Duale Hochschule BW

Stuttgart

Studiengang Elektrotechnik

Labor Nachrichtentechnik

Versuch 1: Session Initiation Protocol (SIP)

Dipl. Ing. Wolfgang Lautenschlager

Gruppe: _____ Semester: _____

Versuchstag: _____

Teilnehmer:

Name: _____

Name: _____

Name: _____

1. SIP-Labor Übungen – Einleitende Erläuterungen

Im SIP-Labor lernen Sie den praktischen Umgang mit IP-basierten „Telefonanlagen und Telefonen“ kennen.

Als Grundlage für das SIP-Labor sind die IETF Protokolle SIP, SDP und RTP und die Einordnung in der Protokollarchitektur Voraussetzung für das weitere Verständnis. Das SIP-Labor besteht aus einer „Vermittlungseinrichtung“, bestehend aus einem SIP-Telefonieserver, der im Laboraufbau in einem lokalen Netzwerk mit den weiteren Komponenten wie z.B. den Telefonen vernetzt ist. In den Versuchen lernen Sie, wie und welche Parameter der Einrichtungen konfiguriert werden. Die Abläufe der Telefonanrufe lernen Sie mit Hilfe des Protokolltools Wireshark kennen. Wie Wireshark eingesetzt werden kann und wie das Tool konfiguriert werden muss ist ebenfalls Thema dieses SIP-Labors.

In realen Netzwerkumgebungen können Störungen auftreten, die z.B. beim Surfen im Internet keine große Rolle spielen, die aber bei realzeitkritischen Anwendungen wie der Sprachübertragung gravierende Störungen hervorrufen können. Den Einfluss dieser Störungen wird im SIP-Labor mit Hilfe des Netem-Tools, einem Netzwerkemulator, detailliert behandelt und soll Ihnen zu einem geschärften Blick verhelfen.

2. Protokolle IETF

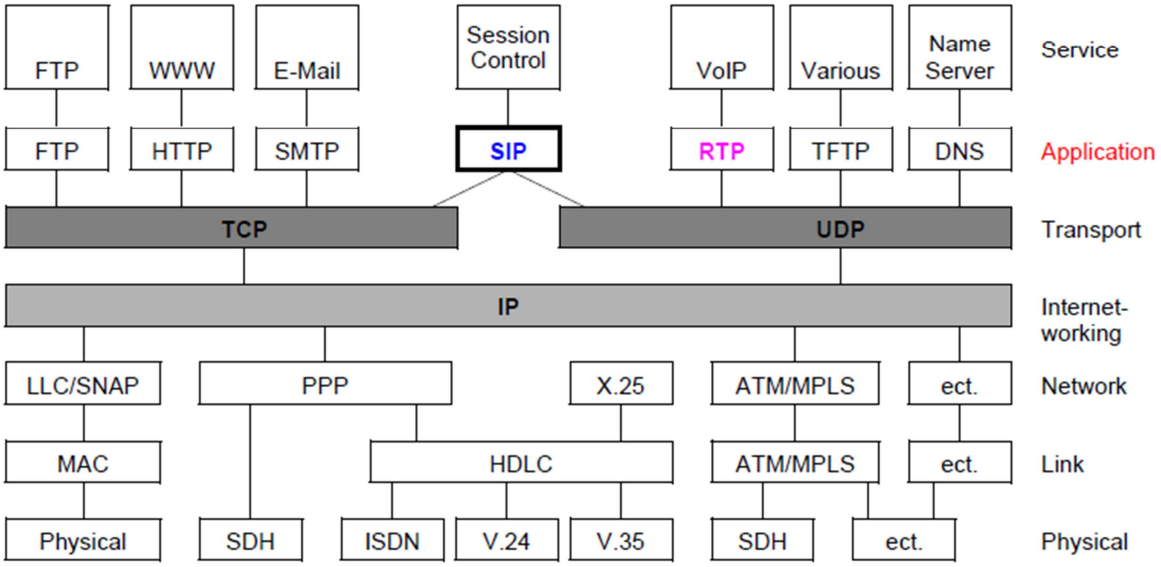
Das Internet wird mehr und mehr die Basis jeder Kommunikationsart. Durch die Einbeziehung von Echtzeitanwendungen wie Video, Sprache und Gaming kommen erweiterte Forderungen an das Netz. Allein die Übertragung der Nutzinformationen selbst reicht aber für die Bildung von steuerbaren Verbindungen nicht aus. Hierfür müssen Signalisierungsinformationen ausgetauscht werden, die im ursprünglichen Internet nicht verfügbar waren.

Die Internet Engineering Task Force (IETF) wählte Ende der 90’er Jahre einen etwas anderen Ansatz basierend auf einer ganzen Reihe von Protokollen. Für den Transport der Nutzinformationen wurde das Realtime Transport Protocol (RTP, (RTP, RFC 3550) gewählt, für die Steuerung der Verbindungen wurde das Session Initiation Protocol (SIP) definiert. Siehe hierzu Abbildung 1. Das SIP ist im Wesentlichen ein Signalisierungsprotokoll für Echtzeit- und Multimediaverbindungen im Internet. Oberhalb der Schicht 4 (gebildet von TCP und UDP) ist es im Sinne der IETF ein Application-Protocol, dass zwischen Client (z. B. im Terminal) und dem Server ausgetauscht wird. Auf dieser Basis entstanden zunächst VoIP-Anwendungen im Internet und Peer-to-Peer-Anwendungen für Sprachkommunikation. Dieser Ansatz wurde dann auch von den Herstellern der VoIP-TK-Anlagen und schließlich auch von der Standardisierung für die Weiterentwicklung von UMTS, der 3GPP, übernommen. Unter dem Begriff der Next Generation Networks (NGN) definierte die ITU-T ein architektonisches Konzept für das zukünftige Festnetz, dieses basiert auf den Internetprotokollen sowie u.a. RTP für die Übertragung von Nutzinformationen und SIP für die Steuerung der Sessions. Dieser Ansatz ist unabhängig von der jeweiligen Netzzugangstechnik.

Das Session Initiation Protocol (SIP, RFC 3261) ist dabei ein textbasiertes Client-Server-Anwendungsprotokoll, welches zum Aufbau, der Modifizierung oder Abbau von Multimedia Verbindungsbeziehungen zwischen ein oder mehreren Endpunkten

dient. In seinem Nachrichtenaufbau und deren Ablauf ist es HTTP sehr ähnlich. Das Session Description Protocol (SDP, RFC 2327) dient zur Beschreibung des Inhalts und Formats von Multimediaverbindungen und wird im sog. Message Body von SIP übertragen. Die SIP-Nachrichten werden mittels UDP (Standard) oder TCP (die Ausnahme) transportiert, standardmäßig ist hierfür der Port 5060 vorgesehen.

Abbildung 1: Protokoll-Stack in der Internet-Welt



Grundsätzlicher Ablauf: Ein SIP-Terminal besteht aus einem User-Agent-Client (UAC), der Anfragen an den User-Agent-Server (UAS) richtet. Die Anfragen, Requests, Nachrichten werden in ihrem Anwendungszweck, der sog. Methode, engl. Method, unterschieden. Der UAS antwortet mit Responses. Diese Rollen können wechseln: ein SIP Terminal kann für Anfragen auch ein Server sein, die die Anfragen mit ein oder auch mehreren Responses beantwortet. Der SIP-Proxy-Server dient zur Weiterleitung der Steuernachrichten im Netz.

FRAGE 1

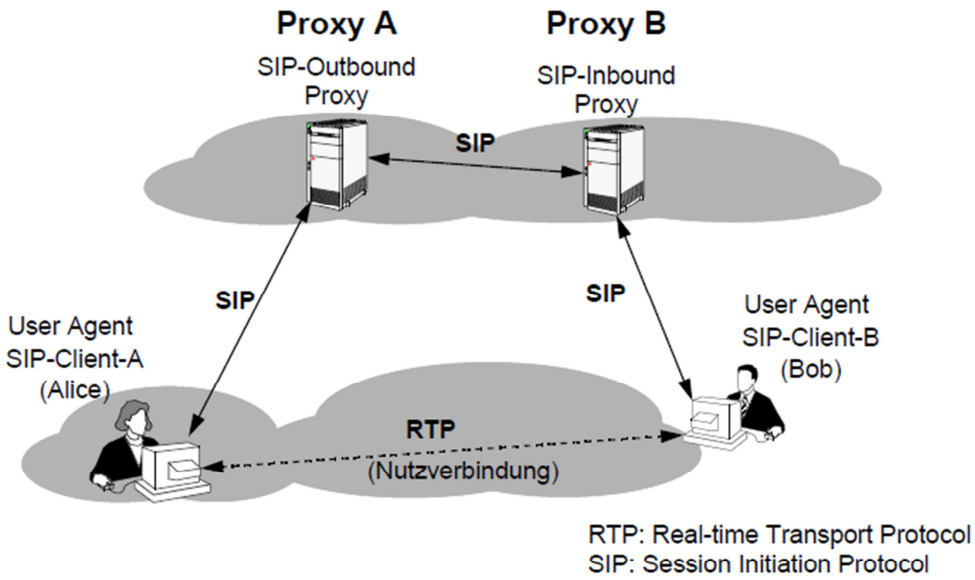
Was ist der grundlegende Unterschied zwischen einer Telefonverbindung im heutigen Telekommunikationsnetz und einer VoIP-basierten Telefonverbindung?

3. Architektur

Funktional wird bei einer Verbindung zwischen zwei User Agents zwischen dem Outbound- und dem Inbound-Proxy-Server unterschieden. In Abbildung 2 ist der Proxy A der Outbound-Server für den User Agent A (für Alice), dieser unterstützt den User Agent von Alice um abgehende Gespräche (Outbound) aufzubauen. Auf der anderen Seite ist der Proxy B der Inbound-Proxy für den User Agent von Bob, dieser unterstützt den User Agent von Bob um kommende Sessions entgegen zu nehmen (Inbound). Man spricht bei dieser Darstellung auch vom sog. SIP-Trapezoid, aufgrund der sich ergebenden Darstellungsform, dem Trapez. Nach der ursprünglichen Idee dieser Architektur sind die Proxy-Server nur für den Aufbau der Sessions notwendig. Die Nutzverbindung, auf der Basis des Real-time Transport Protocol (RTP), zwischen A

und B wird, nach dem ersten Verbindungsaufbau, direkt zwischen beiden aufgebaut. Die Proxy sind dann nicht mehr notwendig. Das hier dargestellte Trapez ist also nicht in allen Zuständen in dieser Form existent.

Abbildung 2: SIP-Trapezoid



Die Bezeichnungen „Outbound“ und „Inbound“ hängen von der jeweiligen Betrachtungsweise und der jeweiligen Richtung des Aufbaus von Sessions ab. In der Praxis lassen sich beide nur für einen Augenblick einer bestimmten Session logisch unterscheiden. Die Realisierung umfasst immer beide Funktionen und häufig werden noch weitere logische Einheiten im gleichen Gerät implementiert. Beispiele für solche Server wären der DHCP-Server oder der SIP-Registrar, bei dem die SIP-Clients registriert werden.

FRAGE 2

Welcher gravierender Unterschied besteht zwischen HTTP und SIP in Bezug auf die Client-Server Architektur, wenn man Endgerät (Telefon bzw. PC) und Netzwerkknoten (Proxy bzw. Webserver) betrachtet. Welche Funktionalitäten können im Fall HTTP bzw. SIP im Endgerät vorhanden sein und welche im Netzwerkknoten?

3.1 Adressen

Nach den IETF-Festlegungen können diverse Adressierungsarten für die Selektion eines Terminals verwendet werden. Mit SIP können E-Mail-artige Adressen genauso wie E.164-Telefonrufnummern verwendet werden. Hier einige Beispiele für die SIP-Adressierung:

- sip:user@domain, Beispiel: sip:FranzMaier@firma.de
- sip:user@host, Beispiel: sip:Maier@sipserver.firma.de
- sip:user@IP_Address, Beispiel: sip:Maier@249.198.241.30

- sip:phone_number@domain, Beispiel: sip:+49-711-12345678@sipgate.com
- Tel-URI (RFC2806): Beispiel: tel:+49-711-123456

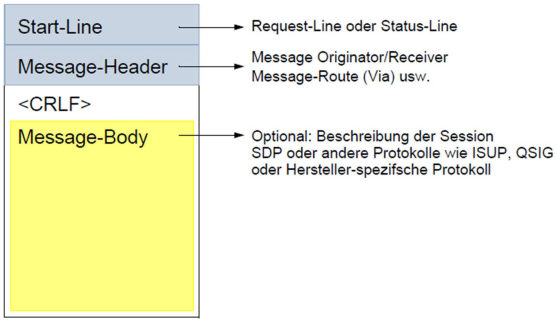
3.2 SIP-Nachrichten

Das Session Initiation Protocol (SIP) selbst ist ein textbasiertes Client-Server-Protokoll. Ein SIP Request (Methode, engl. Method) und Response bestehen aus textuellen Beschreibungen, jede SIP-Nachricht besteht aus einer Start-Line, einem Header und einem Body, siehe Abbildung 3.

- Die Start-Line kennzeichnet den Typ der folgenden Nachricht und die verwendete SIP-Version. Unterschieden wird dabei zwischen Request und Response, verwendet wird dann jeweils entweder eine Request- oder Status-Line.
- Der SIP-Header enthält Verbindungsparameter und Dienstinformationen. Hier sind grundsätzliche Informationen zur Session enthalten. Beispiel hierfür sind: wer ist der Initiator einer Session, wer dessen Partner und welcher Weg wird verwendet. Viele Definitionen wurden hierbei von HTTP übernommen, neu sind Felder wie Call-ID, CSeq, Via, usw.).

Der SIP-Body enthält die Beschreibung der Nutz-Verbindung, hierfür kann ein eigenes Protokoll verwendet werden, wie beispielsweise das Session Description Protocol (SDP). Beschrieben werden hiermit beispielsweise der verwendete Audio- bzw. Video-Codec sowie dessen Parameter, IP-Adressen, TCP-Portnummer, Time-of-Session.

Abbildung 3: SIP-Nachricht



3.2.1 SIP-Request

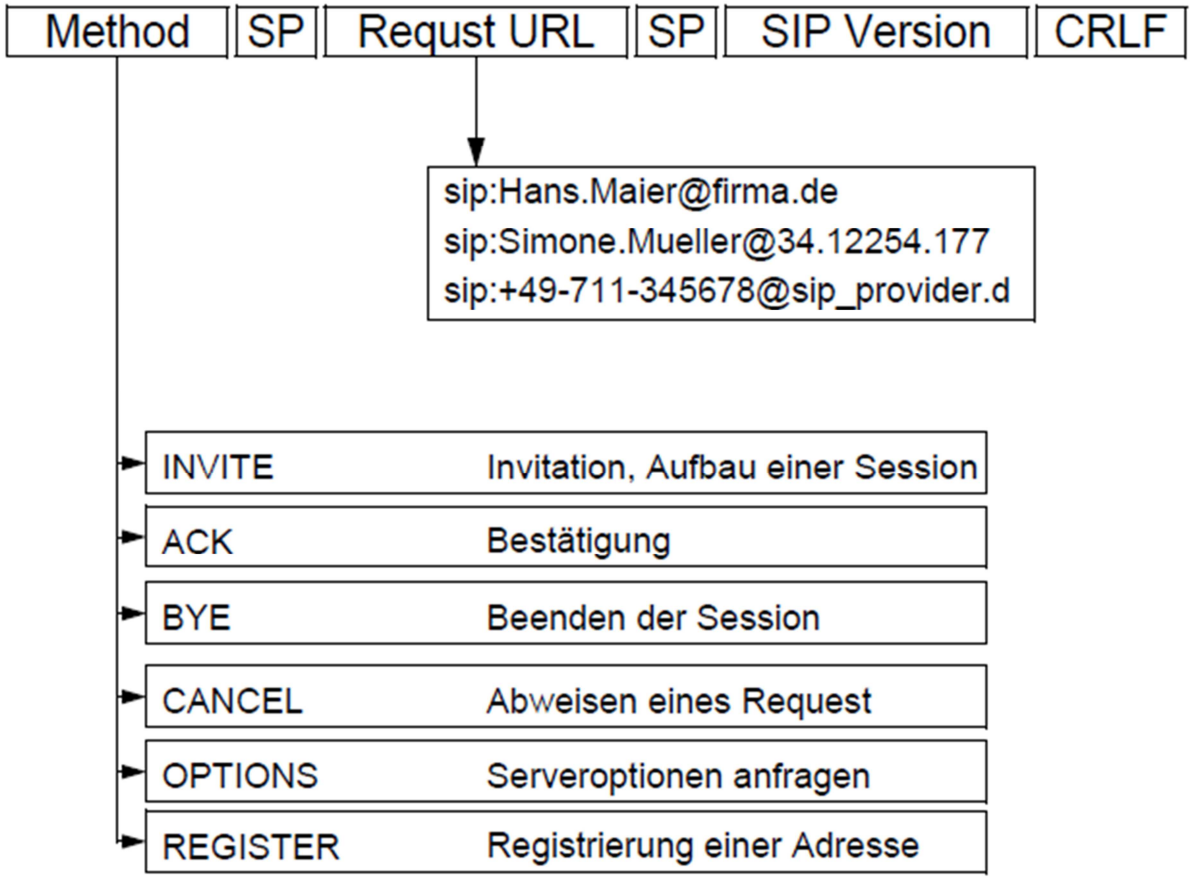
Ein SIP-Request beginnt mit einer Request Line, die drei Elemente beinhaltet:

- Request Method: die den Request-Typ kennzeichnet (INVITE, ACK, BYE usw.) und mit einem Leerzeichen (blank Space, SP) endet.
- Request URL: Der Uniform Ressource Locator (URL) ist die Adresse, an den diese Anfrage (Request) gerichtet wird. Auch dieses Element wird mit einem Leerzeichen abgeschlossen.
- SIP Version: Hiermit wird die verwendete Version von SIP für die Interpretation der Nachrichten und deren Ablauf gekennzeichnet. Hiermit soll die gesicherte

Interpretation der Nachrichten garantiert werden, ggf. kann auch zwischen Client und Server die SIP-Version verhandelt werden.

