

Voice over IP

Klaus Kusche
Jänner 2016

Inhalt

- **Ziele & Voraussetzungen**
- **Was ist VoIP, warum VoIP ?**
Vorteile, Ziele, Einsatzbereiche
- **Probleme & deren Lösung ...**
 - ... für das “Wählen”**
 - ... für das “Sprechen”**
- **Die finale Hürde:**
NAT & Firewalls

Ziele

Verständnis

- ... der Ziele, Einsatzbereiche und Vorteile von VoIP
- ... der Komponenten von VoIP
- ... der grundlegenden Probleme der Telefonie über IP-Netze
- ... und ihrer Lösungen

Kenntnis

- ... der wichtigsten Abkürzungen, Fachbegriffe und Standards

Voraussetzungen

Grundkenntnisse IP-Netze:

- IP, Unterschied TCP / UDP
- IP-Adressierung
(DHCP, private IP-Adressen, ...)
- DNS, ev. DynDNS
- Begriffe: Server, Proxy, ...
- Firewalls, NAT, VPN

Begriffe (1)

VoIP

= ***“Voice over IP”***

= ***IP-Telefonie***

= “Normales” Telefonieren
(wie mit “altem” Telefon)

über bestehende,

IP-basierte Netzwerke

Begriffe (2)

Gegenteil:

POTS

= ***“Plain Old Telephone Service / System”***

POTS ursprünglich:

Analog, **Leitungs-vermittelt**

=> eigene phys. End-to-End-“Draht”-Verbindung

POTS heute:

Digital, **Zell-vermittelt** (ISDN, ATM, SDH/Sonet)

=> Synchroner End-To-End-Bitstrom

mit garantierter Bandbreite (= “virtuelle Leitung”)

Motivation / Anwendung (1)

Ursprünglich: VoIP für

geschlossene “Inhouse”-Telefonie

= Nebenstellen-Telefonanlagen

in Firmen, Behörden, ...

Ziel:

- In Bürogebäuden: *Statt zwei nur eine Verkabelungs- und Vermittlungs-Infrastruktur*
- Nutzung bestehender Netz- bzw. VPN-Verbindungen zur ***abhörsicheren intern-Telefonie mehrerer Standorte / Außenstellen***

Motivation / Anwendung (2)

Ziel für den Telefon-Kunden:

IP-Anschluss ist meist vorhanden,
zusätzlicher IP-Traffic kostet nichts,
unabhängig vom Ziel-Land

==> Gäbe es ein

einziges, weltweites VoIP-Telefonnetz
(so wie das “alte” Telefonnetz)

könnte man weltweit “gratis” telefonieren!

Derzeit: Meist nur innerhalb eines VoIP-Providers gratis

Motivation / Anwendung (3)

Für Telekom- und Netz-Provider:

Die weltweite IP-Infrastruktur

- ... erreicht mehr Teilnehmer als POTS
- ... zieht viel mehr Investitionen an als POTS
- ... ist flexibler und universeller als POTS
- ... hat (meist) reichlich Kapazitäten frei
- ... nimmt POTS viel Umsatz / Teilnehmer weg
(durch Handy, Skype, ...)

=> ***“Aus 2 mach 1”***: Erhaltung und Ausbau von POTS
ist ***wirtschaftlich nicht mehr argumentierbar***

Anmerkungen (1)

In diesem Vortrag:

VoIP am Beispiel

*hersteller-unabhängiger,
offener Internet-Standards
(= IETF RFC's)*

Konkurrierend:

- Telefon-Standards (ITU H.323 etc.)
- Viele proprietäre Protokolle (Skype, IAX2, XMPP+Jingle, ...)

=> Im Prinzip ähnlich, aber inkompatibel

Anmerkungen (2)

VoIP steht hier stellvertretend für

- diverse Hardware-Technologien:
 - ◆ z.B. *“Voice over Cable”*
für Kabelfernseh-Netze
- beliebige Inhalte:
 - ◆ *Video-Konferenzen* etc.
 - ◆ Multiuser-Echtzeit-Games
 - ◆ ...

Telefonieren ...

... besteht aus

2 getrennten Schritten

Telefonieren ...

