Internet Engineering Task Force
INTERNET-DRAFT
draft-schulzrinne-iptel-challenges-00.ps

Henning Schulzrinne
Columbia U.
June 1, 2002
Expires: December 2002

Internet Telephony — Challenges and Open Issues

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Copyright Notice

Copyright (c) The Internet Society (2002). All Rights Reserved.

Contents

Abstract

Implementing Voice-over-IP poses major challenges, as it is a service with a long-set tradition of expectations, regulatory encumbrances and a large range of end systems to be used by mostly non-technical users. This document enumerates some of these challenges and possible solutions.

1 Introduction

When other Internet services were introduced, they provided a fundamentally new communication capability and thus users were willing to accept limitations in reach, reliability, quality, usability and security. However, Internet telephony will be measured against the very mature technology of circuit-switched telephony, even if it may offer new services, lower cost and better quality in the long run.

Below, we summarize some of the key challenges that differentiate VoIP from traditional Internet services.

1.1 Reliability

If Internet telephony supplants traditional telephony, it has to be reliable enough to do without a back-up means of communication and to serve emergency needs where property and human lives are at risk. Using the telephone to call tech support when "the network" is down is no longer an option. (Maybe users will have to start sending postcards instead.)

The yardstick for traditional telephony is a reliability of "five nines", or 99.999%, equivalent to no more than five minutes of down time per year. This is only achievable, for example, if access routers can be upgraded without taking them down or if there is a second standby router.

Current, typical ISP reliability is between 99% and 99.9%, although it is hard to compare these measurements to those of the PSTN. In assessing reliability, we currently do not have a coherent way to measure reliability, while a number of well-defined metrics exist in the telephone network, based on the impact of outages on end users. For example, does it count as downtime only if an Internet telephone can make no outgoing phone calls? Or if a certain fraction of the Internet is unreachable? Does uptime require a minimum quality, in terms of bandwidth, loss and delay?

A related problem is that circuit restoration times in modern PSTN plants is on the order of 50 ms, while typical routing restoration takes seconds to minutes [1].

If IP restoration times could be made to approach PSTN intervals, IP telephony reliability could be better than PSTN reliability, as PSTN switch failures will cause the termination of all pending calls, while IP router and switch failover should not affect the call state beyond the brief restoration interval. (As discussed below, the traditional notion of short calls may not hold in the future, so that maintaining calls over long durations may become more important.)

1.2 Autoconfiguration

Traditional telephones are probably the devices requiring the least amount of configuration to make work. (We will ignore trying to configure speed-dial buttons or similar functionality.) To be a plausible replacement for existing phones, Internet phones have to be able to plug into a network and function. Beyond the standard network parameters such as address and DNS server, auto configuration includes finding the outbound signaling server [2] to traverse firewalls or the user identity for single-user devices. (It would be unreasonable to assume that users have to program their own "phone number" into an Internet phone.) It is desirable to be able to operate a local network without servers, as described in the zero-configuration requirements [3].

1.3 Embedded Systems

Telephones are classical network appliances. Unlike the Internet appliances of recent hype, Internet telephones may well occupy an even lower rung of end system capabilities. (Here, Internet telephones are defined as multimedia networked devices that connect directly to a subnet, typically an Ethernet, without the assistance of another host.) For example, they will almost invariably have no disk storage and very limited memory (of at most a few hundred kB to a few MB) and operating system capabilities, given that the true (unsubsidized) price point for conventional analog telephones is between $20 and $500. Indeed, it is quite possible to design an Internet telephone that does not support TCP. This strongly impacts the types of software-intensive capabilities that can be assumed to exist in such end systems, including encryption.

Other examples of Internet telephony embedded systems include small Internet telephony gateways, such as residential gateways in DSL and cable modems.

Internet telephones need to be able to upgrade their software remotely, typically done via tftp [4] to minimize code complexity.