Die einzelnen Elemente werden durch Leerzeichen (Space, SP) voneinander getrennt, siehe Abbildung 4. Zum Abschluss trennet eine Leerzeile (Carriage Return, Line Feed – CRLF) Request Line und Header der SIP-Nachricht.

Abbildung 4: Aufbau einer SIP-Request Method



- SIP unterscheidet die folgenden Grundtypen von Nachrichten (Methods):
- INVITE: lädt einen Kommunikationspartner beispielsweise zu einem Telefongespräch, einer Videokommunikation, einem Datenaustausch oder einer Konferenz ein (Beginn eines Dialogs). Mit dieser Verbindungsaufforderung wird immer eine wechselseitige Kommunikation etabliert, nur durch spezielle Parameter im Session Description Protocol (SDP) können auch einseitige Verbindungen (send-only oder receive-only) angefordert werden.
- ACK: Positive Bestätigung, unterstützt damit einen gesicherten Nachrichtenaustausch. ACK wird als zusätzliche Bestätigung (nach ggf. Ringing und OK) auf einen INVITE-Request gesendet und ist selbst aber ein Request, der nicht mit einer Response beantwortet wird.
- BYE: Hiermit beendet ein User Agent eine bestehende Kommunikation, die Nutzverbindung wird aufgehoben (Beenden eines Dialogs).

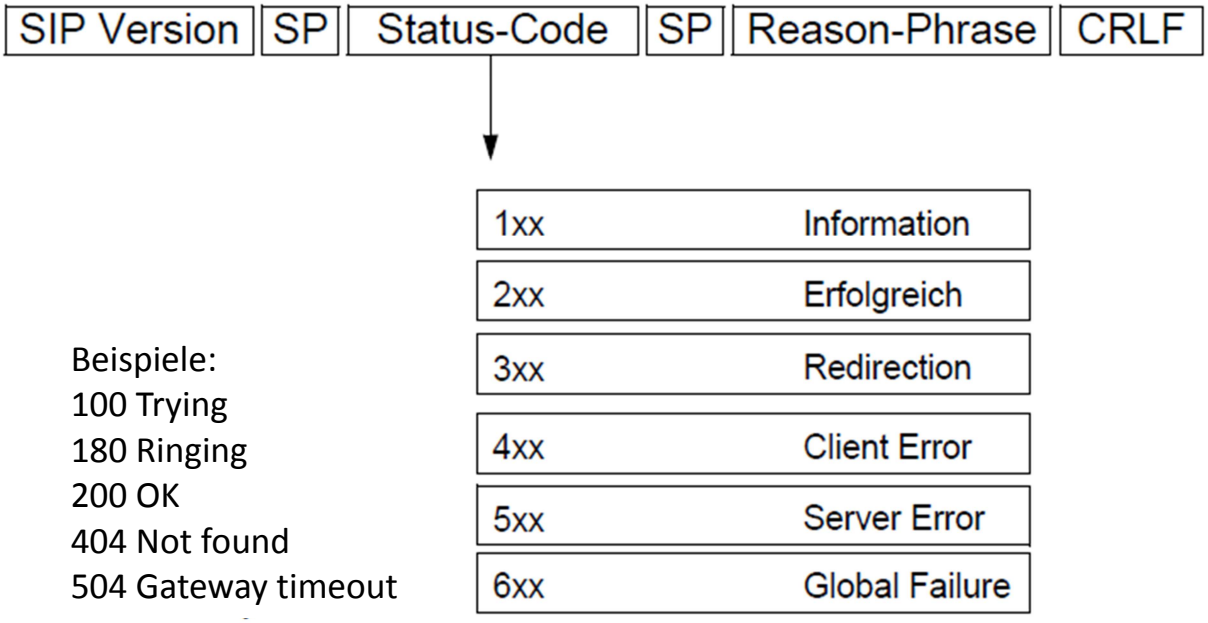
- CANCEL: Abweisen des gesendeten Request, beispielsweise die Abweisung einer kommenden Verbindung als Antwort auf die gesendete INVITE. Die Zuordnung der CANCEL-Nachricht zum abgewiesenen Request erfolgt mit Hilfe der Call-ID.
- OPTIONS: Mit dieser Methode kann ein User Agent Informationen zu den Eigenschaften (Codecs oder unterstützte Methodes) von Endsystemen (User Agents) erfragen und bereitstellen, ohne direkt eine Verbindung aufzubauen.
- REGISTER: Übermittelt Standortinformationen (genaue Adress-Informationen) über einen Benutzer. Die Benutzeridentifikation (Name, URI oder Rufnummer) wird dadurch der augenblicklichen IP-Adresse zugeordnet.

3.2.2 SIP-Response

Im Unterschied zu den Request-Methoden haben die Response-Nachrichten einen geringfügig unterschiedlichen Aufbau, siehe Abbildung 5. Am Anfang jeder Response steht die Status Line (vergleichbar der Request Line bei den Anfragen), diese beinhaltet:

- SIP-Version: Wie bei den Anfragen wird hiermit die für den Nachrichtenaufbau und -ablauf verwendete Version von SIP übermittelt. Der SIP-Version folgt ein Leerzeichen (Single Space).
- Status-Code: Antwort auf die Anfrage im Request. Siehe Abbildung 5
- Reason Phrase: Nach einem weiteren Leerzeichen kommt dann der Status-Code in Textform (Code: 200 – Text: OK, Code: 180 – Text: RINGING usw.). Die Zeile wird dann mit einem CRLF abgeschlossen.

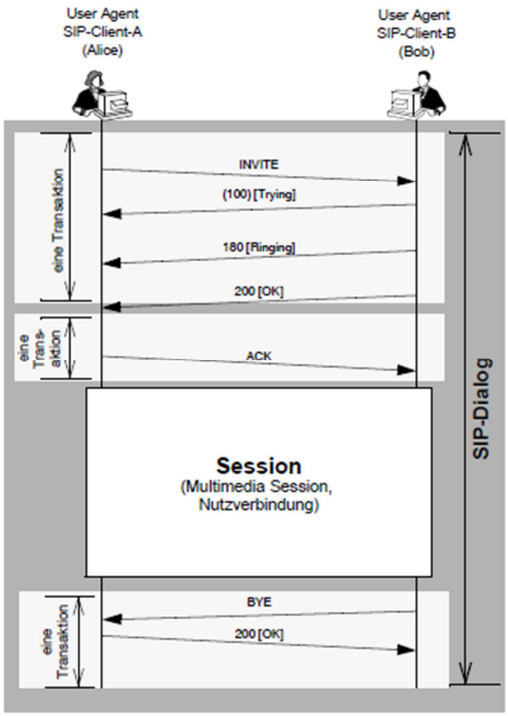
Abbildung 5: Aufbau einer SIP-Response Nachricht



3.2.3 Transaktion, Dialog und Session

Ein SIP-Request beginnt eine Transaktion (Transaction), zu der mehrere Bestätigungen oder Antworten (Response) gehören können, siehe Abbildung 6. Der Austausch der SIP-Nachrichten zwischen zwei SIP-Instanzen wird als SIP-Dialog bezeichnet. Die Aufgabe der SIP-Nachrichten ist es (unter anderem), Multimedia-Verbindungen (Sessions) zwischen zwei oder mehreren Instanzen aufzubauen, zu unterhalten, zu modifizieren und auch wieder abzubauen.

Abbildung 6: SIP Transaktion



3.2.4 Statusinformationen

SIP-Anfragen werden vom SIP-Server mit einem oder mehreren Statusinformationen (Response) quittiert, siehe Abbildung 6. Die Statusmeldungen bestehen immer aus einem Status-Code (100, 404 usw.) und der Rückmeldung in Text-Form, Reason-Phrases), die eine genauere Angabe für das Senden der Statusmeldung (die Begründung) liefert. Die Statusinformationen (Status Codes, hier nur Ausschnitte) sind in sechs Grundtypen eingeteilt:

- Informational (1xx, Ausschnitt): Diese Status Codes kennzeichnen Antworten nach dem Empfang einer Anforderung, deren Bearbeitung noch nicht abgeschlossen ist:
    - 100 Trying,
    - 180 Ringing,
    - 183 Session Progress.
- Success (2xx): Die Anforderung wurde empfangen und erfolgreich bearbeitet.
    - 200 OK.
- Redirection (3xx, Ausschnitt): Die Anforderung wurde nicht vollständig bearbeitet, es sind weitere Aktionen erforderlich:
    - 301 Moved Permanently,
    - 302 Moved Temporarily,
- Client Error (4xx, Ausschnitt): Negative Rückmeldung, die Anforderung kann nicht erfüllt oder bearbeitet werden:
    - 401 Unauthorized,
    - 403 Forbidden,
    - 404 Not Found,
    - 405 Method not Allowed,
    - 486 Busy Here,
    - 487 Request Terminated,
- Server Error (5xx, Ausschnitt): Die Anforderung war fehlerhaft oder kann von diesem Server nicht bearbeitet werden:
    - 500 Internal Server Error,
    - 504 Gateway Time-out,
- Global Failure: Die Anforderung kann von keinem Server bearbeitet werden:
    - 600 Busy Everywhere,

FRAGE 3

Wie erfolgt die Kommunikation zwischen Client und Server und was kann dabei abgehandelt werden?

FRAGE 4

Aus welchen Teilen besteht eine SIP- (oder auch HTTP-) Nachricht?

3.2.5 Back-to Back User Agent

In SIP-Architekturen können, je nach Aufgabenstellung, die Proxy unterschiedlich ausgeführt sein. Man unterscheidet:

- Stateless Proxy Server: Dieser Proxy leitet die Nachrichten weiter, ohne den Zustand des Session-Aufbaus zu speichern.
- Stateful Proxy Server: Dieser Proxy speichert den Zustand der Transaktion bzw. des Session-Aufbaus und kann in Abhängigkeit von den empfangenen Nachrichten unterschiedlich agieren.

Ein Back-to-Back User Agent (B2BUA) ist ein logisches Netzelement in SIP-Anwendungen mit der Möglichkeit zur Steuerung von VoIP-Multimediaverbindungen

- Der Back-to-Back User Agent (B2BUA) beinhaltet zwei getrennte User Agents, die „Rücken-an-Rücken" zusammengeschaltet sind (Back-to-Back). In beiden Richtungen werden die Verbindungen in diesem Gerät beendet (terminiert), bzw. neu begonnen (initiiert) und in zwei sog. Call Legs aufgeteilt.
- Da sämtliche SIP-Nachrichten zur Verbindungssteuerung den Der Back-to-Back User Agent passieren können damit Leistungsmerkmale wie z.B. Call Management (billing, automatic call disconnection, call transfer, usw.) realisiert werden.
- Ein Back-to-Back User Agent kann auch in Kombination mit einem Media Gateway eingesetzt werden und wird dann auch als Session Border Controller (SBC) bezeichnet. Dieser terminiert die Session und den Dialog, d. h. die Signalisierung und den Nutzkanal.

3.3 Session Description Protocol

Im ersten Teil der Beschreibung werden generelle Aspekte, Rahmenbedingungen und übergreifende Parameter der gesamten Verbindung dargestellt. Im Folgenden sind Parameter der Session Description für eine einfache Verbindung (Ausschnitt) dargestellt:

v=Protokoll-Version

o= Originator - Initiator der Session, Session-Identifizierung

s=Session Name, Name der Session

i=zusätzliche Information zur Session (optional), Medientitel

u=URI und weitere Beschreibung (optional)

e=E-Mail-Adresse (optional)

p=phone number, Telefonnummer (optional)

c=Connection Data, Verbindungsdaten (dieser Parameter ist nicht erforderlich,wenn diese Informationen bei der Beschreibung der einzelnen Medien enthalten ist) (optional).

b=bandwith, erforderliche Bandbreite (optional)

t=time, Zeit der Session-Aktivität

r=Repeat, optionale Wiederholungen (ein oder mehrfach)

z=xxx Zeitzohne (optional)

k=Key, Schlüssel zur gesicherten Übertragung (optional)

m=media keine, eine oder mehrere Medienbeschreibungen

a=attribute, eine oder mehrere Eigenschaften der Session

3.3.1 Medienbeschreibung

Der allgemeinen Session Description und dem genauen Zeitpunkt der Medienübertragung folgt je verwendeter Kommunikationsart eine eigene Medienbeschreibung. Eine Session-Description kann eine oder mehrere Media-Descriptions enthalten. Für eine Video-Kommunikation kann beispielsweise eine Medienbeschreibung die Audio- und eine zweite die Video-Eigenschaften beschreiben.

In der Antwort müssen die angebotenen Medien-Beschreibungen mit den unterstützten Eigenschaften der Gegenseite enthalten sein. Die Beschreibungen sind optional, unterstützt die andere Seite eine Medienart nicht, muss der Parameter zu dieser Medienart auch in der Antwort enthalten sein. Um zu kennzeichnen, dass diese Eigenschaft nicht unterstützt wird, wird der zugeordnete Port auf „0“ gesetzt. In der Antwort muss die Anzahl und die Reihenfolge der Medienbeschreibungen mit der in der Anfrage identisch sein.

Ein typisches Beispiel für eine einfache Verbindung enthält die folgenden Parameter:

m= [medienname, Port, Protokoll und Format]

i= [Medientitel, optional]

c= [connection information, Adressinformationen – optional, wenn nicht bereits im Session-Level definiert]

b= [Informationen zur benötigten Bandbreite – optional]

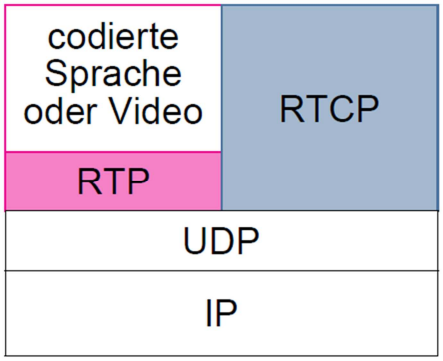
k= [Schlüssel zur Dekodierung von verschlüsselten Nutzdaten – optional]

a= [weitere Eigenschaften der Media-Beschreibung – optional]

3.4 Real-time Transport Protocol

Die Basis für die Sprach- und Videoübertragung im Internet ist das Real-time Transport Protocol (RTP, RFC 3550), das den Transport von Audio- und Videodaten in Paket-Form ermöglicht. RTP beinhaltet auch die Bereitstellung einer Dienstesynchronität. Zwischen Sender und Empfänger werden hierfür ständig Zeit- und Synchronisationsinformationen ausgetauscht. Paketüberholungen, wie sie in verbindungslosen Netzen immer vorkommen können, müssen vom Empfänger ausgeglichen werden. Für diese Aufgaben enthält jedes RTP-Paket eine Sequence Number (fortlaufende Durchnummerierung der einzelnen Pakete) und einen Timestamp (Zeitstempel). Weiterhin ist in dem RTP-Header eine eindeutige Identifikation des Senders und Empfängers enthalten. Zum RTP gehört auch das Real-time Control Protocol (RTCP), das die gleiche Adresse, aber einen anderen Port verwendet. RTP und RTCP verwenden UDP oder zukünftig auch DCCP als, siehe Abbildung 7. Prinzipiell kann auch TCP verwendet werden, hierbei ergeben sich allerdings Laufzeitprobleme durch ggf. vorhandene Wartezeiten auf Bestätigungen.