... besteht aus

2 getrennten Schritten

1) Wählen:

= Endgerät lokalisieren, Verbindung herstellen

2) Sprechen:

= bidirektional Daten übertragen

Teilnehmer lokalisieren (1)

Gegeben:

Telefon-Nummer

(wegen Interoperabilität mit POTS)

Gesucht:

IP-Adresse

Teilnehmer lokalisieren (1)

Gegeben:

Telefon-Nummer

(wegen Interoperabilität mit POTS)

Gesucht:

IP-Adresse

Probleme:

- IP-Adresse des Endgerätes **ändert sich** immer wieder
(mobile Devices, DHCP, mehrere Geräte im Wechsel, ...)
- Endgeräte haben meist **keinen Domain-Namen**
==> Bestehendes DNS hilft nicht!

Teilnehmer lokalisieren (2)

Lösungsidee ähnlich DynDNS:

Teilnehmer lokalisieren (2)

Lösungsidee ähnlich DynDNS:

Sobald das Endgerät eingeschaltet wird:

*Registriert sich
mit seiner **Telefon-Nummer**
& seiner **aktuellen IP-Adresse**
bei einem **“Vermittlungsserver”***

==> Vermittlungsserver kann Telefon-Nummern
in aktuelle IP-Adressen übersetzen

==> Anrufer fragt IP-Adresse des “Gegenübers”
beim Vermittlungsserver ab

SIP

Protokoll dafür:

SIP (RFC 3261)

= “*Session Initiation Protokoll*”

Zweck:

Server-basierter ***Sitzungsauf- und Abbau***
zwischen 2 oder mehr Endgeräten

(d.h. SIP-Requests laufen von einem Quell-Endgerät
über einen oder mehrere (ev. auch keinen) SIP-Server
zu einem Ziel-Endgerät)

SIP Endgeräte

SIP-Endgerät = “*User Agent*”

- Reine Software (“*Telefon-App*”) am PC / Handy / ...
=> z.B. Telefonieren via WLAN statt via Handy-Netz
- Eigenständiges “*IP-Telefongerät*” mit Netzwerk-Anschluss (Ethernet, WLAN) (der Normalfall in Firmen)
- Für bestehende “*Alt-Telefone*”: *Umsetzer* auf POTS-Analog- bzw. ISDN-Anschluss (z.B. in Home-Routern)

SIP Server

- **“SIP Registrar Server”**:
Zuordnung Nummer ==> IP-Adresse
= das, was ich “Vermittlungsserver” nannte
- **“SIP Proxy Server”**: Wie HTTP-Proxy, z.B. für NAT
- **“SIP Redirect Server”**: Eine Art “Router” für SIP
- **“SIP Gateway”**: Verbindung zum POTS-Netz
oder proprietären VoIP-Netzen

*Die SIP-Server-Infrastruktur ist die wesentliche **Dienstleistung eines VoIP-Providers!***

SIP Protokoll (1)

- Aufbau ähnlich HTTP: *Requests & Responses Plain Text*, lesbar!
- Adressierung mittels URL's:
 - **sip:***user@server* (oder **sips:***...* : *verschlüsselt*)
user ... meist Nummer, auch Name möglich
server ... DNS-Name oder IP-Adresse des SIP-Servers
 - **tel:***nummer*
(ohne Server-Angabe)
- Unterbau meist **UDP** (Ports 5060 bzw. 5061),
wahlweise auch TCP möglich

SIP Protokoll (2)

Beispiele Requests:

- **REGISTER** ... Endgerät meldet sich an
- **INVITE** ... Wählen
- **ACK** ... Datenstrom starten
- **BYE** ... Auflegen

Beispiele Responses:

- **1xx** ... Zwischenstatus (z.B. *“es klingelt”*)
- **200** ... OK
- **3xx** ... Redirect
- **4xx - 6xx** ... Fehler

SIP SDP

SDP

= ***Session Description Protocol***

(Unter-Protokoll von SIP)

Aushandlung der Datenverbindungs-Details:

- Zu verwendendes *Datenübertragungs-Protokoll*
- *IP-Adressen* der Endgeräte,
wenn RTP: *Dynamische IP-Ports*
- *Medien-Typ & Codec*,
Bitrate, Komprimierung, Verschlüsselung, ...

