

Das SIP-Labor

VoIP-Protokolle in der Aus- und Weiterbildung

Dr. Gerd Siegmund

Mit der Einführung von IP-basierten Kommunikationsprotokollen bilden sich viele komplexe Funktionen auf den Austausch von Anwendungsprotokollen auf der Basis des Internetprotokolls ab. Zusammenhänge sind hier oft nur schwer anschaulich zu vermitteln. Das hier vorgestellte SIP-LAP ist ein Demonstrations-System für den Einsatz in Seminaren oder als Ergänzung zu Vorlesungen an Hochschulen zum Thema Session Initiation Protocol (SIP) und VoIP konzipiert. Das Demonstrationssystem soll dabei den theoretischen Unterricht ergänzen und unterstützen, SIP-Kenntnisse vertiefen und den Teilnehmern die Möglichkeit geben, selbst in einem SIP-System aktiv zu werden (Hands-on). Die Teilnehmer sollen dabei sowohl Grundfunktionen (Registrierung, einfacher Verbindungsaufbau) des Systems aber auch erweiterte Funktionen sowie Fehlerfälle kennen lernen. Die Basis für die Verwendung des Systems durch den Kursteilnehmer ist die Interpretation der ausgetauschten Nachrichten mit Wireshark (früher Ethereal – ein frei verfügbares Analyse-Tool) sowie die Darstellung der übertragenen Nutzinformationen.

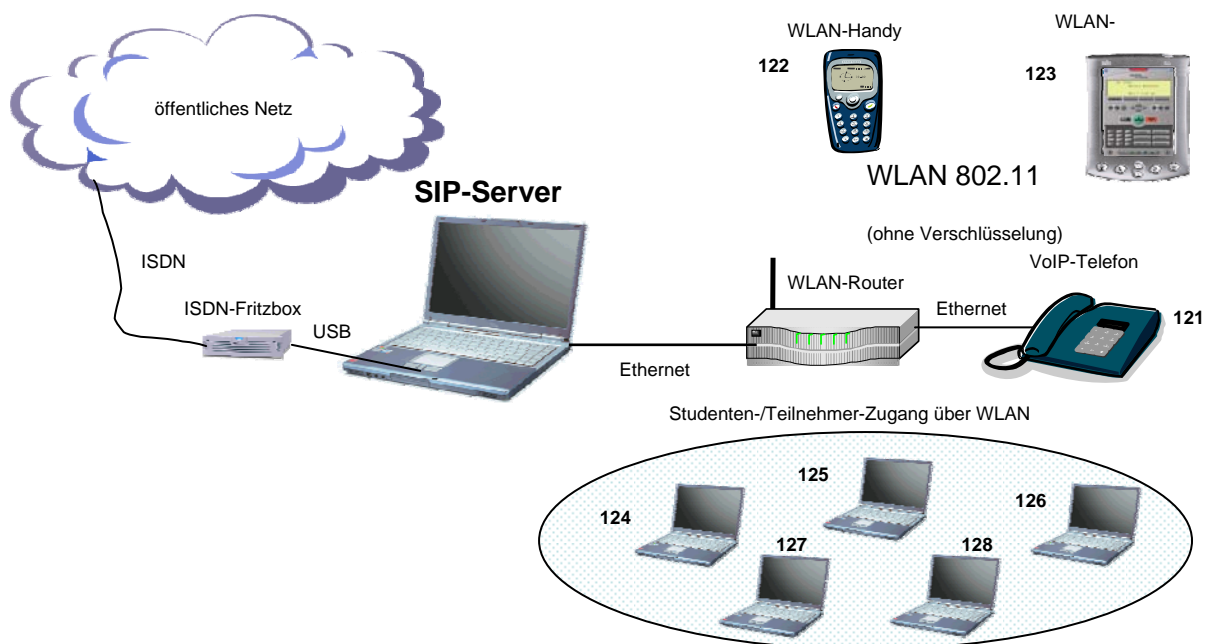


Bild 1. Konfiguration [1]