Abbildung 7: RTP und RTCP



3.4.1 Aufbau der RTP-Nachrichten

Der Transport der Sprachinformationen erfolgt oberhalb der IP-Ebene mit UDP, Empfangsbestätigungen sind nicht erforderlich. Der Aufbau einer RTP-Nachricht ist in Abbildung 8 dargestellt.

Die Elemente im RTP-Header haben die folgende Bedeutung:

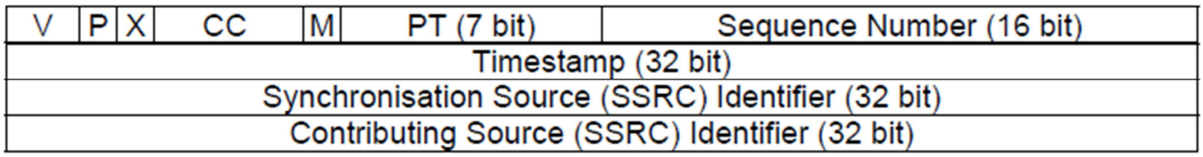
- Version (V): Im Versionsfeld wird die RTP-Version (z. B. 2, die gegenwärtige Version) übermittelt.
- Padding (P): Das einzelne Padding-Bit kennzeichnet ein oder mehrere Padding

Byte am Ende der Nutzübertragung. Wie groß der aufgefüllte Bereich ist, muss im letzten Byte des Padding-Bereichs am Ende des Nutzfeldes übertragen werden.

- Extension (X): Durch das Extension-Bit werden spezielle, ein Byte lange Erweiterungen, gekennzeichnet.
- CSRC Count (CC): Ob und wie viele Contributing Source Identifier im RTP-Header übermittelt werden, wird im CSRC Count übermittelt.
- Marker (M): Die Bedeutung des Marker-Bit ist vom jeweils verwendeten RTP-Profil abhängig. Es wird beispielsweise für die Unterstützung von Silence Suppression (Erkennung der Sprachpausen) verwendet. Das Bit wird in jedem ersten Paket mit Sprachproben nach einer vorangegangenen Sprachpause auf „1" gesetzt.
- Payload Type (PT): Das Feld Payload Type (PT) kennzeichnet die im Payload-Teil transportierten Nutzinformationen. Mit diesem Feld können die verschiedenen Quellcodierungen unterschieden werden. Für die Audio-/Video-Kommunikation (Audio/Video-Profile) sind dies beispielsweise die Sprach- (z. B. G.711, G.722, G.723 usw.) oder Video-Codecs (H.263, H.261 usw.), die auf der Empfängerseite für die Decodierung der Nutzinformationen notwendig sind. Weitergehende Festlegungen zur Interpretation der Nutzinformationen können durch definierte RTP-Profile festgelegt werden. Einige Standard-Profile (Static Payload Type Assignements) sind bereits vordefiniert, daneben können Festlegungen für die Interpretation der Nutzinformationen durch das Signalisierungsprotokoll (z. B. SIP) festgelegt werden.
- Sequence Number (SQ): Mit der Sequenznummer werden RTP-Pakete vom Sender durchnummeriert, wodurch Reihenfolgenfehler und der Verlust von Paketen für den Empfänger erkannt werden können.
- Timestamp (TStamp): Der Parameter beginnt mit einer zufällig ermittelten Zahl, die mit jeder Entnahme eines Srach- oder Video-Signals um eins erhöht wird.
- Synchronisation Source (SSRC): Eine eindeutige Zuordnung der Kommunikationspartner ist durch den Synchronisation Source Identifier (SSRC) und den Contributing Source Identifier (CSRC) gegeben.

Contributing Source Identifier (CSRC): Mit dem Contributing Source Identifier (CSRC) werden die verwendeten Quellen eines zusammengefassten Kommunikationsstroms gekennzeichnet.

Abbildung 8: ufbau einer RTP-Nachricht



V: Version (2 bit)
P: Padding (1 bit)
X: Extension (1 bit)
CC: CSCR Count (4 bit)
M: Marker (1 bit)
PT: Payload Type (7 bit)

FRAGE 5
In welchen Nachrichten muss ein Message-Body enthalten sein und welche Information ist darin enthalten?

3.4.2 Real-time Control Protocol (RTCP)

Aufgaben des RTCP
Das Real-time Control Protocol (RTCP, Bestandteil des RFC 3550 für RTP) steuert die eigentliche Nutzdatenübertragung (nicht zu verwechseln mit der Signalisierung) und gibt dem Sender der Nutzinformationen eine Rückkopplung zur Qualität der Übertragung. Der Informationsaustausch mit RTP basiert auf UDP und zukünftig, bei dem es keine Rückkopplung in Form von Quittungen gibt. Diese Quittungen ersetzen aber nicht die RTCP-Rückkopplungen. RTCP sendet regelmäßig Kontrollinformationen an die Teilnehmer einer Session. Durch das Protokoll RTCP kann dem Sender periodisch eine Rückkopplung über die erreichte Qualität der Übertragung (QoS-Feedback) gegeben werden. Beispiele für solche Steuerinformationen sind:

- Rückkopplung über die Qualität der Verbindung, diese Informationen können direkt die adaptiven Codecs steuern. Durch dieses Feedback können auch Fehler bei der Informationsverteilung erkannt werden.
- Identifikation des Senders durch sog. Canonical Names (CNAME), die eine eindeutige Kennzeichnung einer Verbindung auch bei einem Wechsel des SSRC ermöglichen.
- Anpassung der Senderate von RTCP-Informationen in Abhängigkeit von der Anzahl der Session-Teilnehmer, um eine zu große Netzbelastung zu verhindern.
- Übermittlung von Zusatzinformationen (optional) wie beispielsweise die Namen der Session-Teilnehmer an einer Konferenz.
- Übermittlung von Überlastzuständen auf der Seite des Empfängers.

Beispiele für die RTCP-Nachrichten (unterschieden im Payload Type Feld – PT-Feld) sind:

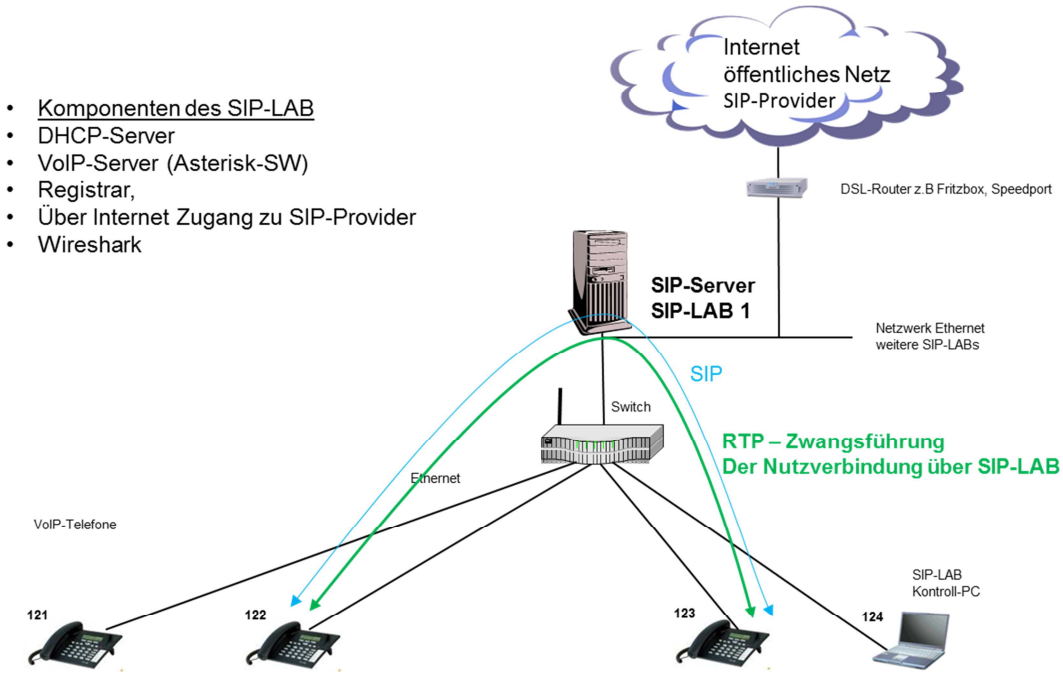
- Sender Report (SR, PT=200): Hiermit werden Informationen des Senders und über die Anzahl der gesendeten Pakete übertragen.
- Receiver Report (RR, PT=201): Diese Nachrichten übermitteln die Qualitätsinformationen für die Rückkopplungen zum Sender. In der Nachricht sind Informationen über die Anzahl verlorener Pakete, den Jitter sowie eine Zeitstempelung zur Berechnung der Umlaufverzögerung (Round Trip Delay) enthalten.
- BYE (PT=203): Ende einer Session-Teilnahme, mit dieser Nachricht wird die Trennung des Nutzkanals (z. B. von einer Konferenzeinrichtung, Mixer) signalisiert (nicht zu verwechseln mit der Beendigung des Session, BYE im SIP-Ablauf).

FRAGE 6
Über welches Protokoll der Transportschicht erfolgt die Übertragung des SIP-Protokolls und des RTP-Prptokolls (mit kurzer Begründung)?

4. Versuchsaufbau

Der Versuch wird mit Hilfe des SIP-LAB durchgeführt, dies ist als ein Demonstrations-System zum Thema VoIP und dem Session Initiation Protocol (SIP) konzipiert. Den Kern des Demo-Systems bildet ein SIP-Server (Open-Source-Software „Asterisk" auf Linux-Basis) mit allen notwendigen Hilfsanwendungen. Die Verbindungen zwischen den Terminals und dem Server werden durch einen Standard Router mit mehreren Ethernet-Schnittstellen und ggf. mit ungesicherten WLAN-Zugang realisiert. Für die Versuchsdurchführung sind zwei SIP-UA notwendig, die beispielswiese auf einem PC bzw. Notebook installiert sind. Nach Installation der UA-Software kann jedem Teilnehmer eine freie Rufnummer zugeordnet werden und der SIP-Server mit seiner festen IP-Adresse konfiguriert werden. Die gewählte Rufnummer muss dabei einer der konfigurierten Rufnummern (121 bis 139) entsprechen. Die PCs (bzw. Notebooks) können über Ethernet-Schnittstellen direkt am Router angeschlossen werden oder die Verbindung erfolgt über die WLAN-Schnittstelle (SSID: SIP-LAB, ohne Passwort-Sicherung).

Abbildung 9: Versuchsaufbau



FRAGE 7

Welcher Typ von SIP-Proxy-Server wurde in diesen Versuchen verwendet (mit kurzer Begründung)?

4.1 SIP-SERVER

Im SIP-LAB laufen alle zentralen Programme und Server für den Betrieb des autonomen VoIP-Systems. Neben dem zentralen SIP-Server sind Server für die Registrierung von SIP-Usern (der SIP-Registrar) automatische, dynamische IP-Adressenvergabe (Dynamic Host Configuration Protocol – DHCP), sowie einen Apache-Web-Server (Download-Bereich für die SIP-UA). Der Server hat die feste IP-Adresse z.B. 192.168.100.1 oder bei Versuchsanordnungen mit mehreren SIP-Telefonieservern 192.168.100.11 bis 192.168.100.14. Diese Adresse ist in allen SIP-User-Agents als SIP-Proxy-Adresse einzustellen.

4.2 Registrierung

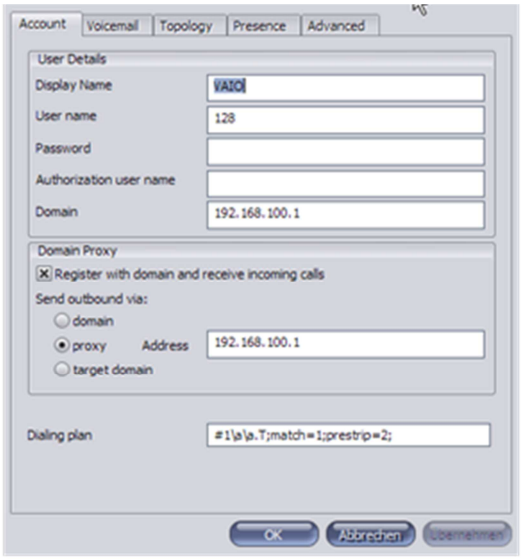
Mit der Anfrage „REGISTER" können sich die Terminals registrieren, d.h. die Zuordnung zwischen der dynamisch vergebenen IP-Adresse (durch DHCP) und der Rufnummer (z. B. 121) bzw. URL-Adresse (z.B. fon1@192.168.100.1). Im Demo-System erfolgt keine Authentifizierung der Teilnehmer, ein Passwort ist daher beim Einrichten des SIP-UA nicht erforderlich.

4.3 Konfiguration des SIP-User-Agent

Für die Versuchsdurchführung können beliebige SIP-User-Agents eingesetzt werden. In dem folgenden Abschnitt wird die Konfiguration des X-Lite User Agent dargestellt , siehe Abbildung 10. Andere SIP-UA müssen entsprechend konfiguriert werden, haben aber andere Benutzeroberflächen.

Unter dem Menüpunkt „SIP-Account, Einstellungen“ kann in der ersten Zeile ein frei gewählter Name vergeben werden. Dieser Name wird bei Anrufen beim Kommunikations-Partner angezeigt. Als „User-Name“ muss eine gültige Rufnummer oder URI vergeben werden (z. B. 121 bis 139). In der Rubrik „Domain“ und etwas tiefer in der Rubrik „Proxy Address“ muss die feste IP-Adresse des SIP-Servers eingegeben werden (192.168.100.1). Diese Einstellung wird mit „OK“ bestätigt und dann sollte alles funktionieren. Ein Neustart des Systems ist nicht erforderlich.

Abbildung 10: X-Lite User Agent

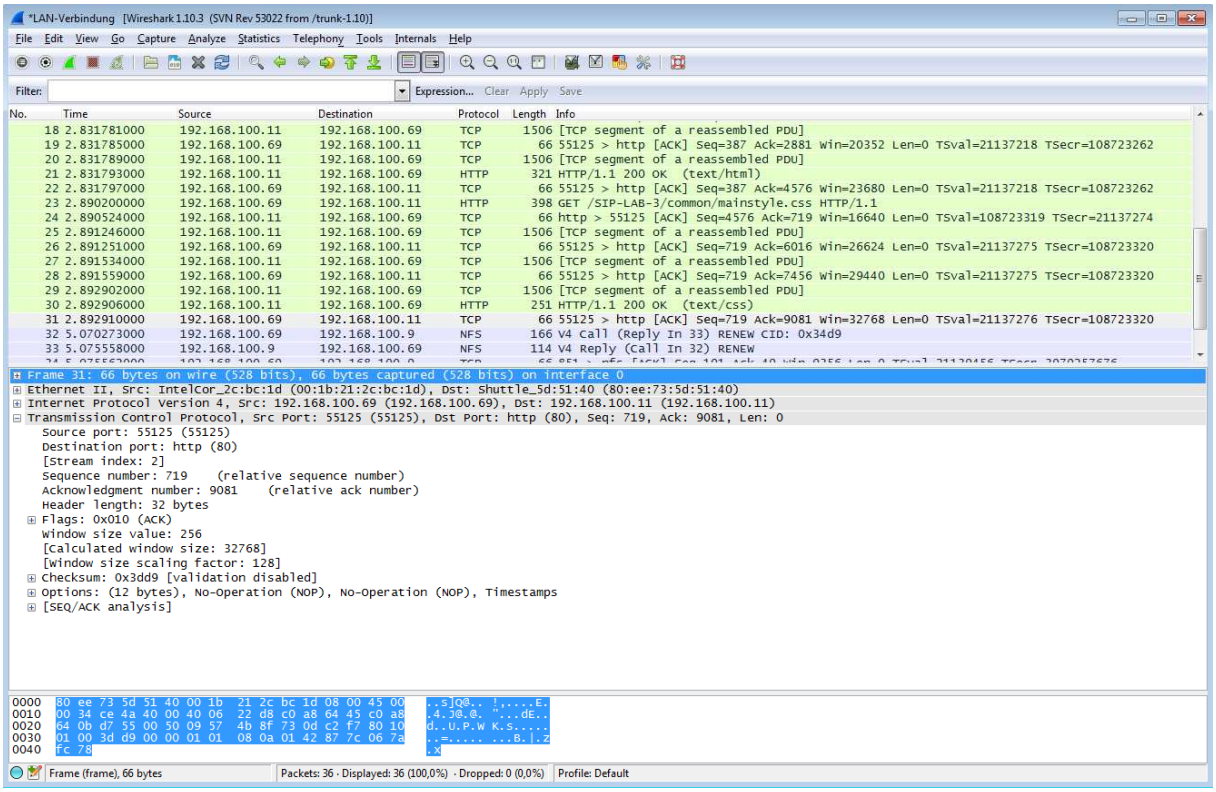


5. Wireshark

Mit dem Programm WireShark können Netzschnittstellen am Computer/Notebook des SIP-UA mitgelesen und die transportierten Nachrichten interpretiert werden. Wireshark ist eine frei verfügbare und einsetzbare Software zum Mitlesen und zur Analyse von Rechner-Netzen. Alle Pakete werden mit einem Zeitstempel versehen und in einer Zusammenfassung ihres Aufbaus dargestellt , siehe Abbildung 11. Die meisten Inhalte können dann im zweiten Fenster noch weiter detailliert werden im dritten Fenster wird die komplette Nachricht noch in Hex-Form dargestellt.

