jun  5 12:51:39 mpls lsp LSPIndirekt primary Down
Jun  5 12:51:39 mpls lsp LSPDirekt primary No Route
Jun  5 12:51:39 mpls lsp LSPIndirekt primary No Route
Jun  5 12:51:39 mpls lsp LSPDirekt primary Deselected as active
Jun  5 12:51:39 MPLS lsp LSPDirekt down on primary()
Jun  5 12:51:39 mpls lsp LSPIndirekt primary Deselected as active
Jun  5 12:51:39 MPLS lsp LSPIndirekt down on primary()
Jun  5 12:51:41 TED free LINK j2-re0.00(172.168.1.160)->Transit.00(172.168.1.10)
Jun  5 12:51:41 TED_2_CSPF Start
Jun  5 12:51:41 mpls lsp LSPDirekt primary CSPF: link down/deleted
Jun  5 12:51:41 CSPF adding path LSPDirekt(primary ) to CSPF queue 1
Jun  5 12:51:41 CSPF creating CSPF job
Jun  5 12:51:41 mpls lsp LSPIndirekt primary CSPF: link down/deleted
Jun  5 12:51:41 CSPF adding path LSPIndirekt(primary ) to CSPF queue 1
Jun  5 12:51:41 TED_2_CSPF end elapsed time 0.000159s
Jun  5 12:51:41 CSPF job starting
Jun  5 12:51:41 CSPF for path LSPDirekt(primary ), starting at j2-re0.00
Jun  5 12:51:41 CSPF final destination 172.168.1.20
Jun  5 12:51:41 CSPF starting from j2-re0.00 (172.168.1.160) to 172.168.1.20, hoplimit 254
Jun  5 12:51:41 CSPF Reached target
Jun  5 12:51:41 CSPF completed in 0.000083s
Jun  5 12:51:41 CSPF ERO for LSPDirekt(primary ) (1 hops)
Jun  5 12:51:41         node 10.0.2.2/32
Jun  5 12:51:41 mpls lsp LSPDirekt primary CSPF computation result accepted
Jun  5 12:51:41 mpls lsp LSPDirekt primary Clear Call
Jun  5 12:51:41 CSPF job starting
Jun  5 12:51:41 CSPF for path LSPIndirekt(primary ), starting at j2-re0.00
Jun  5 12:51:41         path exclude: 0x00000004
Jun  5 12:51:41 CSPF final destination 172.168.1.20
Jun  5 12:51:41 CSPF starting from j2-re0.00 (172.168.1.160) to 172.168.1.20, hoplimit 254
Jun  5 12:51:41         constrains exclude 0x00000004
Jun  5 12:51:41 CSPF completed in 0.000077s
Jun  5 12:51:41 CSPF couldn't find a route to 172.168.1.20
Jun  5 12:51:41 mpls lsp LSPIndirekt primary CSPF failed: no route toward 172.168.1.20
Jun  5 12:51:41 CSPF job done
Jun  5 12:51:41 mpls lsp LSPDirekt primary Up
Jun  5 12:51:41 mpls lsp LSPDirekt primary Record Route: 10.0.2.2 S
Jun  5 12:51:41 mpls lsp LSPDirekt primary Selected as active path