Das System besteht aus einer Reihe von einzelnen Komponenten, die im Hörsaal bzw. Trainings-Raum zu einem autonomen „Stand-alone“-Netz zusammengeschaltet werden. Ein im Linux Betriebssystem integrierter Netz-Emulator (netem) ermöglicht die Simulation des Verhaltens realer Netze mit deren typischen Laufzeiten, Varianz der Verzögerungen, Paketüberholungen und Paketverlusten. Die Parameter können dabei in weiten Grenzen frei variiert werden. Die Konfiguration besteht aus einfachen Standard-Komponenten, frei verfügbarer Software (Open Source) und kleinen Hilfsprogrammen sowie einer speziell abgestimmten Konfiguration der Systeme.

Den Kern des SIP-LAB bildet ein SIP-Server mit allen notwendigen Hilfsanwendungen. Dieser wird durch die Open-Source-Software „Asterisk“¹⁾ auf Linux-Basis gebildet. Alle notwendigen Anwendungen sind unter Linux auf einem Standard-Notebook installiert.

Die Verbindungen zwischen den Terminals und dem Server werden durch einen Standard Router mit mehreren Ethernet-Schnittstellen und WLAN-Zugang realisiert. Die Kursteilnehmer können damit sehr leicht Teil vom Gesamtsystem werden. Über die ungeschützte WLAN-Schnittstelle kann von Kursteilnehmern mit WLAN-Notebooks die Software für freie SIP-User-Agents mit einem normalen Internet-Browser geladen werden. Nach Installation der Software kann jedem Teilnehmer eine freie Rufnummer zugeordnet werden. Die gewählte Rufnummer muss dabei einer der konfigurierten Rufnummern (s. Abschnitt „Rufnummernplan“) entsprechen.

Optional kann ein Zugang zum öffentlichen Netz über einen ISDN-Adapter mit USB-Schnittstelle realisiert werden. Damit können Gespräche von den SIP-User Agents ins öffentliche Netz geführt werden und diese können hierdurch von außen erreicht.

Die einzelnen Komponenten

Vernetzung

Außer der Stromversorgung und optional einem Standard- ISDN-Basis-Anschluss sind keine Anschlüsse im Schulungsraum erforderlich. Die Vernetzung der Komponenten erfolgt mit Ethernet- bzw. (nur für den ISDN-Adapter) USB-Schnittstellen. Für die Schulungsteilnehmer und zum Anschluss von mobilen SIP-User-Agents wird eine offene WLAN-Schnittstelle bereitgestellt.

DER SIP-SERVER

Im Notebook des Seminarleiters laufen alle zentralen Programme und Server für den Betrieb des autonomen Systems. Neben dem zentralen -Server sind Server für die Registrierung von SIP-Usern (Registrar) automatische, dynamische IP-Adressenvergabe (Dynamic Host Configuration Protocol – DHCP), einen Apache-Web-Server (für die Kursteilnehmer zum Download von zentralen Dateien, Software oder Traces), sowie Hilfsprogramme für den ISDN-USB-Adapter (CAPI).

Der SIP-Server beinhaltet neben dem „Dynamic Module Loader“, einem „Scheduler and I/O Manager“ die Behandlung der Nutzinformationen (File Format API) für die verschiedenen Codecs, einen „Codec-Translator“ für die „Übersetzung“ zwischen verschiedenen Codecs durch die UAs, der „Asterisk Channel API“ für die Bearbeitung der Signalisierung und eine „Asterisk Application API“ zur Unterstützung komplexerer Funktionen und Anwendungen (wie Konferenz, Voicemail, Directories, Paging usw.). Weitere Beschreibungen können dem Asterisk-Handbuch entnommen werden. Die aktuelle Version der Asterisk-Software und des Asterisk-Handbuchs kann unter <http://www.Asterisk.org> und <http://www.digium.com> abgerufen werden, weitere Informationen sind auch in den Büchern [3] und [4].