“Wählen” gelöst?

Gegeben:

Telefon-Nummer

(wegen Interoperabilität mit POTS)

Gesucht:

IP-Adresse

“Wählen” gelöst?

Voraussetzung für **sip:...@host** :

*host = Name oder Adresse
des **SIP-Servers** (oder des Endgerätes)
der “Zielperson” muss bekannt sein!*

Ok innerhalb einer Firma / eines VoIP-Providers

Aber: Woher den **host** ermitteln bei ...

- ... **tel:...** ?
- ... Anrufen aus POTS über Gateways?
- ... fremdem / unbekanntem VoIP-Provider?

“Vorwahlen” ?

1. Idee: Ableitung des zuständigen Servers
aus einer “VoIP-Vorwahl” je VoIP-Provider

Ist eine schlechte Idee

- ... bei *Rufnummern-Mitnahme*
aus POTS oder von anderem VoIP-Provider
- ... bei *Notruf- und Sonderdienste*-Nummern
- ... wegen *rechtlicher Lage in D:*
Vorwahl-Bindung von Festnetz-Nummern an Wohnort,
nur geographische Vorwahlen erlaubt
(oder sehr teure 032... Sondernummer)

ENUM

Idee:

Nutzung der bestehenden DNS-Infrastruktur !

ENUM

Idee:

Nutzung der bestehenden DNS-Infrastruktur!

Standard:

ENUM (“ITU E.164 Number Mapping”)

- Abbildung *Telefonnummer* \Rightarrow *Domain-Name*

12345 wird 5.4.3.2.1.e164.arpa

- *DNS-Lookup* dieses ENUM-Namens liefert *Adresse* des für diese Nummer zuständigen *SIP-Servers*

Theorie & Praxis ...

ENUM+SIP würden ein weltweites, einheitliches, provider-unabhängiges VoIP-Netz bieten:

ENUM+SIP würden direkte VoIP-Kontaktaufnahme zu SIP-Servern bzw. zu Kunden aller Provider nur mittels Telefonnummer über normales IP erlauben (an den VoIP-Gateways beider Provider vorbei)

Aber: ENUM (oder die SIP-Ports) werden von VoIP-Providern meist boykottiert bzw. blockiert:

Provider möchten “externe” VoIP-Gespräche Call-By-Call an ihren Gateways abrechnen!

“Sprechen” (1)

Anforderung:

Übertragung eines Bitstroms

(digitalisierte, komprimierte,
verschlüsselte Sprache)

“in Echtzeit”

über IP

“Alte” Übertragungstechniken für digitale Telefonie (ISDN, ATM, SDH/Sonet) wurden genau dafür optimiert: Garantieren ***synchronen Bitstrom fixer Bitrate*** (durch Multiplex- und Zeit-Slot-Verfahren)

“Sprechen” (2)

Genauer:

- ***Konstante bzw. garantierte minimale Datenrate***
(z.B. ISDN: 64 Kbit/sec netto unkomprimiert)
==> sonst Unterbrechung (Ton setzt aus)
- ***Möglichst geringe Laufzeit / Verzögerung***
==> sonst kein Gesprächs-Dialog mehr möglich!
(ab 100 ms Mund-zu-Ohr merklich,
ab 150 ms störend, ab 500 ms gefühlt unbrauchbar)
==> “Zu späte” Daten sind verlorene Daten!
- ***Minimale Datenverluste (< 5 %) sind tolerierbar!***

IP

- Übertragung einzelner Pakete, nicht Streams
- Stark schwankende Paket-Laufzeit
- Ev. verlorene Pakete
- Ev. Paket-Ankunft in falscher Reihenfolge

==> Genau das Gegenteil von dem,
was VoIP braucht!

Aber: IP-Fehler-Behandlung passt für VoIP:
Fehler werden erkannt,
falsche Daten werden einfach **verworfen**

TCP/IP

- Paketweise Übertragung von *Streams*:
Korrekte Reihenfolge, keine Datenverluste, ...