Jun  5 12:51:41 MPLS lsp LSPDirekt up on primary() Route 10.0.2.2 S
Jun  5 12:51:41 TED free LINK Transit.00(172.168.1.10)->j2-re0.00(172.168.1.160)
Jun  5 12:51:44 mpls lsp LSPIndirekt primary 10.0.2.2: Explicit Route: wrong delivery
Jun  5 12:51:53 mpls lsp LSPIndirekt primary No Route
```

Neuer Pfad für *LSPDirekt*

Kein alternativer Pfad für *LSPIndirekt* gefunden

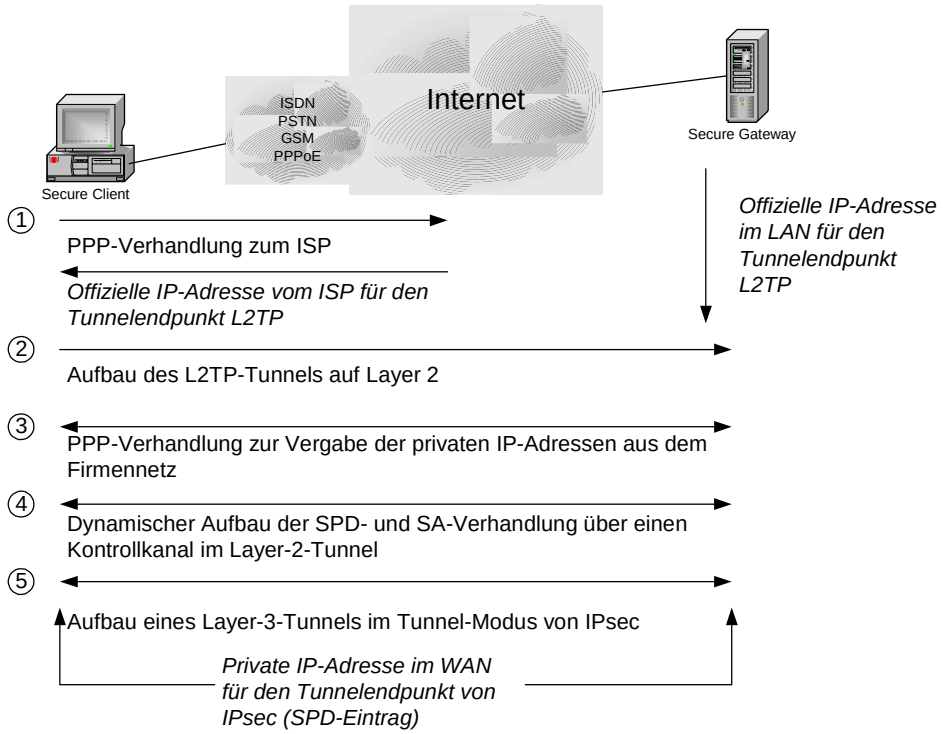
Ergebnisse MPLS

1. Für das **Re-routing** des LSP_Direkt musste der CSPF-Algorithmus einen alternativen Pfad zum M20 berechnen, welcher eine verfügbare Bandbreite von 100 MBit/s aufwies
2. Die einzig mögliche Verbindung bestand im Link M160-M10-M20
3. Aufgrund der **Reservierungen** des LSP_Indirekt standen hier aber nur 55 MBit/s zur Verfügung
4. Da die **Setup-Priorität** des LSP_Direkt aber höher als die **Holding-Priorität** des LSP_Indirekt ausgelegt war, musste der LSP_Indirekt abgebaut und seine Ressourcen für den Aufbau des LSP_Direkt zur Verfügung gestellt werden
5. Das darauf folgende Re-routing des LSP_Indirekt war nun aufgrund der verfügbaren Ressourcen und der **Pre-emption*-Werte** nicht mehr möglich, was ebenfalls die Messungen bestätigten
6. **Fazit**: es wurde der Routing-Pfad von der Quelle bestimmt und Reservierungen nach Dienstgütekriterien eingehalten!

* Die Pre-emption stellt eine Priorisierung des LSP dar und gliedert sich in Setup- und Holding-Priorität

Eigene Lösungen

Lösung der Remote-Access-Probleme von IPsec



- Lösung: IPSec-over-L2TP
- Nachteile
  - Höherer Overhead
  - Komplexität der Implementierung
  - unverschlüsselte Versenden der IP-Adressen, da IPsec erst nach dem Tunnelaufbau und anschließender PPP-Verhandlung aktiv wird

QoS in den Endgeräten

- Endsysteme müssen an einer QoS-Signalisierung teilnehmen, z.B. mit Hilfe von RSVP, COPS oder durch implizite Signalisierung, wie das Pre-Marking der Datenpakete
- Ein Endsystem muss unter Umständen die SLA in Form einzelner Übertragungsparameter kennen
- Es muss unter Umständen Datenpakete selektieren können. Das beinhaltet z.B. die Entscheidung, welche UDP-Datenströme priorisiert werden. Oftmals stellt sich die Frage auch allgemeiner: welcher Sender einer Multicast-Sitzung wird überhaupt empfangen?