1)Asterisk ist ein eingetragener Markenname der Fa. Digium

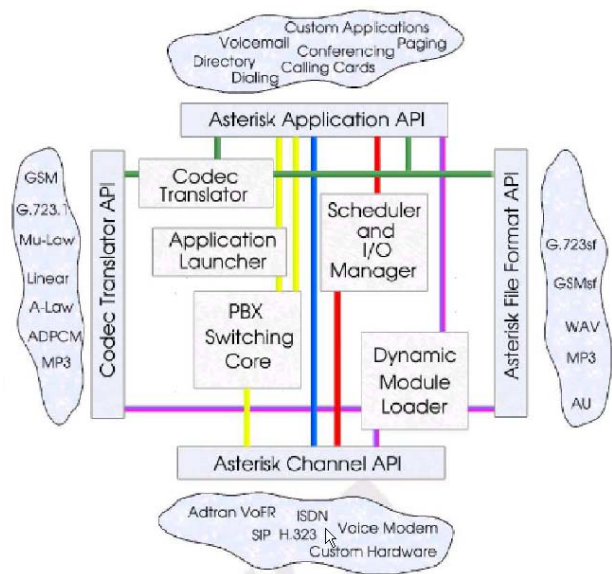


Bild 2. Asterisk-Übersichtsbild [2]

ROUTER

Der Router stellt alle Verbindungen zwischen den einzelnen Systemen (Server, IP-Telefon, Softphone über Ethernet oder WLAN usw.) zur Verfügung. Für den Einsatz des Routers gibt es außer der notwendigen WLAN-Schnittstelle keine besonderen Anforderungen.

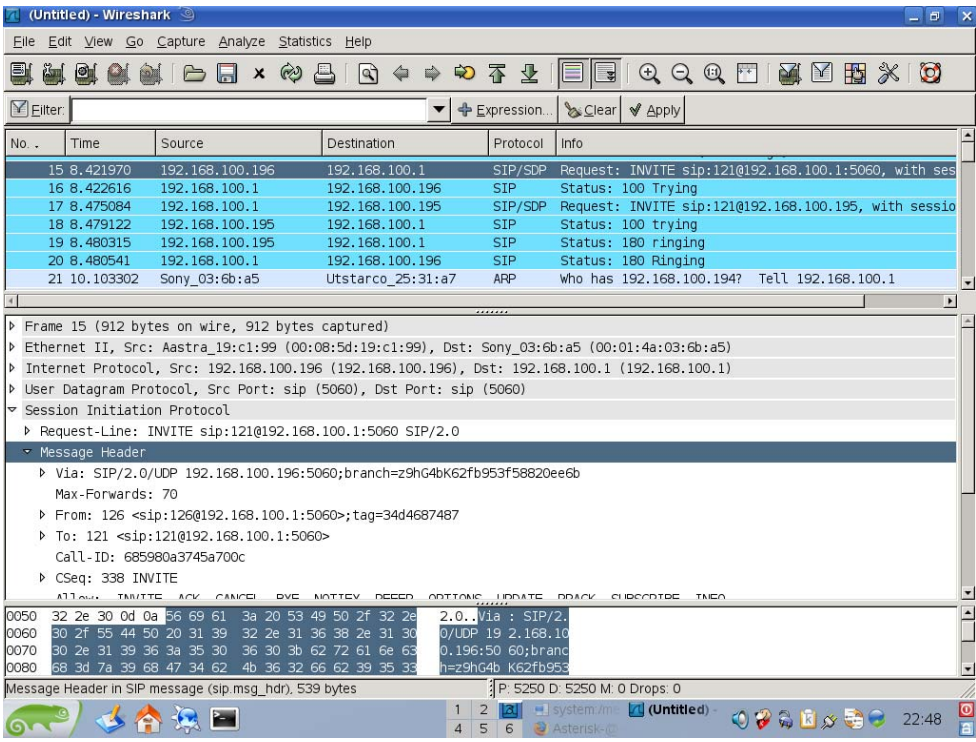


Bild 3. Screenshot WireShark [1]

WIRESHARK

Mit dem Programm Wireshark (früher Ethereal) können für Lehrzwecke Netzchnittstellen am Computer mitgelesen und die transportierten Nachrichten interpretiert und dargestellt werden. Wireshark ist in der dargestellten Version 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. Die meisten Inhalte können dann im zweiten Fenster noch weiter detailliert werden im dritten Fenster wird die komplette Nachricht noch in Hex-Form dargestellt.