TCP/IP

- Paketweise Übertragung von Streams:
Korrekte Reihenfolge, keine Datenverluste, ...
- Keine Durchsatz- oder *Laufzeit-Garantie*
- Bei *Fehlern / Verlusten*: Paket-Wiederholung

... kann den Datenstrom

beliebig lange verzögern!!!

==> Paket-Wiederholungen sind für VoIP sinnlos,
sogar ausgesprochen konterproduktiv!

==> TCP/IP ist für VoIP erst recht unbrauchbar!

RTP

RTP / SRTP (RFC 3550)

= *“Real-Time Transport Protocol”*

Direkt Endgerät <==> Endgerät (ohne Server!)

über UDP (paketweise!) statt TCP

*==> Muss schwankende Laufzeit, ... **selbst** behandeln:*

- *“Jitter Buffer”*
- Ev. *RTCP* (Real-Time Control Protocol)
für dynamische Bandbreiten-Anpassung usw.

Der Jitter Buffer (1)

Jitter

Hier: *Laufzeit-Schwankungen* der Pakete
(Abweichung der Pakete vom Soll-Ankunfts-Zeitpunkt)

Jitter-Buffer

= *Paket-FIFO* beim Empfänger

- Sammelt einlangende Pakete, *verzögert* sie kurz
 - Liefert sie in *gleichmäßigem Takt* an den Decoder
- ==> *Gleicht einzelne Paket-Verspätungen und -Vertauschungen aus*

Der Jitter Buffer (2)

Tuning-Parameter für die Qualität:

- Jitter-Buffer kurz:
 - **Weniger Latenz**
 - **Mehr Paket-Verlust** durch zu späte Pakete
(weniger Verspätung ausgleichbar)
=> *Mehr Ton-Störungen & Dropouts*
- Jitter-Buffer lang:
 - **Mehr Latenz**
=> *Gesprächs-Dialog stärker beeinträchtigt*
 - **Weniger Paket-Verlust**

Firewalls & NAT (1)

... vertragen sich *ganz schlecht* mit VoIP:

Firewalls & NAT (1)

... vertragen sich *ganz schlecht* mit VoIP:

- Pakete “*von außen nach innen*” sind meist nur zulässig als *Antwort* auf Pakete “*von innen nach außen*”

Bei SIP und RTP kommt
erstes Paket ev. von außen!

- Firewall-Regeln gelten für *fixe Portnummern*

RTP handelt über SIP
dynamisch wechselnde Portnummern aus!

Firewalls & NAT (2)

- Geräte in Firmen und Geräte hinter Home-Routern haben meist ***private IP-Adressen*** (10.x.x.x, 192.168.x.x, ...) ==> sind am Internet ***nicht gültig!***

SIP überträgt IP-Adressen in den Nutzdaten

==> Endgerät meldet sich mit ***privater*** Adresse am SIP-Server an

==> “Anderes Ende” bekommt

ungültige private Adresse des Gegenübers

==> Direkte RTP-Verbindung schlägt fehl!

Firewalls & NAT (3)

- Bei *mehreren Endgeräten* hinter einem NAT-Router:
Haben nach außen *alle dieselbe IP-Adresse*
Wenn mehrere Endgeräte zugleich SIP nutzen:
=> Für außenstehenden SIP-Server
nicht mehr unterscheidbar
(alle melden sich mit gleicher Adresse an)
=> NAT-Router kann *einlangende SIP-Pakete*
nicht auf das richtige Endgerät weiterleiten

Firewalls & NAT (4)

***Ständiges Ärgernis,
keine einheitliche Optimal-Lösung!***

- Viel ***“schwarze Magie”***:
 - ***STUN*** (RFC 3489, “*Session Traversal Utilities for NAT*”)
 - Abgleich *Paket-Header-Adresse* mit *SIP-Adresse* im Paket
- Firewall (NAT-Router) ***analysiert und transformiert Inhalt von SIP-Paketen, gibt RTP entsprechend frei***
(Linux NAT beherrscht SIP Connection Tracking!)
- ***SIP-Proxy-Server*** (oder SIP-User-Agent)
direkt auf Firewall bzw. NAT-Router laufen lassen
- VoIP über ***VPN-Tunnel*** durch NAT bzw. Firewall

“The end”

Fragen?