- Durch das Problem der Burstiness kann ein System schon allein durch Traffic Shaping bzw. durch geschicktes Packet Scheduling bei einer Konkurrenz um Netzressourcen gewinnen
- Als Lösung wird eine Betriebssystemerweiterung auf Basis einer Middleware-Software oder auf einer Netzwerkkarte vorgeschlagen:
  - Dadurch können für jedes Protokoll bis zu 32 Port-Nummern bzw. Bereiche von Port-Nummern definiert werden, die einer QoS-Klasse zugeordnet werden
  - Es können Aliase sowohl für die Port-Nummern als auch für die QoS-Klassen vergeben werden.
  - Man kann keine IP-Adressen zur Filterung verwenden

QoS-Plattform (1)

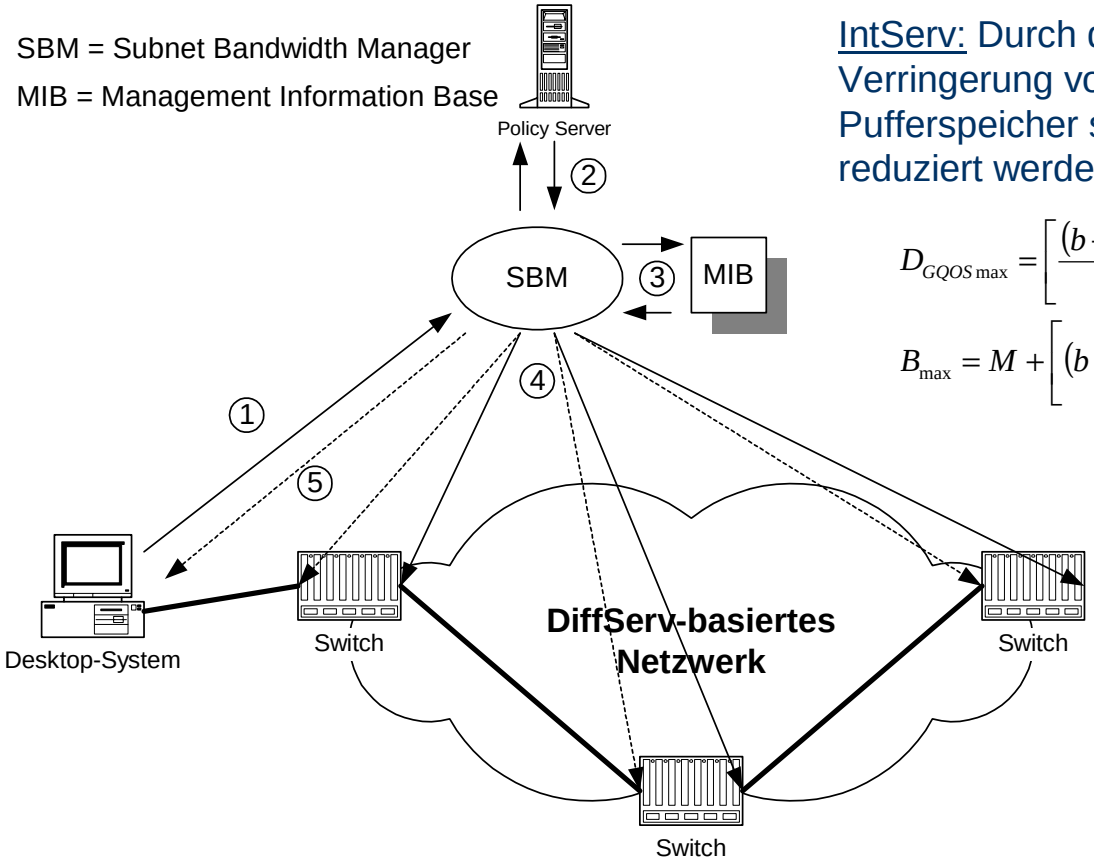
- Einführung eines Subnet Bandwidth Manager (SBM) zur Erkennung von Layer-2-Ressourcen
- Designated SBM (DSBM) zur Steuerung von Ressourcenanfragen
- Erweiterungen von RSVP sind dafür nötig, da PATH-Nachricht sich an den DSBM richten muss
- Definition von Mapping-Tabellen zwischen IEEE802.1D(p) (Layer 2), IntServ-/DiffServ-Ansätzen (Layer 3) und QoS-Agenten (Layer 4)
- Priorisierung wird vorgenommen, die aufgrund zusätzlicher Mechanismen wie Queueing mit einer hohen Wahrscheinlichkeit die angeforderte QoS unterstützt
- Einsatz von ATM im Kernnetz ermöglicht die garantierte QoS durch SVC-Verbindungen mit spezifischen Verkehrsparametern. Nachteilig ist der hohe Overhead und Managementaufwand
- Bei DiffServ ist das korrekte Scheduling-Verfahren zu wählen

Class-of-Service	IP Precedence	IP TOS	IEEE802.1D
Network Critical	7	14	7
Interactive Voice	6	10	6
Interactive Multimedia	5	14	5
Streaming Multimedia	4	4	4
Business Critical	3	6	3
Background	2	1	2
Best-effort	1	1	1
Standard	0	1	0

QoS-Plattform (2)

SBM = Subnet Bandwidth Manager

MIB = Management Information Base



IntServ: Durch die Vergrößerung von *R* und die Verringerung von *P* kann der Bedarf an Pufferspeicher sowie die Gesamtverzögerung reduziert werden:

$$D_{GQOS\max} = \left[\frac{(b-M)}{R}\cdot\frac{(P-R)}{(P-r)}\right] + \frac{M+C_{tot}}{R} + D_{tot}$$

$$B_{\max} = M + \left[(b-M)\cdot\frac{(P-R)}{(P-r)}\right] + C_{tot} + (D_{tot}\cdot R)$$

R = Reservierungsgrad
P = Peak Rate

Zusammenspiel IntServ-DiffServ

Grenze zwischen IntServ- und DiffServ-Regionen:

- Alle Router im Netz behandeln die Datenflüsse auf Microflow-Basis (IntServ). Lediglich auf einer Verbindung im Netz werden die Flüsse zu Datenflussaggregaten (DiffServ) zusammengefasst. In diesem Fall hat das Netz im Grunde die Eigenschaften einer IntServ-Region. Dabei werden aber Router, in denen extrem viele Microflows zusammenlaufen, wesentlich entlastet.
- Alle Router im Netz sind DiffServ-fähig und lediglich im Quell- und Ziel-Host werden die Daten auf Microflow-Basis behandelt. Dabei wird das gesamte Netz von der Per-Flow-RSVP-Signalisierung entlastet. Die Hosts haben aber die Möglichkeit, Ressourcen explizit vom Netz anzufordern sowie das Ergebnis der Anforderung zu erfahren.

Zusammenspiel MPLS-DiffServ

- Bei **DiffServ** werden die Pakete durch die Kundendomäne klassifiziert, überwacht und falls notwendig am Rande des Netzwerks verworfen. Dies wird durch Verkehrsprofile festgelegt, die in den SLAs zwischen dem Kunden und dem ISP verhandelt wurden. Nachdem die Pakete mittels DSCP markiert wurden, werden sie unterschiedlichen Warteschlangen (Queues) zugeordnet.
- Aufgrund des **Provisioning Factor (PF)** verhalten sich die Premium Queue (PQ), Assured Queue (AQ) und Default Queue (DQ) unterschiedlich, sodass auch ggf. Datenverluste entstehen können:

$$PF(PQ) > PF(AQ) > PF(DQ) > 1.0$$

- Durch die Verwendung von **MPLS** werden die IP-Header nicht nach DSCP-Informationen untersucht, da sie ein Label besitzen. Deshalb muss ein Mapping zwischen dem DSCP- und dem EXP-Feld des MPLS-Headers erfolgen