Wireshark ist für viele unterschiedliche Rechnerplattformen (Windows, Mac OS, Linux usw.) frei verfügbar. Nach eigener Aussage im Wireshark-Handbuch ist ca. alle 4 bis 8 Wochen eine neue Software-Version verfügbar (<http://www.wireshark.org/download.html>).

Im SIP-LAB wird Wireshark für Linux auf dem SIP-Server und für die SIP-User-Agents die Version auf PC-Windows-, Linux oder MAC-Basis eingesetzt. Der SIP-Server ist im Kommunikationsverlauf der zentrale Netzknoten, der mit dem Initiator und dessen Partner kommuniziert. Die mitgelesenen Nachrichten können in einem eigenen Format gespeichert werden und später (auch auf einer anderen Plattform) ausgewertet werden.

NETZEMULATION (NETEM)

In der aktuellen Linux-Version ist das Netz-Emulationshilfsprogramm (Net Emulation – Netem) integraler Bestandteil des Linux-Kernel. Netem ermöglicht die Simulation, des Verhaltens größerer IP-Netze. Mit ihm lassen sich Verzögerungen (Delay) mit vorgegebener Varianz und Verteilung, Änderung der Paketreihenfolge (Reordering), der Paketverlust (Loss) und Paketduplizierungen (Duplicate) in weiten Bereichen durch den Anwender einstellen. Die Netem-Anweisungen beziehen sich immer auf eine bestimmte Schnittstelle des Systems, auf dem Linux läuft. Die verschiedenen Schnittstellen des Systems, z. B. Eth0 und Eth1, können jeweils unterschiedliche Vorgaben für ihr Verhalten haben. Der SIP-Server im SIP-LAB verwendet nur eine Ethernet-Schnittstelle zum Router, über diese laufen zentral alle Nachrichten im System (Nutz- und Signalisierungsinformationen). Die Einstellungen (Verzögerungen, Verlust usw.) für die Schnittstelle Eth1 wirken sich auf alle Nachrichten in der vom Server ausgehenden Richtung zu dem Router und den anderen Komponenten aus. Die gilt auch für Downloads der Software vom integrierten Web-Server. Grundsätzlich könnten auch weitere Filter angewendet werden, die dann die Einstellung nur für bestimmte Protokoll-Typen wirksam werden lassen (z. B. nur Nutzinformationen von VoIP).