Die mitgelesenen Nachrichten können in einem eigenen Format gespeichert werden und später (auch auf einer anderen Plattform) ausgewertet werden.

Abbildung 11: Wireshark Darstellung



5.1 Aufzeichnung der Abläufe mit WireShark

Abbildung 12: Wireshark Einstellungen 1

Die Aktivitäten am Asterisk-Server können nun mit Wireshark (Ethereal) aufgezeichnet werden. Vor Beginn der Aufzeichnungen muss die Ethernet-Schnittstelle selektiert werden, siehe Abbildung 12. Ein Notebook mit einer Ethernet-Schnittstelle und einem Schnittstelle festgelegt werden.

Im Unterpunkt „Optionen“ können Einstellungen für das Mitlesen WLAN-Zugang hat dabei bereits zwei mögliche Interfaces. Unter „Capture“ und „Interfaces“ kann die

der Nachrichten eingestellt werden. Hier muss (normalerweise) nichts verändert werden, die Einstellungen sollten so aussehen:

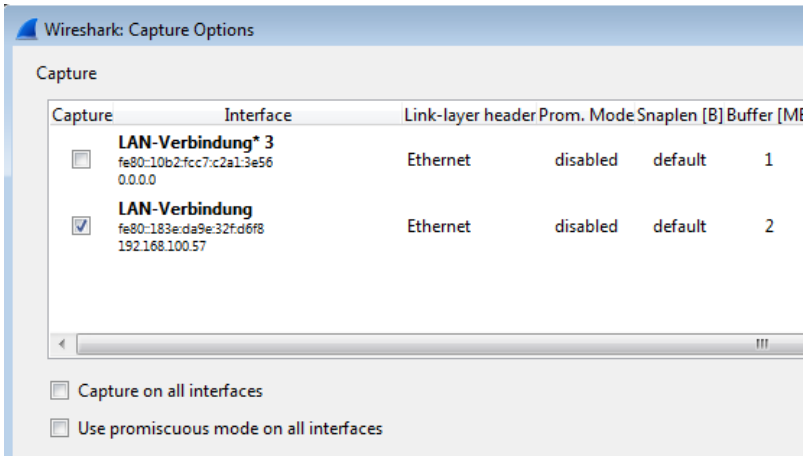
Abbildung 13: Wireshark Einstellungen 2

Mit den „Display Options“ „Update list of packets in real time“ und „Automatic scrolling in live capture“ können die gelesenen Pakete direkt angezeigt werden, Abbildung 13 zeigt beispielhaft die Einstellung an einem Kontroll-PC.. Das automatische „scrollen“ erlaubt den Blick auf die aktuell eingelesenen Pakete. Zur genaueren Auswertung sollte dann die Aufzeichnung mit der Auswahl „Capture“ und „Stopp“ angehalten werden.

5.2 Alternative Optionen

Die Option „Capture packets in promiscuos mode“ erlaubt Wireshark alle Pakete im LAN zu lesen, siehe Abbildung 14. Ist diese Option nicht gewählt können nur die Pakete vom und zum eigenen Computer gelesen werden. Hinweis: In manchen Fällen funktioniert Wireshark nicht im promiscuos-mode. Sollte nach der richtigen Wahl der Ethernet-Schnittstelle und dem Start des Mitlesens keine Pakete gelesen werden, ist die Option „Capture packets in promiscuos mode“ abzuschalten.

Abbildung 14: Wireshark Einstellungen 3



Wird die Option „Update list of packets in real time“ nicht gewählt, werden alle Paket nach dem Start gelesen und gespeichert, aber nicht angezeigt. In diesem Fall sollte auch die Option „Hide capture info dialog“ abgeschaltet sein, dann erhält man während des Mitlesens einige Informationen zu den gelesenen Paketen, siehe Abbildung 15:

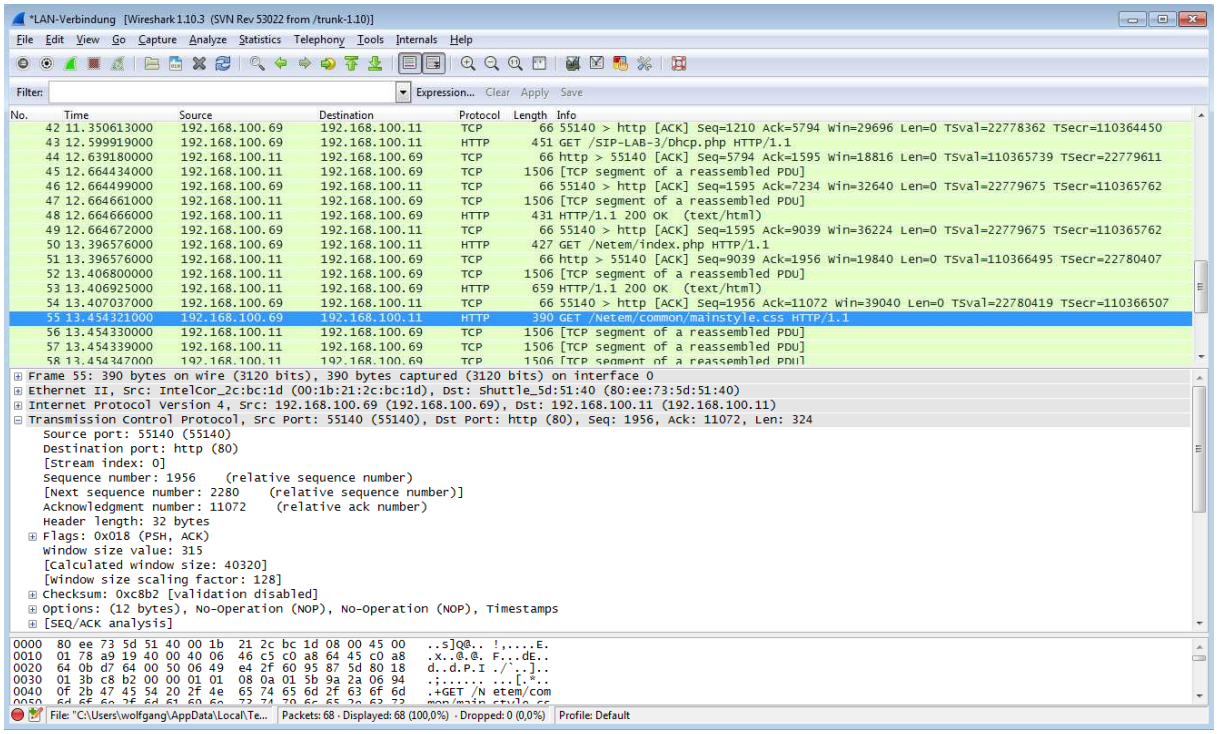
Abbildung 15: Wireshark Captured Packets

FRAGE 8

Was ist der „promiscuos mode“ und wann kommt er zum Tragen?

Mit „Stopp“ hält man die Aufzeichnung an und Wireshark stellt alle aufgezeichneten Nachrichten in einem Fenster dar, siehe Abbildung 16:

Abbildung 16: Wireshark Anzeige 2



FRAGE 9

Unterscheidet sich der Testaufbau in Bezug auf die Architektur von realen VoIP-Netzarchitekturen (mit kurzer Begründung)?

6. Versuche

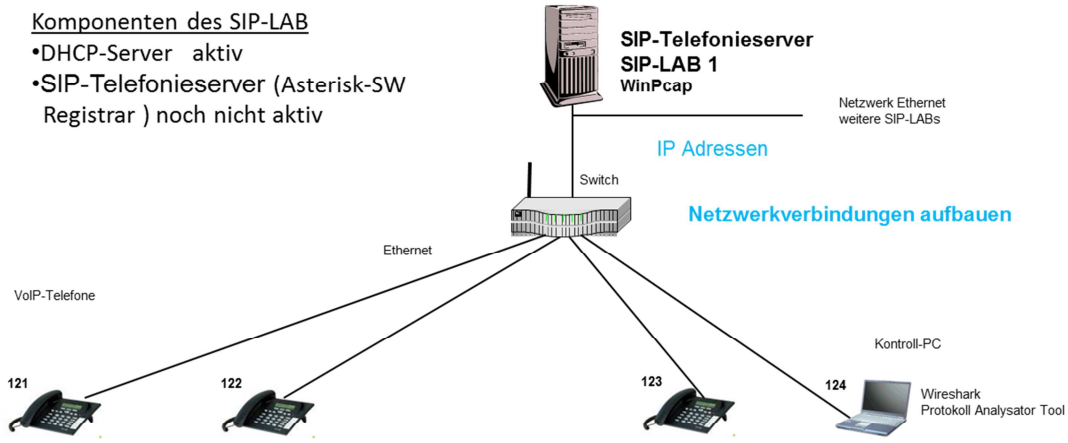
6.1 Versuch 1 Netzwerkinbetriebnahme SIP-LAB

6.1.1 Zielsetzungen

Die Zielsetzung dieses Versuchs ist Inbetriebnahmen und Kennenlernen des SIP-LABs zusammen mit dem Rest der Versuchsanordnung.

6.1.2 Vorbereitung des Versuchs

Abbildung 17: Versuchsaufbau



Sie schalten das SIP-LAB ein (sofern noch nicht geschehen). Der DHCP-Server im SIP-LAB-3/1 ist in Betrieb, das ist Grundlage für die Netzwerkfunktion.

Sie verbinden Telefone und Kontroll-PC mit dem Switch und schalten alle Geräte ein.

Netzwerkeinstellungen des Kontroll-PCs muss auf „IP-Adresse automatisch beziehen" eingestellt sein.

SIP-LAB Statusabfrage Zugriff vom Kontroll-PC auf die Startseite des SIP-LABs:
http://192.168.100.xx/SIP-LAB-3, wobei xx=11, … xx=14
für SIP-LAB-3-1 … SIP-LAB-3-4 ist.

Der SIP-Telefonieserver ist noch nicht aktiv.

6.1.3 Aufgabenstellung

Ermitteln Sie die Netzwerkeinstellungen des Kontroll-PCs (Windows XP, Windows 7 oder Windows 8) und tragen Sie die ermittelten Werte in die Tabelle 1 ein:

- IP-Adresse und Netzwerkmaske
- Standard Gateway Adresse
- Adresse Domain Name Server (DNS)

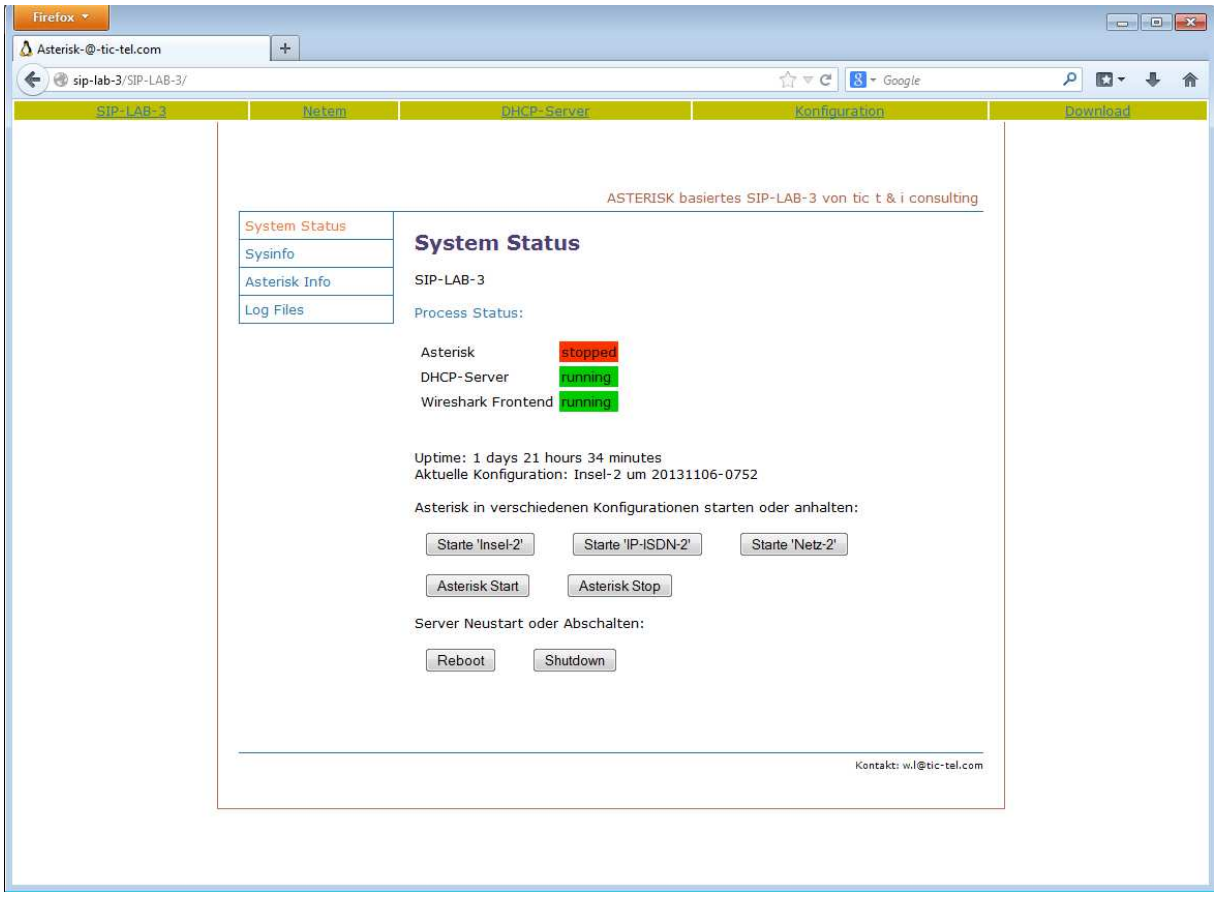
Ermitteln Sie die Netzwerkeinstellungen der Telefoneinstellungen. Als Hilfsmittel steht Ihnen die Telefontastatur zur Verfügung. Gibt es weitere Hilfsmittel?

Zum Abschluss ermitteln Sie mittels „ping" die Responsezeiten ausgehend vom Kontroll-PC zum SIP-LAB, zu den Telefonen, Gateway, beliebige Internet URL

Tabelle 1

Gerät	IP-Adresse	Netzmaske	DHCP-Server	Standardgateway	DNS	Ping (ms)
SIP-LAB						
Telefon 1						
Telefon 2						
Telefon 3						
Kontroll-PC						
URL						

Abbildung 18: Screenshot Bedienseite des SIP-LAB



6.1.4 Abschluss des Versuchs

Im Laborbericht dokumentieren: die Werte in Tabelle 1 eintragen

6.2 Versuch 2 Wireshark Einführung mit Ping

6.2.1 Zielsetzungen

Was ist Wireshark? Wireshark ist ein freies Protokollanalysetool, wahrscheinlich das am weitesten verbreitete. Die Möglichkeiten sind schier unermesslich und die Benutzung dieses Tools nicht gerade einfach. Allen Aufgaben in diesem SIP-Labor liegt die Wireshark-Version 1.10.3 zugrunde, falls noch andere Versionen zum Einsatz kommen sollten, gibt es möglicherweise Abweichungen in Funktionalität und Erscheinungsbild.

Das eigentliche Ziel dieses Versuchs ist Protokollanalyse einzuüben auf verschiedenen Netzwerkschnittstellen mit einfachen Ping-Testsequenzen.

6.2.2 Vorbereitung des Versuchs

Falls Wireshark noch auf einem Kontroll-PC installiert werden müsste ist das bei den gängigen Windows Versionen unproblematisch.

Die Installationsdatei für Wireshark 1.10.3 für 32 und 64 bit und das User Manual liegen auf den SIP-LABs unter http://192.168.100.xx/Download zum Download bereit, wobei xx=11, … xx=14 für SIP-LAB-3-1 … SIP-LAB-3-4 ist.

Nach der Installation kann Wireshark gestartet werden. Erscheinungsbild wie im einleitenden Text als Abbildung 12, 11 und 16 dargestellt.