- Die **BA*-Klassifikation** muss dann aufgrund des EXP-Feldes im Kernnetz vorgenommen werden. Allerdings sind diese Felder unterschiedlich lang. Während DSCP 6 Bit anbietet, ermöglicht EXP nur 3 Bit: die ersten beiden Bits werden zur Beschreibung der Dienstklasse genutzt und das letzte Bit für Service Queues; der Rest wird nicht klassifiziert

* Behavior Aggregate: ein Bündel von Verkehrsflüssen bei DiffServ, die vom Netz auf gleiche Weise behandelt werden

Security bei VoIP

<table>
<tr><td>Audio - applika-tionen</td><td>Video-applika-tionen</td><td colspan="3">Terminal-Kontrolle und -Management</td><td>Daten</td></tr>
<tr><td>G.711
G.722
G.723
G.728</td><td>H.261
H.263</td><td rowspan="3">RTCP</td><td rowspan="3">Terminal zu Gatekeeper-Signalisierung

RAS</td><td rowspan="3">H.225.0
Q.931
Verbindungs-Signalisierung
(Call Setup)
SSL/TLS</td><td rowspan="3">H.245 Kontroll-kanal
SSL/ TLS</td><td>T.124</td></tr>
<tr><td colspan="2">Encryption</td><td rowspan="2">T.125</td></tr>
<tr><td colspan="2">RTP</td></tr>
<tr><td colspan="3">Unzuverlässiger Transport (UDP)</td><td colspan="2">Zuverlässiger Transport (TCP)</td><td rowspan="3">T.123</td></tr>
<tr><td colspan="5">Netzwerk Layer (IP) / IPsec</td></tr>
<tr><td colspan="5">Link Layer (IEEE 802.3)</td></tr>
<tr><td colspan="6">Physikalischer Layer (IEEE 802.3)</td></tr>
</table>

H.235:

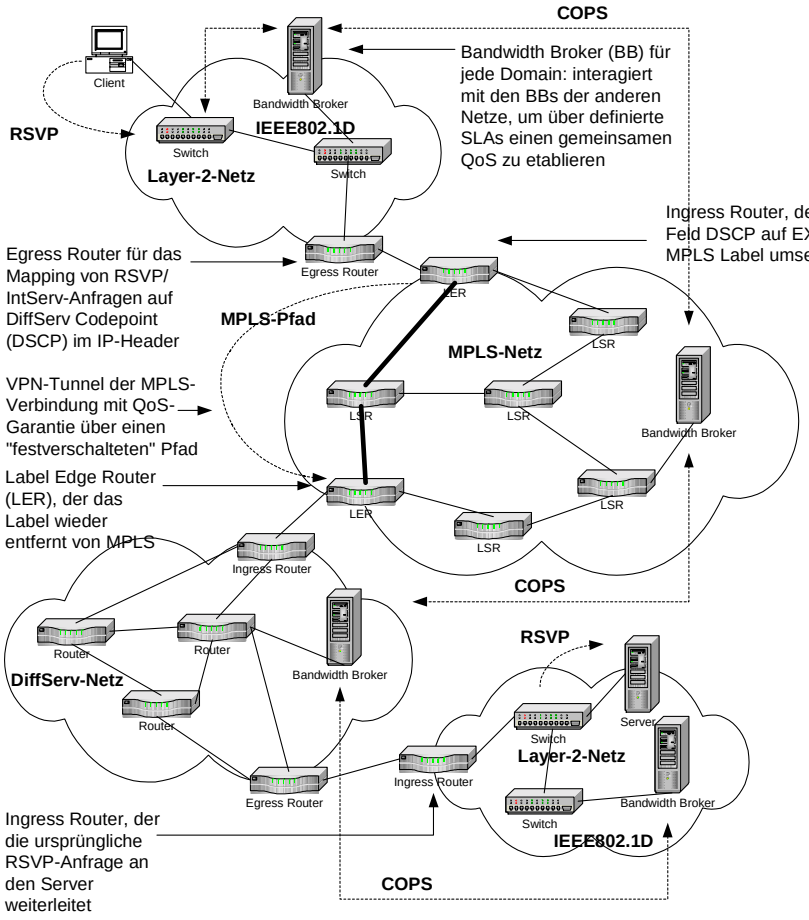
- **SecureChannel:** Bei Verwendung eines sicheren H.245-Kanals wird der Schlüssel nicht mehr zusätzlich geschützt, sondern im Klartext übertragen.
- **SharedSecret:** Wurden der geheime Schlüssel und der Kryptoalgorithmus bereits im Vorfeld etabliert, wird das Shared Secret verwendet, um den Sitzungsschlüssel zu verschlüsseln.
- **CertProtectedKey:** Zertifikate können eingesetzt werden, wenn der H.245-Kanal nicht sicher ist, oder als zusätzliche Maßnahme bei einem sicheren H.245-Kanal.