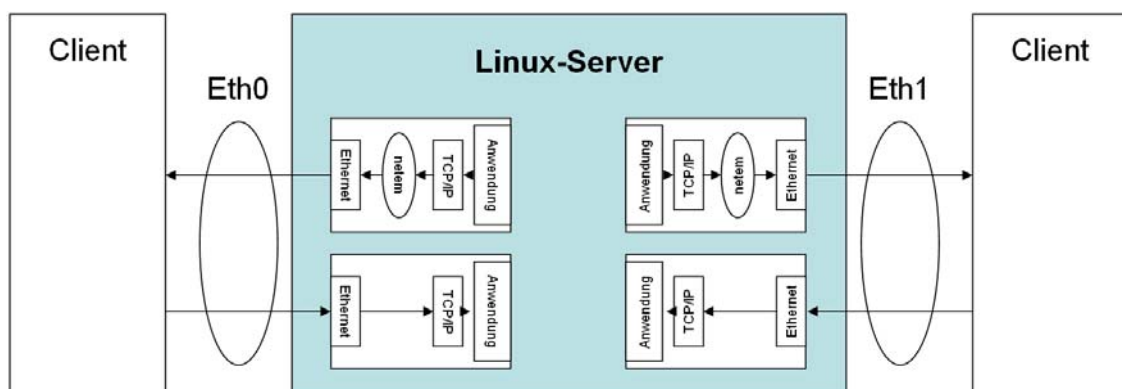


Bild 4. Wirkung von Netem auf Schnittstellen

Die Verzögerungen sind nicht für jedes Paket immer gleich, sondern variieren, wie in realen Netzen, von Paket zu Paket. Die Variation kann in einem festen Rahmen definiert werden oder entspricht einer vorgegeben Verteilungsfunktion. Dabei kann man zwischen der Gleich- und Normal-Verteilung oder einer Verteilung mit einer Langzeitabhängigkeit (long-range dependence, sog. heavy tailed) mit selbstähnlichen Verhalten (z. B. Pareto) oder einer Kombination aus beiden Grundarten wählen. Reine Sprachnetze weisen eine Short-range Dependence auf, d. h. die charakteristischen Werte korrelieren in überschaubaren Zeiträumen, in den IP-Netzen ist das nicht so. Um also das Demo-System ein möglichst reales Verhalten zu geben ist der Start von Netem mit entsprechenden Parametern notwendig.

Das SIP-LAB nutzt nur einen Ausschnitt der Netem-Möglichkeiten, weitere Informationen zu dem vollen Leistungsumfang des Netem-Tools sind unter <http://linux-net.osdl.org/index.php/Netem> zu finden.

WEB-SERVER

Auf dem Notebook wird mit dem Start von Linux ein Apache-Web-Server gestartet, der von jedem, der Zugang zum System hat, erreicht werden kann. Kursteilnehmer können über die WLAN-Schnittstelle so die Software für die Installation eines SIP-User-Agent (beispielsweise den ebenfalls freien X-Lite-Client), Wireshark sowie Traces und weitere Unterlagen zur Schulung auf das eigene System laden. Weitere Dokumente oder Wireshark-Mitschnitte können vom Kursleiter auf dem Server eingestellt werden.

MITWIRKUNG DER TEILNEHMER

Nach dem Download der notwendigen Software vom Web-Server und der Installation sowie dem Start des X-Lite User Agent muss dieser noch für den Betrieb am Demo-System konfiguriert werden. Hierfür sind nur zwei Einstellungen notwendig, der User Agent muss eine eindeutige Adresse (Rufnummer oder URI) erhalten und ihm muss der zugeordnete Server mit seiner festen IP-Adresse (SIP-Server im Notebook) angegeben werden. Mit diesen Einstellungen sollte dann alles funktionieren. Ein Neustart des Systems ist nicht erforderlich.

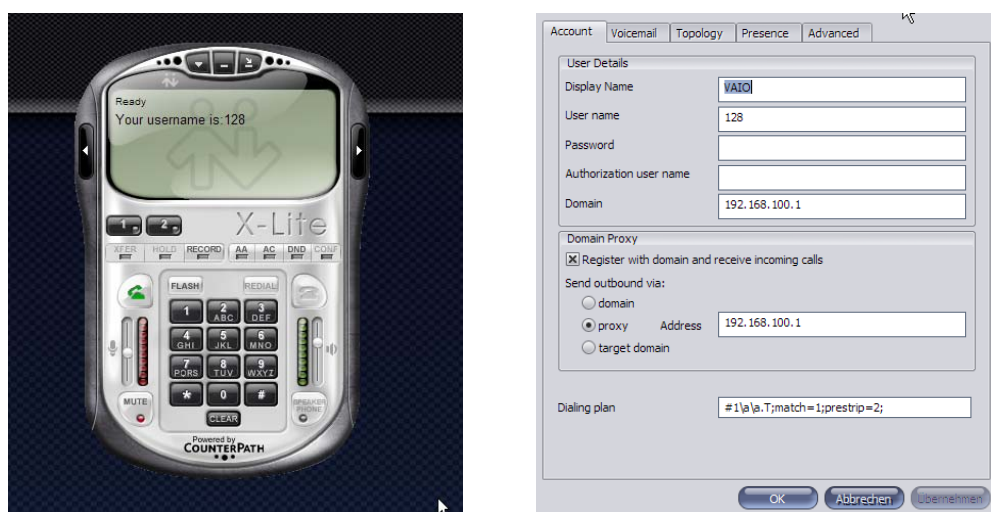


Bild 5. X-Lite und die Konfiguration des UA

Neben dem User Agent kann der Kursteilnehmer auch Wireshark installieren, dann kann er selbst die ausgetauschten Nachrichten mitlesen, speichern und später auswerten. Die Abläufe sind dabei nicht bereits vorbereitet, sondern werden durch die Aktivitäten der Kursteilnehmer erzeugt.