6.2.3 Aufgabenstellung

Auf einer lokalen Netzwerkschnittstelle („LAN-Verbindung“ im Abbildung 19) soll ein Protokollmitschnitt durchgeführt und analysiert werden. Als Protokollablauf ist eine Ping-Sequenz zu einem beliebigen Ziel einzustellen.

Über „Interface List“ kann die gewünschte Netzwerkschnittstelle ausgewählt werden, während über „Capture Options“ zusätzlich eine Reihe von Parametern eingestellt werden kann.

Abbildung 19: Wireshark Screenshot Auswahl der Schnittstelle

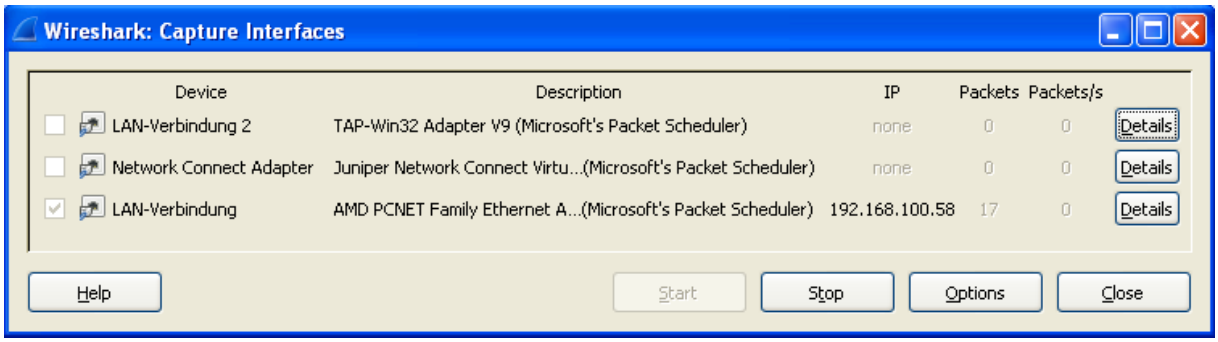
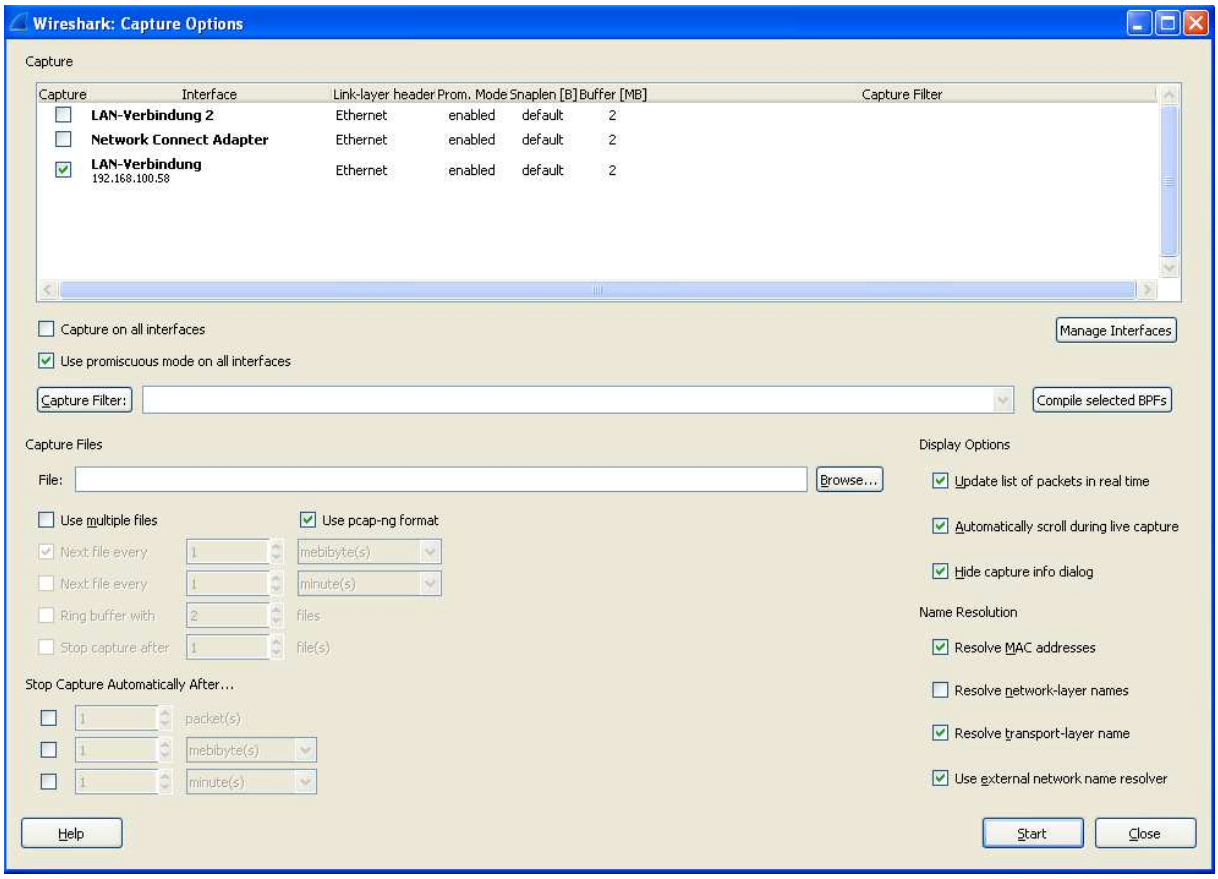


Abbildung 20: Wireshark Screenshot Wireshark Options

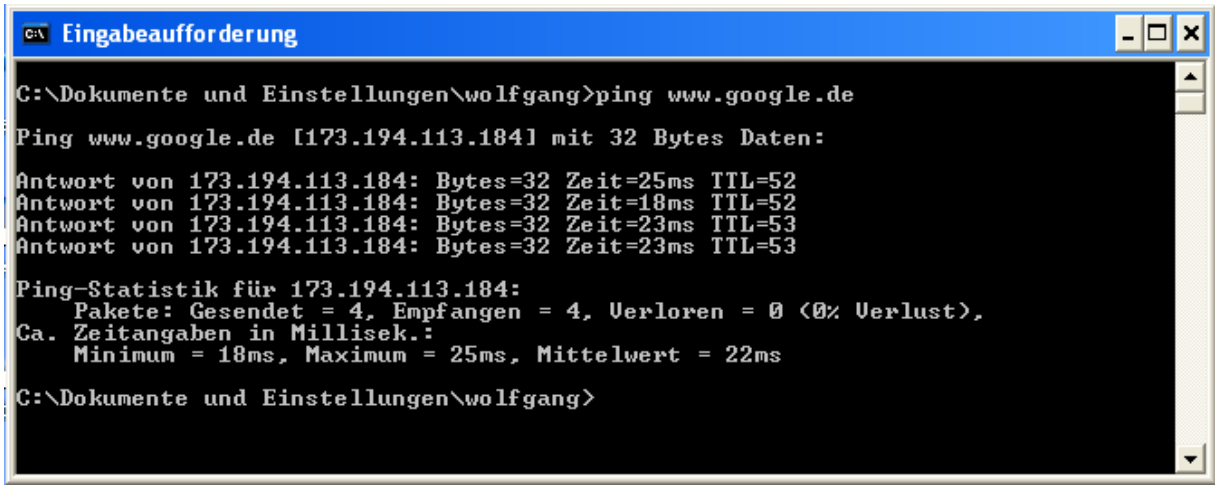


Wichtig sind die folgenden Einstellungen:

- Update list of packets in real time
- Automatically scroll during live capture

Danach “Start” und weiter mit „Ping“ starten

Abbildung 21: „Ping“ starten



6.2.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

- Was ist zu beobachten und kann der Gesamtprotokollmitschnitt beurteilt werden?
- Können Sie noch typische Nachrichten erkennen?
- Was ist zu tun um das Ping-Ergebnis auswerten zu können?
- Wie lautet das Filter um eine Ping-Sequenz isoliert darzustellen?
- Kann hiermit eine Zeitmessung durchgeführt werden?

6.3 Versuch 3 Registrierung und Basic Call

6.3.1 Zielsetzungen

Die Zielsetzung dieses Versuchs ist, im ersten Schritt den Ablauf der Registrierung am SIP- Telefonieserver nach dem Einschalten eines SIP-Telefons zu protokollieren und in einem zweiten Schritt den Ablauf eines Anrufs (Basis Call) zu protokollieren.

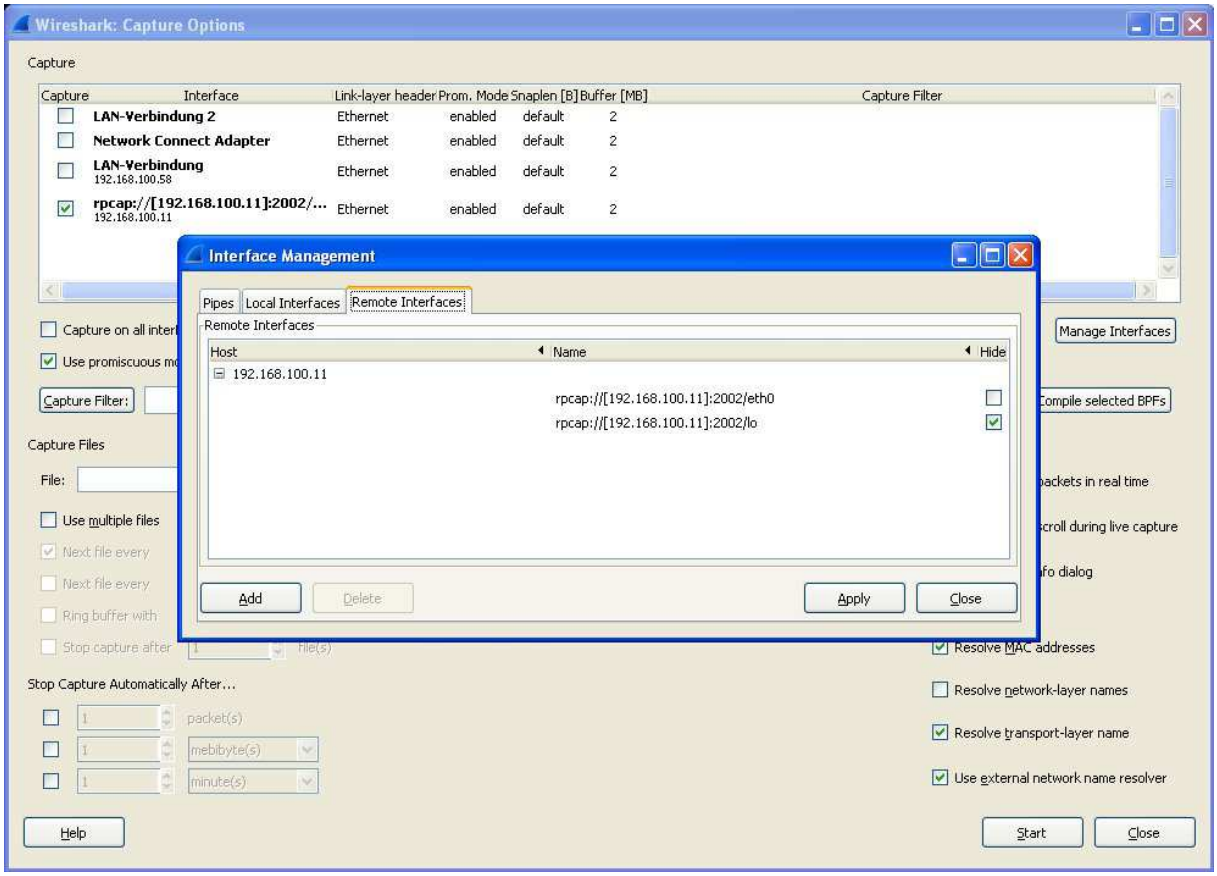
6.3.2 Vorbereitung des Versuchs

Für die Durchführung dieses Versuchs ist ein aktiver Telefonieserver erforderlich. Starten Sie hierzu den Asterisk Server in der Konfiguration „Insel-2“.

Siehe hierzu Abbildung 18 aus Versuch 1.

Meistens wird Wireshark wie in Versuch 2 verwendet, d.h. der Protokollmitschnitt erfolgt an einer lokalen Schnittstelle am PC, siehe Abbildung 20. Bei diesem Versuch soll die Netzwerkschnittstelle des SIP-LABs untersucht werden. Die zu protokollierende Schnittstelle ist in diesem Versuch daher ein „Remote Interface“, das sich am SIP-LAB-3 befindet. Siehe hierzu den folgenden Screenshot

Abbildung 22: Wireshark-Einstellungen der entfernten Netzwerkschnittstelle



Zu diesen Einstellungen kommt man über
Capture > Options > Manage Interfaces und dann Remote interfaces

6.3.3 Aufgabenstellung

Starten Sie Wireshark und schalten Sie anschließend zwei Telefone ein bzw. aktivieren Sie die Accounts.

Zeichen Sie alle SIP-Nachrichten für die Registrierung und anschließend einen einfachen Verbindungsaufbau zwischen zwei Terminals (A und B) mit Wireshark auf. Aus der Aufzeichnung sollen die folgenden Parameter ermittelt und Fragen beantwortet werden.

1) Welche IP-Adressen wurden den Terminals vergeben?

- Terminal A, IP-Adr.:
- Terminal B, IP-Adr.:

2) SIP verwendet einen bestimmten UDP/TCP-Port zum Austausch der Nachrichten, welcher Port wird zwischen Terminal A und Server sowie zwischen Server und Terminal B verwendet? Tragen Sie die verwendeten Ports in das folgende Übersichtsbild ein.

3) Der Nutzkanal wird durch einen festgelegten Port im UDP festgelegt. Welcher Port wird für die Kommunikation zwischen A-Server und B in beiden Richtungen verwendet?

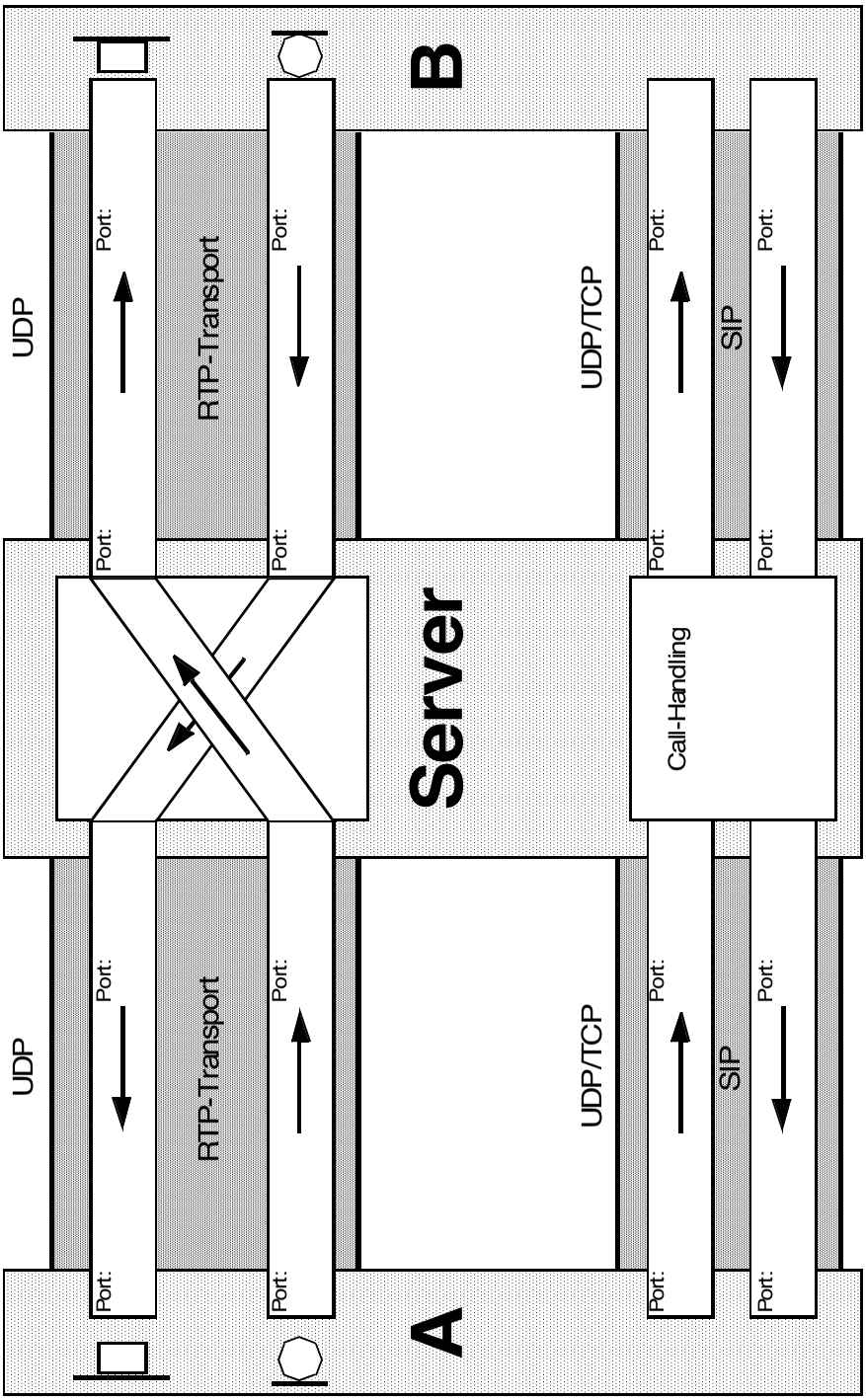
4) Wann und in welcher SIP-Nachricht wurden diese Ports festgelegt? Welche Nachricht an welcher Stelle?