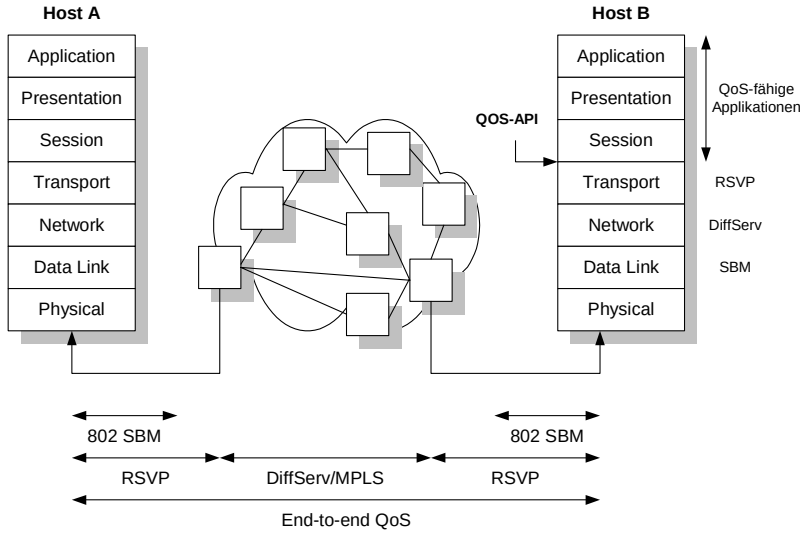
Wissenschaftlicher Fortschritt

Zusammenfassung der Ergebnisse

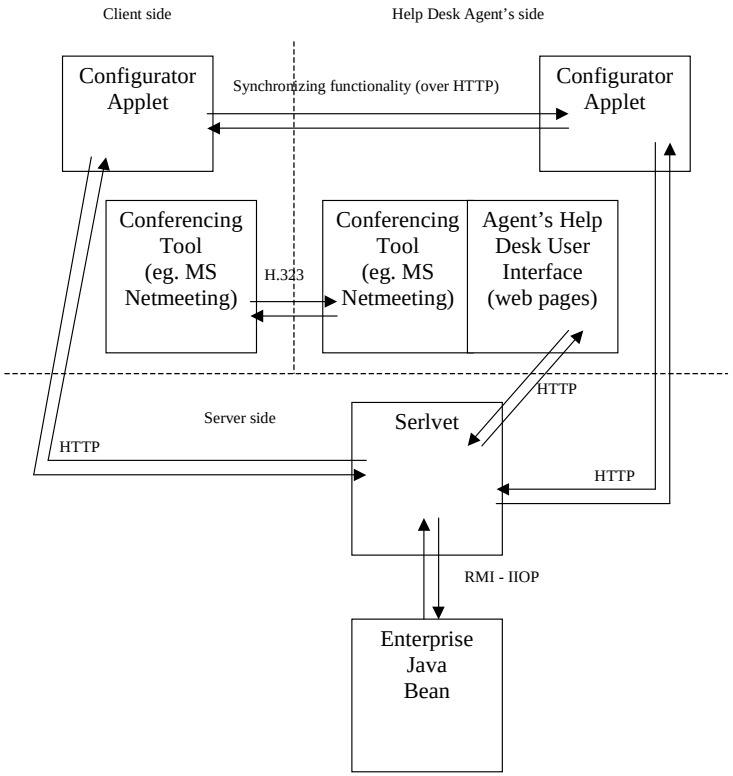
- Definition eines einheitlichen Standards im Bereich Security für das Projekt INTELLECT
- Aufsetzen einer Extranet-Lösung für den sicheren Informationsaustausch
- Sicherstellen des Internetworking zwischen unterschiedlichen Netzen und Protokollen auf verschiedenen OSI-Schichten
- Einführung von Quality-of-Service (QoS) in heterogener Netzumgebung basierend auf dem IP-Protokoll
- Einführung und Implementierung von Echtzeitapplikationen, die auf den definierten QoS-Mechanismen beruhen
- Konzeption und Implementierung eines neuen 3D eCommerce-Systems

Gesamtszenarien





Integration in ein Help-desk-System



- **Desktop-Agenten:** Ermöglicht QoS-Eigenschaften auf IP-Desktops zu übertragen.
- **Server-Agenten:** Fügt QoS-Eigenschaften zu Server-Applikationen hinzu.
- **Policy-Server:** Kontrolliert die Netzressourcen-Zuweisung, d.h., zwingt diese Vorgehensweise und Prioritätenvergabe dem Netz auf.