Erzeugung automatischer Abläufe

Zur Erzeugung von Belastungstests oder die Erzeugung von ganzbestimmten SIP-Nachrichten und Abläufe, die ggf. auch beabsichtigte Fehler enthalten, steht mit SIPp (<http://sipp.sourceforge.net>) ein spezieller User Agent zur Verfügung. SIPp ist ebenfalls ein Open Source Programm, das als User Agent Verbindungen initiieren kann. Die vorgebereiteten Szenarios werden im XML-Format vom Dozenten geschrieben und dann gezielt für vorgegeben Testläufe aufgerufen.

SIP-LAB Oberfläche

Von tic wurde ein spezielles Programm entwickelt, das Gesamtsystem komfortabel von einer Web-basierten Oberfläche aus zu steuern. Hiermit wird der SIP-Server und ggf. vorhandene Gateway zum ISDN gestartet, die Einstellungen für netem vorgenommen und Systeminformationen (Registrierte Benutzer, zugeordnete IP-Adressen, Anzahl der Verbindungen usw.) abgerufen werden. Mit dem Start des Servers steht eine vordefinierte Konfiguration zur Verfügung (Bild 6).

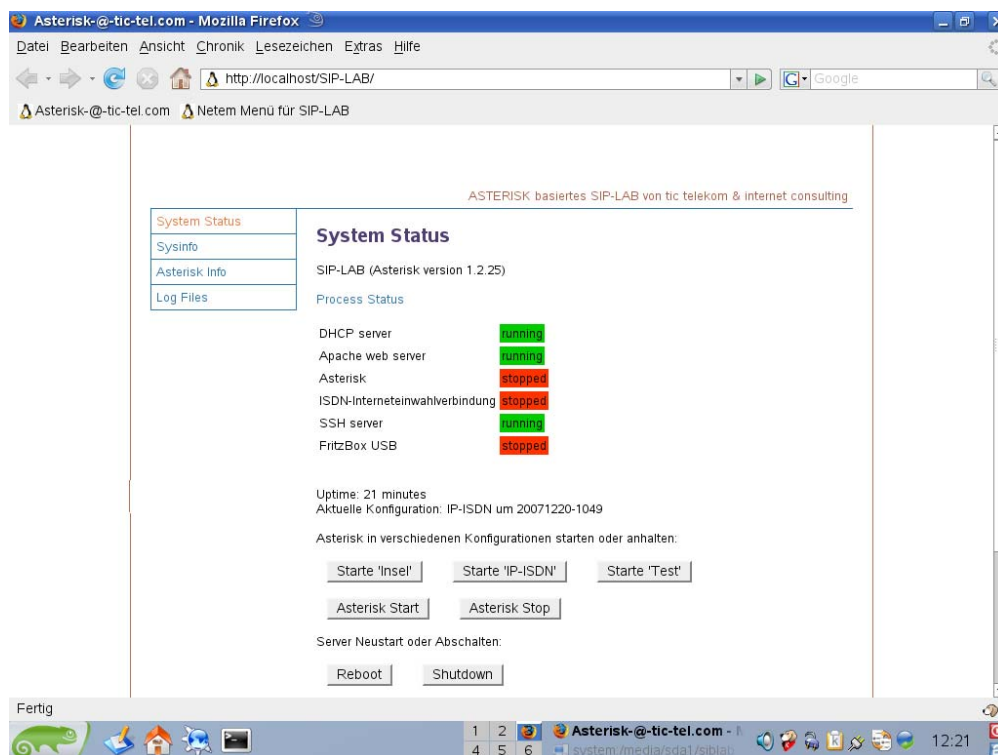


Bild 6. SIP-LAB-Oberfläche von tic

In einem zweiten Fenster können alle Einstellungen von netem eingesehen und verändert werden. Damit können reale Verhältnisse komplexer Netze simuliert werden (Bild 7).