6.3.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Tragen Sie die verwendeten Ports in Abbildung 23 ein.

Abbildung 23: Übersichtsbild zum Basic-Call



6.4 Versuch 4 Netem Einführung

6.4.1 Zielsetzungen

Das Ziel dieses Versuchs ist mit dem Netzwerktool Netem anhand einer einfachen Einstellung vertraut zu werden und mit dieser einfachen Einstellung den Einfluss auf die Sprachkommunikation zwischen zwei Personen zu demonstrieren.

6.4.2 Vorbereitung des Versuchs

Die Voreinstellung des SIP-LABs ist so gewählt, dass der RTP-Strom zwangsweise über das SIP-LAB geführt wird. Damit besteht die Möglichkeit die Pakete zu beeinflussen. Abbildung 25 zeigt die Beispieleinstellung für eine Verbindungsbeeinflussung mit einem Delay von 500 ms.

Abbildung 24: Einstellungen Netem Schritt 1

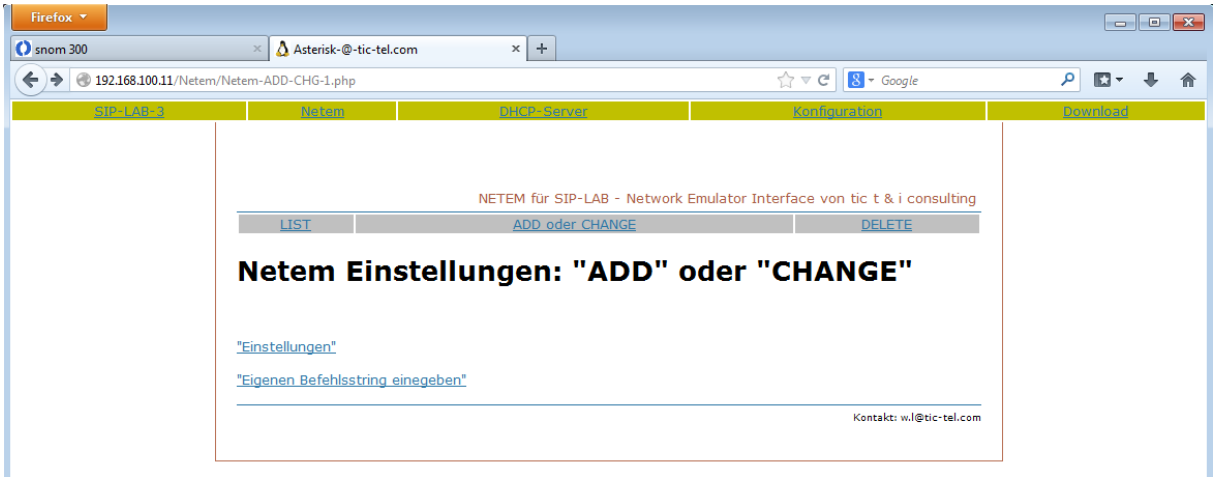
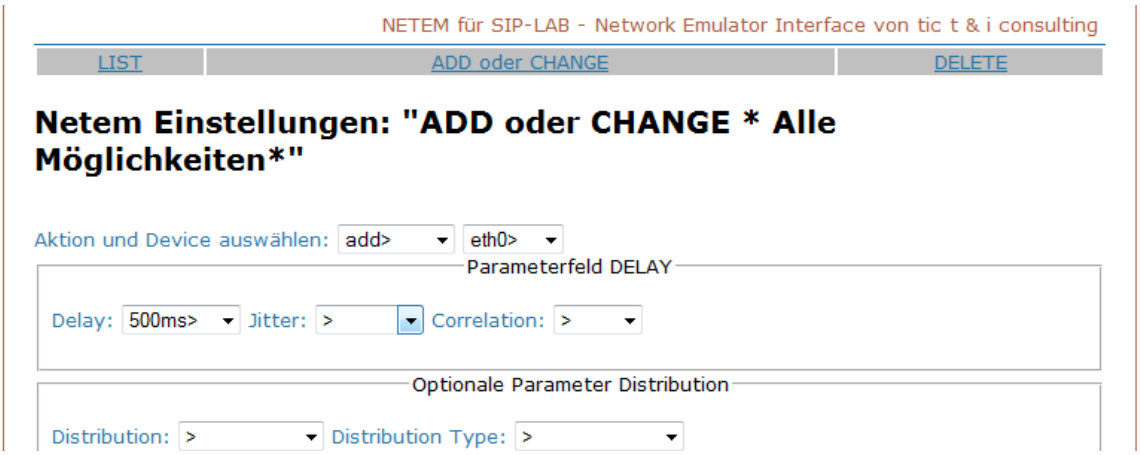


Abbildung 25: Einstellungen Netem Schritt 2 (Auszug)



Zum Schluss „Neue Konfiguration übernehmen" anklicken

6.4.3 Aufgabenstellung

Großes Delay wie im Abschnitt „Vorbereitungen“ beschrieben einstellen und Einfluss auf den Gesprächsfluss untersuchen.

Fragen:

- Mit diesem einfachen Versuch soll der Einfluss auf den Gesprächsfluss beurteilt werden. Wie ist der Gesprächsverlauf bei diesem Delay?
- Was ist die Schlussfolgerung daraus für Sprachkommunikation über Satellitenverbindungen?

FRAGE 10:

Wie groß ist Laufzeit mindestens in eine Richtung bei einer Satellitenverbindung über einen geostationären Satelliten und idealen Ausbreitungsbedingungen?

6.4.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Subjektive Bewertung einer Verbindung mit einer großen Laufzeit:

Ihre Erkenntnisse

6.5 Versuch 5 Codecs und Verständlichkeit

6.5.1 Zielsetzungen

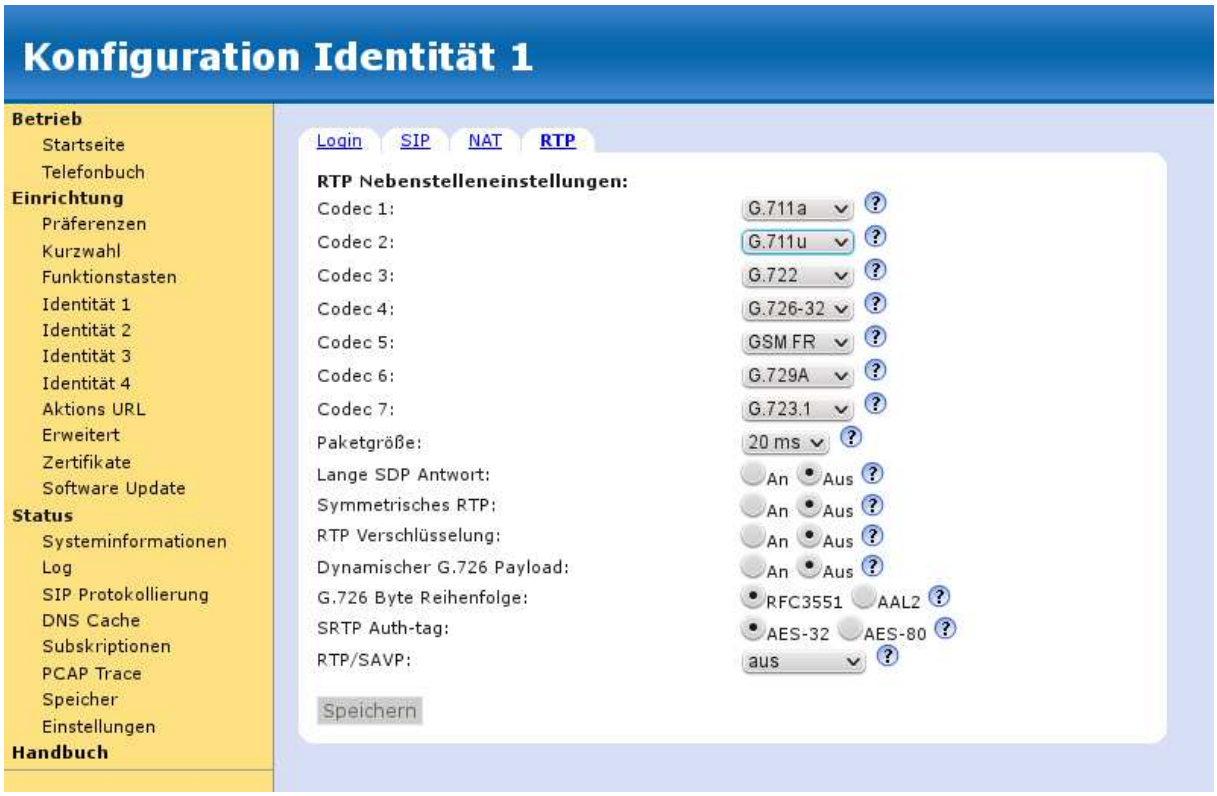
Das Ziel dieses Versuchs ist es zu demonstrieren, dass die Verständlichkeit abhängig ist von der Auswahl des Sprachcodecs.

6.5.2 Vorbereitung des Versuchs

Die Tests erfolgen mit einem Basic Call zwischen zwei Telefonen.

An den Telefonen werden alle verfügbaren Codecs freigeben. Abbildung 26 zeigt ein Beispiel des Telefons Snom 300 in der dargestellten Prioritätsreihenfolge.

Abbildung 26: Einstellungen Beispiel Snom 300.



Im SIP-LAB nur die für den Versuch ausgewählten Codecs freigeben.
Abbildung 27 zeigt die Einstellungen am Beispiel Teilnehmer mit der Rufnummer 121 in der Konfiguration „Insel-2“.

Bisherige Codec Einstellungen: alaw, ulaw, gsm;

Neue Codec Einstellungen: nur alaw.

Nach der Auswahl Einstellungen abschließen.

Abbildung 27: Einstellungen SIP-LAB-3

Sie finden diesen Parameter wie folgt:

http://<IP-Adresse>/SIP-LAB-3 > Konfiguration > Teilnehmer Grundkonfiguration

Konfiguration „Insel-2" > Schritt 1 > Teilnehmernummer auswählen > Schritt 2

6.5.3 Aufgabenstellung

1) Testen Sie die Verständlichkeit der Verbindung unter Verwendung verschiedener Codecs.

2) Müssen die Codecs auf beiden Seiten identisch sein oder kann das eine Terminal (A) den Codec G.711 A verwenden und das andere Terminal (B) den GSM-Codec?

6.5.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Zu 1. Bewertung der Verständlichkeit der Verbindungen für jeden ausgewählten Codec

Zu 2. Beantwortung der Frage, falls nein Begründung warum

Zeichnen Sie ein SIP- und RTP-Protokoll mit Wireshark auf und ermitteln Sie die Zeiten für die Pakete

(Falls RTP nicht erkannt wird:
Go to Preferences > Protocols > RTP, check the box "Try to decode RTP outside of conversations".)

6.6 Versuch 6 Realitätsnaher Verbindungsaufbau mit kompatiblen und inkompatiblen Codes

6.6.1 Zielsetzungen

Die Zielsetzung dieses Versuchs ist, den Ablauf eines Anrufs realitätsnah für zwei verschiedene Fälle zu protokollieren. Und zwar:

- im Fall von compatiblen Codes in beiden Telefonen, und
- im Fall von imcompatiblen Codes in beiden Telefonen

Realitätsnah heißt, dass die RTP-Streams nicht zwangsweise über den SIP-Telefonieserver geführt werden.

6.6.2 Vorbereitung des Versuchs

Für die Durchführung dieses Versuchs müssen die Einstellungen im SIP-Telefonieserver so verändert werden, dass wie oben erwähnt die RTP-Streams nicht zwangsweise über den SIP-Telefonieserver geführt werden. Das geschieht in der SIP-Telefonieserver SIP-LAB mittels eines teilnehmerspezifischen Parameters „canreinvite“, der von der Standardeinstellung „no“ auf „yes“ verändert wird. Sie finden diesen Parameter wie folgt:

http://<IP-Adresse SIP-LAB>/SIP-LAB-3 > Teilnehmer Grundkonfiguration Konfiguration „Insel-2“ > Schritt 1 > Teilnehmernummer auswählen > Schritt 2

6.6.3 Aufgabenstellung

Im Gegensatz zum letzten Versuch sollen jetzt im SIP-LAB je max. 3 Codecs freigegeben werden und in den Telefonen werden je 1, aber inkompatible Codecs ausgewählt.

Konstruieren Sie eine geschickte Auswahl für kompatible und inkompatible Codecs und nehmen Sie mit Wireshark die Protokollabläufe auf.

6.6.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Skizzieren Sie die beiden Abläufe in den folgenden zwei Abbildungen und erläutern Sie die Unterschiede.

Stellen Sie zum nach Abschluss des Versuchs den Parameter „canreinvite“ wieder in die Standardeinstellung „no“.

Abbildung 28: Scenario compatible Codes

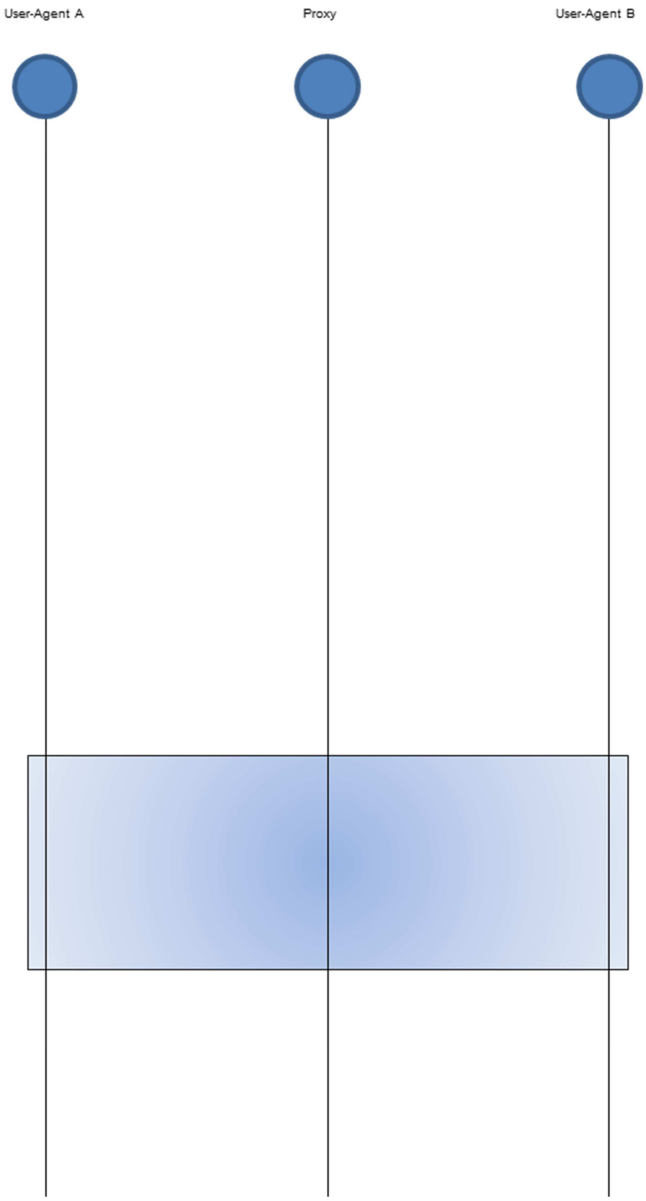
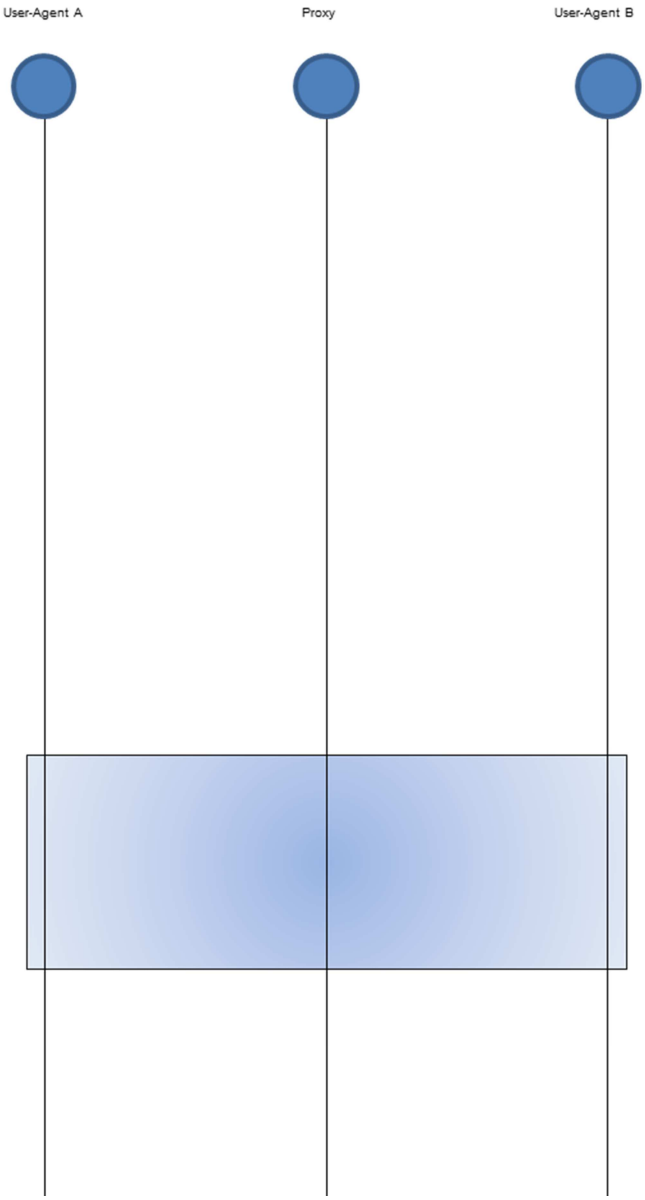


Abbildung 29: Scenario incompatible Codes



6.7 Versuch 7 Rufumleitung durch User Agent (UA)

6.7.1 Zielsetzungen

Das Ziel dieses Versuchs ist eine Möglichkeit einer Rufumleitung durch einen UA darzustellen.