- **QoS-Manager:** Liefert die Netzübersicht betreffend dem QoS im Internet/WAN-Bereich

Implementierung

- **Konfiguration der Router**
  - Konfiguration von **Weighted Fair Queueing (WFQ)** auf den beteiligten Interfaces
  - Aktivierung von **RSVP** auf diesen Interfaces
  - Aktivierung von **NetFlow** (optional)
  - Aktivierung von **VoIP** Call Admission Control Using RSVP (nur Voice-Router)
  - Konfiguration einer **Priority-Queue** für RSVP-Verkehr (optional)
- **DiffServ-Konfiguration**
  - Anlegen einer **Access-List**: Die Access-List dient der Erkennung des zu priorisierenden Verkehrs
  - Konfiguration der **Class-Maps**: Class-Maps dienen dazu, Verkehr einer Dienstgütekategorie in einer Klasse zusammenzufassen
  - Konfiguration der **Policy-Maps**: Policy-Maps werden in Cisco-Routern verwendet, um alle Dienstklassen, die für ein Interface gelten sollen zu integrieren und ihnen die entsprechende Datenrate zuzuweisen
  - Zuweisen der Policy-Maps zu den entsprechenden Interfaces

…weitere Aufgaben

- Integration in die Weboberfläche wurde durch ActiveX umgesetzt (Sicherheitsproblematik!); zukünftig Java1.2
- IPv6 ist zwar bereits vorgesehen gewesen, allerdings fallen einige Probleme weg, die bislang noch aufgeführt wurden (u.a. die dynamischen IP-Adressen bei IPsec und NetMeeting)
- Erweiterung von Echtzeitapplikationen um QoS-Mechanismen
- Mobilitätsszenarien sind bislang nicht betrachtet worden, obwohl hier einige Anwendungen bestehen (z.B. bei Verwendung unterschiedlicher Trägernetze)
- Umsetzung auf Thin Clients (u.a. PDA) sollte beachtet werden
- Einheitliche Policy Control bei IntServ und DiffServ mit dynamischen Reservierungsmöglichkeiten
- Monitoring und Accounting von Diensten

Ende des Vortrags

Eröffnung der Diskussionsrunde

Veröffentlichungen/Vorträge 2002

1. Traffic Engineering für neue Qualitäts-Infrastrukturen - Qualitätszusicherung in IP-Netzen; ONLINE 2002; Congressband III - Next Generation Internet & IP-Services; 25. Europäische Kongreßmesse für Technische Kommunikation; ISBN 3-89077-233-1; Düsseldorf 2002
2. Von der Stange verkauft? - über E-Shop-Systeme und -Lösungen; NET 1-2/02; NET Verlagsservice GmbH; Woltersdorf 2002
3. Echtzeitplattformen für das Internet - Grundlagen, Lösungsansätze der sicheren Kommunikation mit QoS und VoIP; ISBN 3-8273-1914-5; Addison-Wesley Verlag; München 2002
4. Testfälle – VoIP-Software: Funktionalität, Qualität und System-voraussetzung; NET 03/02; NET Verlagsservice GmbH; Woltersdorf 2002
5. H.323 versus SIP im VoIP-Umfeld; Rubrik Datennetze; NET 03/02; NET Verlagsservice GmbH; Woltersdorf 2002
6. Sprachqualität von Voice over IP; Rubrik Datennetze; NET 04/02; NET Verlagsservice GmbH; Woltersdorf 2002
7. MPLS im Test - Testszenarien und Standards für Multiprotocol Label Switching; NET 05/02; NET Verlagsservice GmbH; Woltersdorf 2002
8. Multi-Protocol Label Switching; Handbuch der Telekommunikation; Verlagsgruppe Deutscher Wirtschaftsdienst; 90. Ergänzungslieferung; Juli 2002; ISBN 3-87156-096-0; Köln 2002
9. Wer braucht Quality-of-Service?; NET 07-08/02; NET Verlagsservice GmbH; Woltersdorf 2002
10. Schwachstellen bei der Internet-Sicherheit; NET 09/02; NET Verlagsservice GmbH; Woltersdorf 2002
11. Integrated Network Platform (INP) for Next Generation Networks (NGN); EURESCOM Summit 2002; Powerful Networks for Profitable Services; VDE Verlag GmbH; ISBN 3-8007-2727-7; Berlin 2002
12. Service Discovery Integrated Network Platform; Interworking 2002; IFIP Converged Networking Conference: Data and Real-time Communications over IP; Perth/Australia 2002