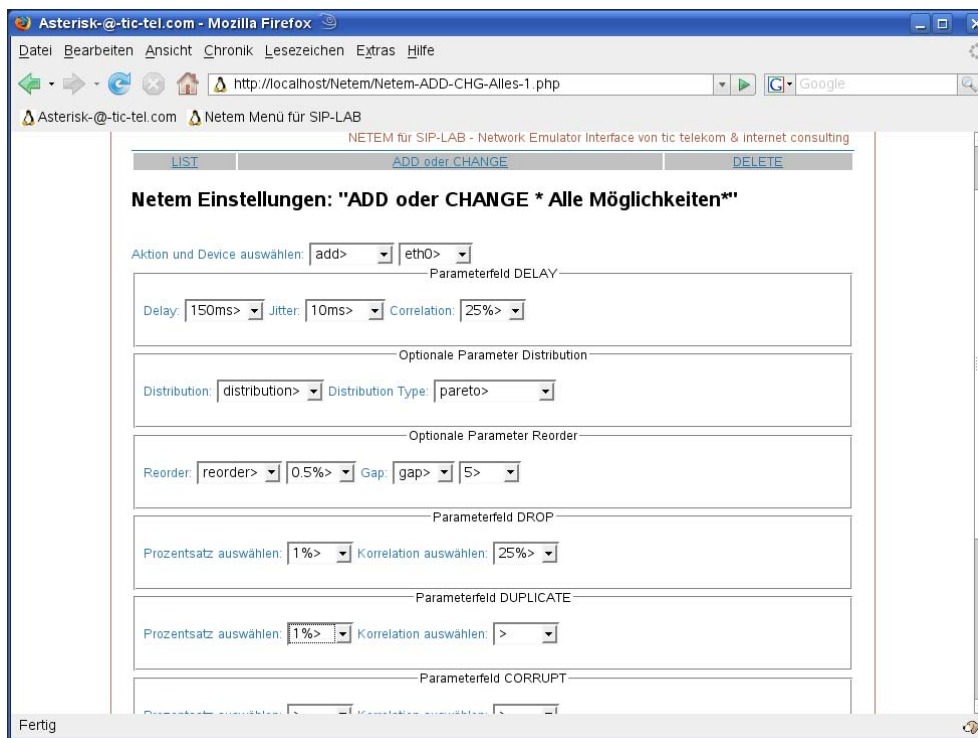


Bild 7. Einstellung der netem-Parameter

Fazit

Die Lehre von den VoIP-Protokollen muss nicht nur graue Theorie bleiben, mit etwas Aufwand kann man aus diesen Standardkomponenten und Open Source Programmen selbst ein SIP-Labor aufbauen und die Zusammenhänge für die Zuhörer „begreifbar“ machen. Das Labor kann theoretische Vorträge ergänzen und als Basis für eine Anzahl von Laborversuchen dienen. In der einfachsten Konfiguration, wie im Bild 1 dargestellt, passen alle Komponenten in einen Pilotenkoffer und können damit in jedem Vortragsraum, ohne besondere Vorkehrungen betrieben werden. Für ein autonomes System ist nur ein 220-V-Anschluss erforderlich. Für Verbindungen zum öffentlichen Netz ist optional ein ISDN- und/oder Internetanschluss erforderlich.

[1] SIP-LAB-Handbuch Ausgabe 3 01.2008, tic telekom & internet consulting <http://www.tic-tel.com>

[2] Aus dem Asterisk-Handbuch <http://www.asterisk.org/> und <http://www.digium.com>

[3] R. Ackermann, H. P. Dittler: IP-Telefonie mit Asterisk, Aufbau von Nebenstellenanlagen für VoIP auf Linux-Basis, dpunkt.verlag, Heidelberg 2007.

[4] G. Flaig, M. Hoffmann, S. Langauf: Internet-Telefonie, VoIP mit Asterisk und SER, Open Source Press, München 2006.