6.7.2 Vorbereitung des Versuchs

Falls der UA X-Lite auf einem Kontroll-PC noch installiert werden muss ist das bei den gängigen Windows Versionen unproblematisch.

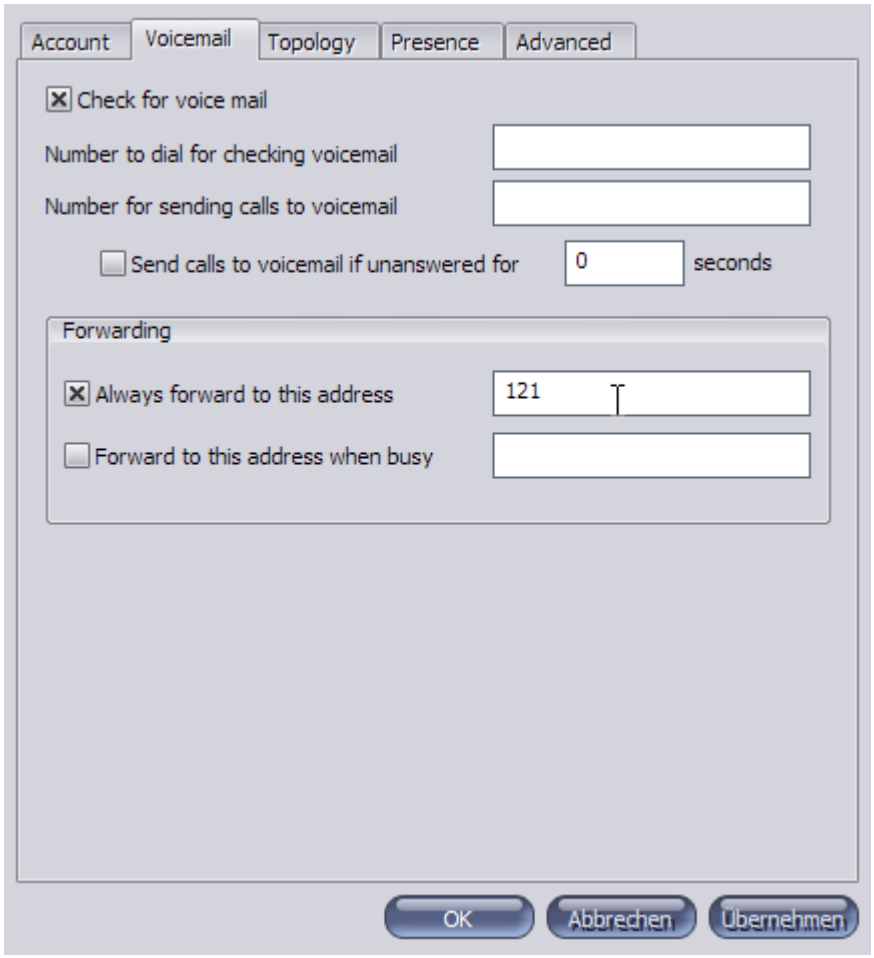
Die UA-Installationsdatei (X-Lite_Win32_1003l_30942.exe) steht auf den SIP-LABs für Windows zum Download bereit. Die Adressen sind: http://192.168.111.xx, wobei xx=11, … xx=14 für SIP-LAB-3-1 … SIP-LAB-3-4 ist.

6.7.3 Aufgabenstellung

Richten Sie eine Rufumleitung am UA ein und zeichnen Sie den Nachrichtenablauf mit Wireshark auf. Stellen Sie den Ablauf in einem Diagramm (Szenario, mit den Terminals, Netzelementen und den ausgetauschten Nachrichten) dar.

Beim XLite findet man die Rufumleitung unter den „SIP-Accout-Settings“ und hier unter „Voicemail“ dann kann man bei Forwarding die Zieladresse eingeben, zu der alle Anrufe umgeleitet werden sollen.

Abbildung 30: Einstellung der Rufumleitung im UA X-Lite (User Agent B)

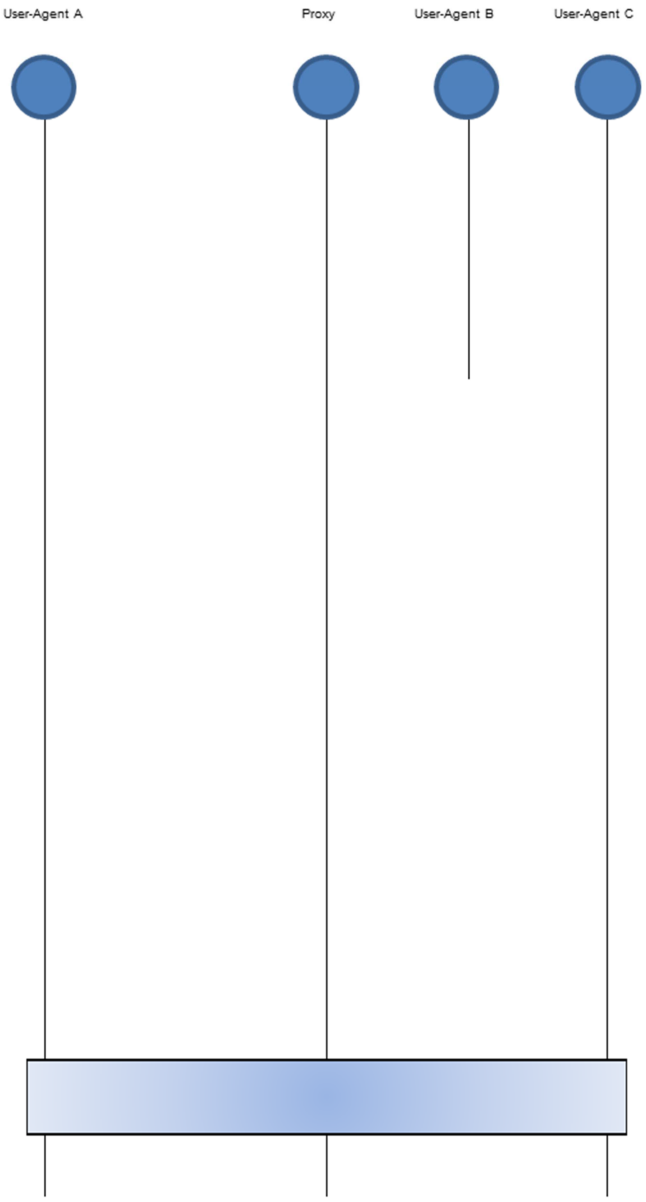


6.7.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Protokolle SIP und RTP mit Wireshark erfassen und Protokollsequenz in Diagramm eintragen.

Abbildung 31:Ablauf Rufumleitung



6.8 Versuch 8 Simulation großer IP-Netze

6.8.1 Zielsetzungen

Das Ziel dieses Versuchs ist mit Hilfe der Testumgebung zu demonstrieren, dass das Netzwerk, das ja im allgemeinen Fall das öffentliche Internet ist, einen großen Einfluss auf die Sprachqualität haben kann.

6.8.2 Vorbereitung des Versuchs

Das SIP-LAB wird als Insel-Konfiguration gestartet. Die Tests werden von den angeschlossenen Telefonen ausgeführt, entweder als Echo-Verbindung im SIP-LAB oder zu einem Tongenerator im SIP-LAB.

6.8.3 Aufgabenstellung

Testen Sie die Veränderung der Netzeigenschaften durch den Netzemulator Netem: Tests mit veränderten Parametern für Delay, Delay-Variation, Distribution, Packet-Reordering, Packet-Loss, Packet-Retransmission.

Tabelle 2: Kennwerte verschiedener Verhältnisse

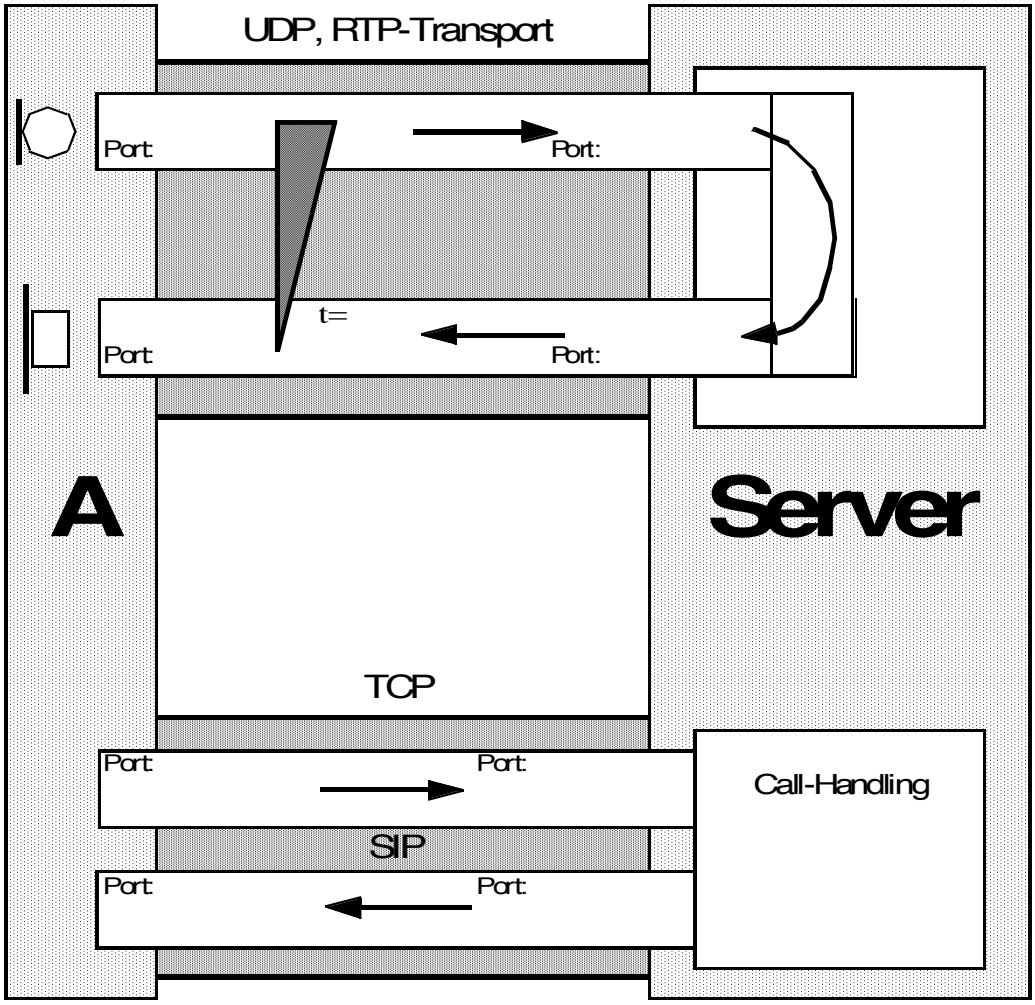
Parameter	**Ohne Netem**	**Gute Bedingungen**	**Durch-schnittliches Netz**	**Sehr schwierige Verhältnisse**
Delay		100ms	200ms	400ms
Delay-Varianz		10ms	40ms	100ms
Verteilung (Delay)		Normal	Pareto-Normal	Pareto-Normal
Reordert		5%	10% mit 20% Cor.	20% mit 50% Cor.
Verlust		0,5%	2%	5%
Duplizierung		-	0,1%	0,2%

Das Einrichten der Bedingungen erfolgt mit dem Netzemulator Netem.

1) Bauen Sie vom Endgerät A aus eine Echo-Verbindung (Rufnummer 175) auf und ermitteln Sie zeitliche Verhalten der RTP-Pakete anhand der Wireshark-Aufzeichnung.

2) Überprüfen Sie die Auswirkung durch Codec-Wahl auf die Laufzeiten.

Abbildung 32: Portzuordnungen



6.8.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Zu 1) Laufzeiten und Jitter

Zu 2) Portauswahl ebenfalls eintragen

6.9 Versuch 9 Untersuchung verschiedener Telefone bei gestörter Übertragung

6.9.1 Zielsetzungen

Das Ziel dieses Versuchs ist es zu demonstrieren, dass das IP-Telefon ebenfalls einen großen Einfluss auf die Sprachqualität habe kann.

6.9.2 Vorbereitung des Versuchs

Das SIP-LAB wird als Insel-Konfiguration gestartet. Die Tests werden von den angeschlossenen Telefonen ausgeführt entweder als Echo-Verbindung im SIP-LAB oder zu einem Tongenerator im SIP-LAB.

Damit Messungen an den analogen Ausgangssignalen mit einem Oszilloskop durchgeführt werden können wird für diesen Versuch ein Adapterkabel zur Verfügung gestellt, das zwischen Höreranschlusskabel und Hörer eingefügt wird. Damit stehen für die Messung sowohl die Hörer- als auch die Mikrofonanschlüsse für die Messung zur Verfügung.

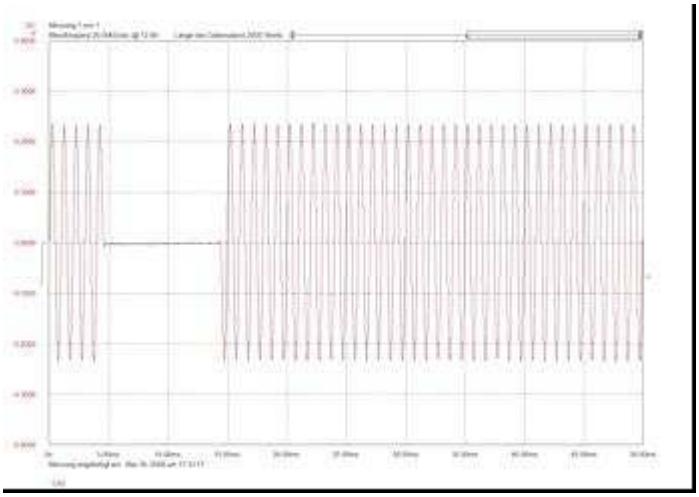
6.9.3 Aufgabenstellung

Mit der Rufnummer 176 können Standard-Töne und Ansagen vom SIP-LAB abgerufen werden und durch Wahl der Rufnummer 177 wird kontinuierlich ein Test-Ton mit einer Frequenz von 1004 Hz mit einem Pegel von 0 dBm0 PCM µ-Law gesendet. Mit diesem Test-Ton kann die Qualität des Decoders beim Empfänger des Terminals geprüft werden. Je nach gewählten Codec und dessen Realisierung haben vorgegebene Verzögerungen, Schwankungen dieser Verzögerungen und vor allem Paketverluste (eingestellt mit Netem) unterschiedliche Auswirkungen.

Die Auswirkungen der Netem-Einstellungen können mit dem Oszilloskop am mit Hilfe des Adapterkabels aufgezeichnet werden. Wie sich die Auswirkungen am Endgerät darstellen hängt maßgeblich von den verwendeten Korrekturmaßnahmen des Terminals ab.

a) Ohne korrigierende Maßnahmen auf der Seite des Empfängers bleibt eine deutliche Lücke, die von der festgelegten RTP-Größe (in diesem Beispiel 10 ms) abhängt.

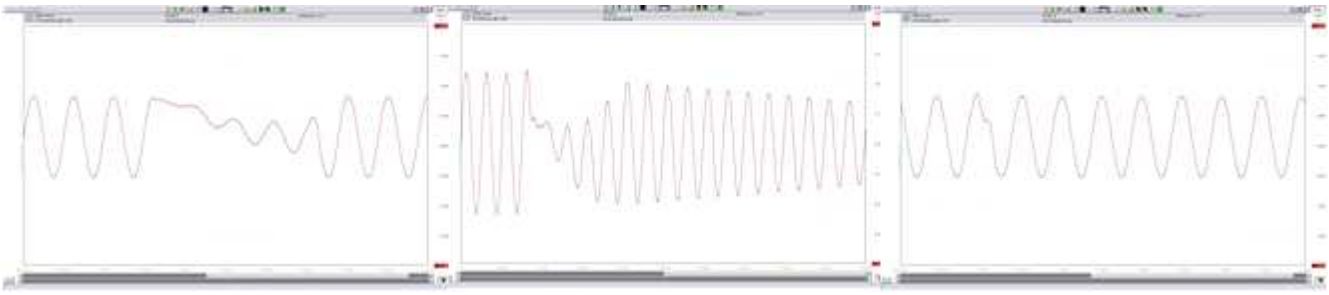
Abbildung 33: Analoges Ausgangssignal „Lücke“



b) Mit Korrektur hängt es von der Art den getroffenen Maßnahmen und deren Realisierung im Endgerät ab, wie sich ein RTP-Paketverlust auswirkt. In der folgenden Abbildung sind Beispiele für korrigierte Paketverluste dargestellt. Die Bedingungen des Netztransports, die Verzögerungen, deren

Schwankungen und der Paketverlust ist in allen Beispielen gleich. Die Bilder wurden aber an unterschiedlichen Endgeräten mit unterschiedlichen Codecs aufgezeichnet.

Abbildung 34: Analoges Ausgangssignal „Lücke, kompensiert“



Aufgabenstellung:

Versuchen Sie ähnliche Bilder mit den unter Versuch 8 gegebenen Netem-Einstellungen und mit verschiedenen Endgeräten am Oszilloskop zu erzielen.

6.9.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

Vergleichen Sie die Messungen an den verschiedenen Telefonen und beschreiben Sie die Qualität der verschiedenen Geräte.

6.10 Versuch 10 Registrierung bei einem Provider

6.10.1 Zielsetzungen

Das Ziel dieses Versuchs ist es an einem Beispiel zu demonstrieren, wie die Registrierung bei einem Provider erfolgt.

6.10.2 Vorbereitung des Versuchs

Einstellungen des SIP-LAB vorbereiten:

- Provider und Zugangsdaten bereitstellen
- Wo sind welche Einstellungen vorzunehmen

6.10.3 Aufgabenstellung

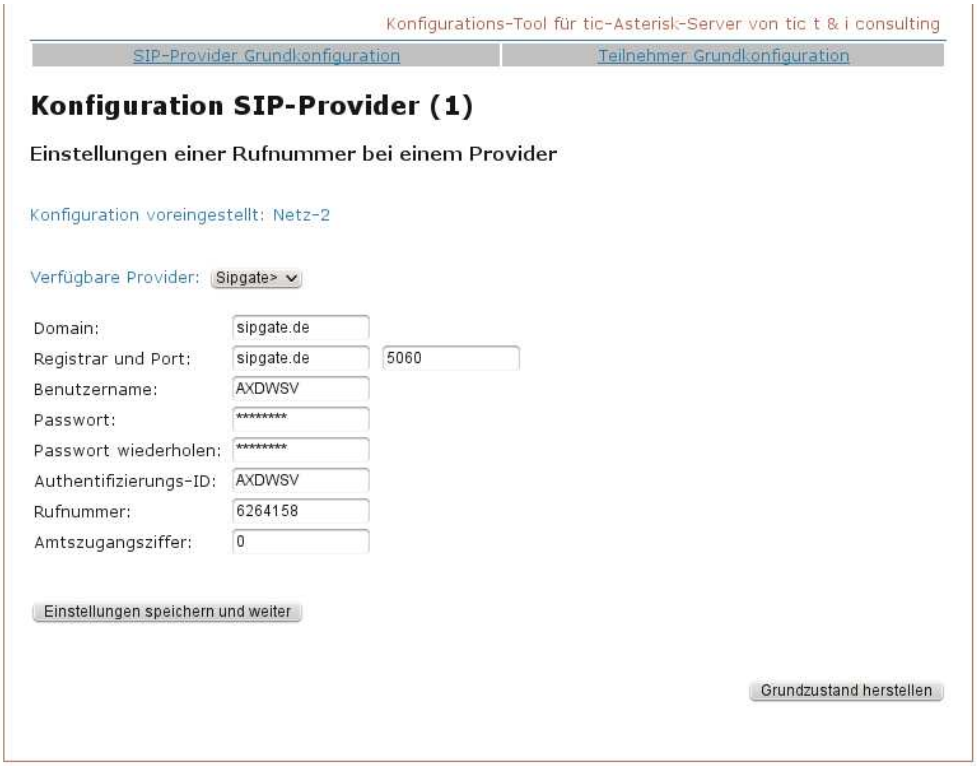
Tragen Sie die Zugangsdaten ein (s. Abbildung 35), d.h.

- IP bzw Domain
- Ben.name / PW
- Rufnummer
- …
- …

Die Zugangsdaten werden bei der Durchführung des Versuchs zur Verfügung gestellt.

Aktivieren Sie den Account, während die Aufzeichnung mit Wireshark aktiviert ist

Abbildung 35: Zugangsdaten Eingabebildschirm



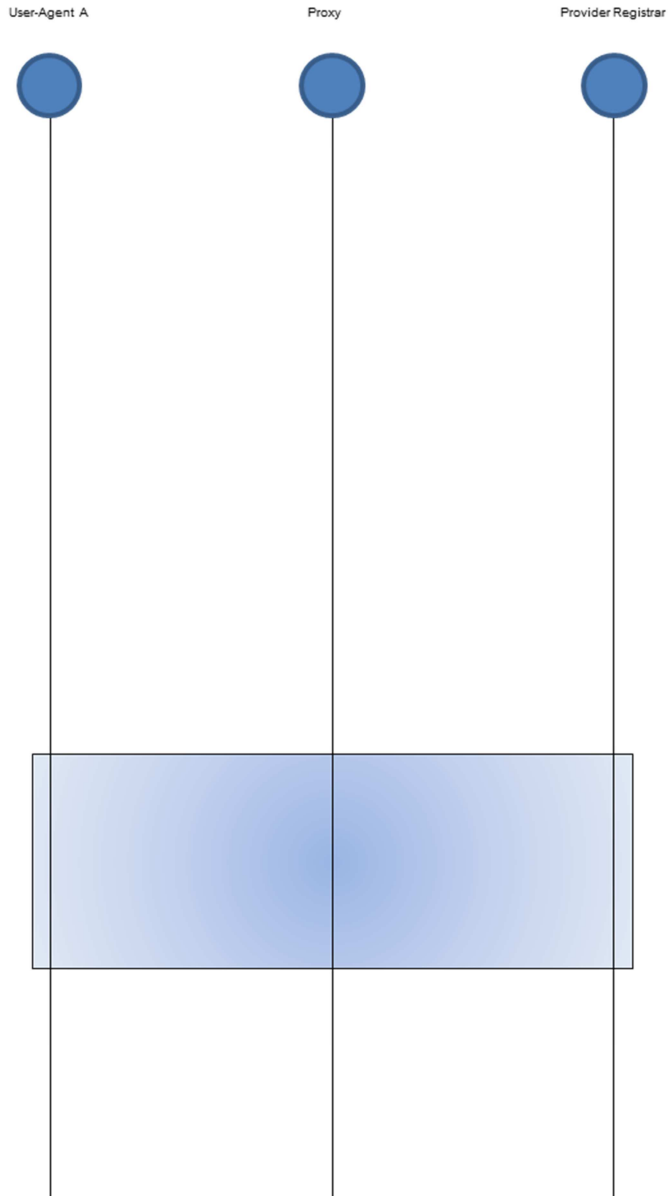
Welche Mittel stehen zur Verfügung um den Registrierungsstatus festzustellen?

6.10.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

SIP-Protokoll der Registrierung mit Wireshark aufzeichnen und in Abbildung 36 eintragen

Abbildung 36: SIP-Protokoll der Registrierung



6.11 Versuch 11 Untersuchung eines Anrufs über einen SIP-Telekommunikationsanbieter

6.11.1 Zielsetzungen

Das Ziel dieses Versuchs ist es zu demonstrieren, wie die Verbindung zu einem Telefonieanbieter hergestellt wird und dass das öffentliche Internet einen großen Einfluss auf die Sprachqualität habe kann.

6.11.2 Vorbereitung des Versuchs

Einstellungen vom vorigen Versuch beibehalten

Telefon (Ursprung) und Zieltelefon (z.B. Mobiltelefon) auswählen.

6.11.3 Aufgabenstellung

Sprachverbindung herstellen, während die Aufzeichnung mit Wireshark aktiviert ist.

6.11.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

RTP-Protokoll mit Wireshark aufzeichnen

Delay bewerten, welche anderen Parameter sind noch von Interesse?

Sprachqualität bewerten

Welche Parameter werden durch den Provider vorgeschlagen/festgelegt?

6.12 Versuch 12 Direktkommunikation zweier User Agents

6.12.1 Zielsetzungen

Dieser Versuch könnte auch ganz am Anfang stehen, da hiermit das grundlegende Verbindungsprinzip der SIP-Telefonie dargestellt wird.

Da aber eine direkte Verbindung zwischen 2 UAs keine praktische Bedeutung hat und die Auswahl der UAs etwas schwierig ist, steht dieser Versuch als Abrundung des SIP-Labors ganz am Ende.

6.12.2 Vorbereitung des Versuchs

In dieser Aufgabe soll zwischen den beiden User Agents eine direkte Gesprächsverbindung hergestellt werden, die nicht wie üblich über einen SIP-Telefonieserver geführt wird, sondern zur Demonstration des grundsätzlichen Abalufs direkt zwischen den UAs aufgebaut wird.

Diese Gesprächsverbindung soll zwischen einem IP-Telefon und einem SIP-User Agent hergestellt werden. Zur Vereinfachung des Versuchs wird hierzu der SIP-User Agent „Linphone“ auf dem Kontroll-PC installiert.

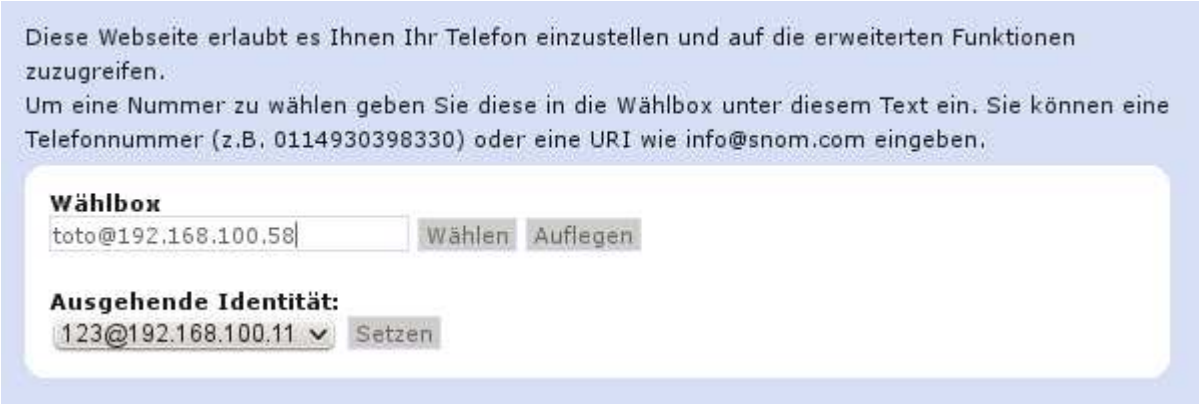
Die Linphone-Installationsdatei (linphone-3.5.0-setup.exe) steht auf den SIP-LABs für Windows zum Download bereit. Die Adressen sind: http://192.168.111.xx, wobei xx=11, … xx=14 für SIP-LAB-3/1 … SIP-LAB-3/4 ist.

6.12.3 Aufgabenstellung

Stellen Sie eine Sprachverbindung zwischen den beiden UAs her und zeichnen Sie den Ablauf auf.

Starten Sie den Verbindungsaufbau auf der Startseite am Telefon Snom 300, siehe Abbildung 37.

Abbildung 37.: Snom 300 Wählbox



Die Angabe des Ziels in der Wählbox muß natürlich den tatsächlichen Gegebenheiten entsprechen.

Der ankommende Anruf wir im UA Linphon wie in Abbildung 38 zu sehen angezeigt

Abbildung 38: Anruf im Linphone

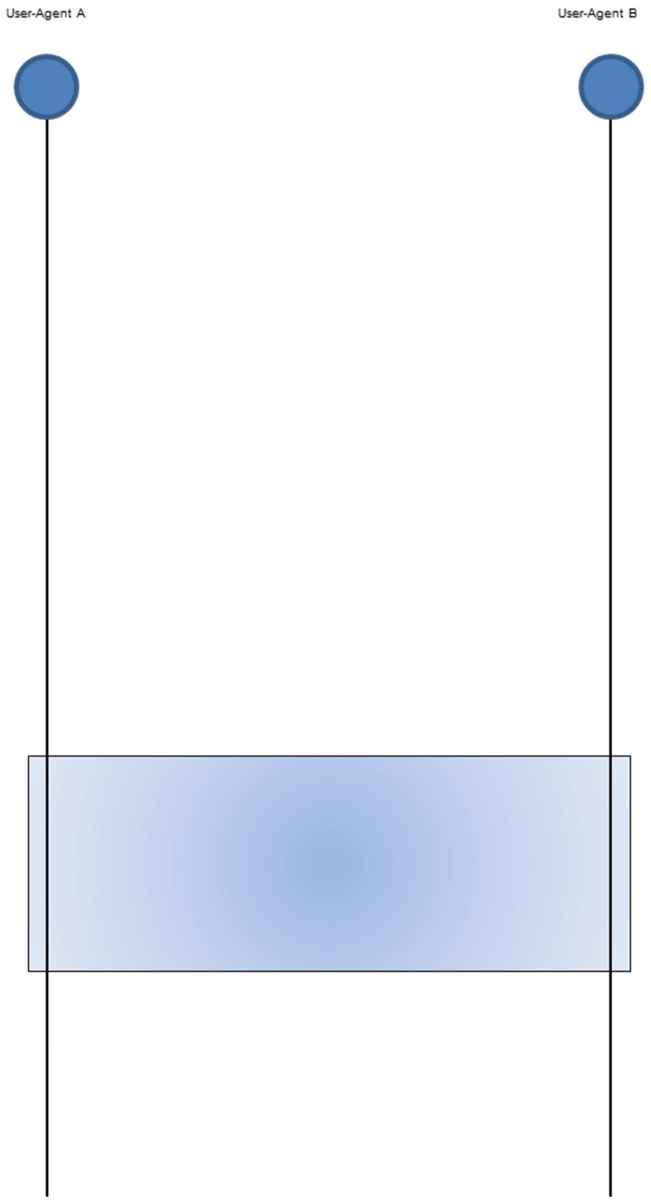
6.12.4 Abschluss des Versuchs

Im Laborbericht dokumentieren:

RTP-Protokoll mit Wireshark

Zeichnen Sie den Ablauf der Nachrichten zwischen den an der Gesprächsverbindung beteiligten Einrichtungen auf und tragen Sie die Nachrichten mit Details in das nachfolgende Diagramm ein.

Abbildung 39:



FRAGE 11:

Warum kommt die Verbindungsart praktisch nicht zum Einsatz?

Probieren Sie auch ob ein Verbindungsaufbau mit den vorhandenen User Agents auch in die andere Richtung einstellbar ist.