1.4 Interoperation

Unlike previous Internet technologies, seamless interoperation with existing systems will be required from the very beginning. (Email and web, for example, only provided gateways to other services later in their development; there was no expectation that a user could just send email to a fax machine or that every email address was reachable by fax.) The problem is made harder by the fact that there are a wide variety of ways to interconnect to the PSTN. In the simplest mode, a PSTN subscriber is only visible to the VoIP user as an Internet-telephony address, and each VoIP user appears at a fixed telephone number. Effectively, gateways emulate subscriber connections. However, it is also possible to integrate VoIP gateways more closely into the fabric of the PSTN, by making them participants in Signaling System 7 networks.

1.5 Symmetry

Most popular Internet applications are client-server, with inter-domain TCP connections made from a large number of clients to a relatively much smaller number of servers. (We ignore here "connections", such as host network management, that are primarily intra-domain.) This asymmetry enables the current generation of firewalls and NATs.

Internet telephony breaks this model, as every end system that wants to be called also becomes a "server". Additionally, unlike web and email, Internet telephony connections almost inherently consist of two classes of streams, namely a control stream and one or more data streams. Combining the two for ease of NAT and firewall handling greatly limits the types of services that can be deployed.

While firewalls can be made to recognize call signaling and open up "holes" for the associated data streams [5], NATs [6] face a more fundamental problem. If every device is to be externally reachable, it has to be allocated an external address or address/port tuple. Thus, the IP address conservation efficiency of NATs is reduced. This problem does not appear to be affected by the middle box communication efforts (MIDCOM), as it addresses how sessions from the "inside" can reach devices on the "outside".

An alternative is the application-layer termination of all inbound media streams at the NAT [7]. Application-layer termination "outside" the NAT will be necessary for inbound calls and if both caller and callee are behind a NAT.

1.6 Quality of Service

Internet telephony imposes constraints on loss and delay. Losses of up to about 5% can be tolerated or compensated for, with the precise rate depending on the codec and the distribution of the losses in time, but beyond about 20% short-term loss even techniques such as forward error correction [8] become ineffective or counter-productive. (Due to delay constraints, retransmission is generally not considered effective for wide-area communications.)

Acceptable Internet telephony delays depend largely on whether the user-to-user connection contains "two wire" pieces or is "four wire" end-to-end. Two-wire refers to traditional analog subscriber lines where audio is sent and received over a single pair of wires, while "four wire" is the term for separate transmission facilities in each direction. Digital subscriber and long-distance lines are all four-wire. So-called hybrids, traditionally special transformers or electronic equivalents, converts between two-wire and four-wire circuits. Since the splitting of inbound and outbound audio is not perfect, this conversion introduces hybrid echoes. If the one-way delays reach about 30 ms, echo cancellers are required. Echo cancellers are also required if there is any significant "leakage" from the far-end speaker to microphone, as will be the case for speakerphones and standard PC speaker-microphone arrangements. (Sound cards with built-in echo cancellation are available.)

With echo cancellers, one-way delays of up to 400 ms are tolerable, although noticeable, with delays of up to around 100 ms preferred [9]. Beyond that, conversational "hand-off" becomes difficult and communication becomes half-duplex.

RFC 2681 [10] describes metrics for round-trip delay measurements. Round-trip delay may be a good first-order predictor of the one-way delays experienced by each side of a conversation, but RFC 2681 lists a number of caveats when extrapolating from round-trip to one-way delays. Thus, the one-way metrics in RFC 2679 [11] may be more appropriate, even though their measurement relies on tightly synchronized clocks.

In addition, the Poisson sampling employed in RFC 2679 and RFC 2681 is not a good representation of the arrival process of audio packets, which can be better characterized as an on-off process with periodic arrivals during relatively long time periods ("talk spurts"). The author is not aware of any comparisons between results derived from Poisson sampling and an arrival process more closely resembling actual traffic statistics.

Finally, the statistics in Section 4 of RFC 2681 or Section 5 of RFC 2679 are not necessarily good predictors of the actual delay experienced by VoIP implementations, as the delay is not just the network delay, but the sum of the encoding, transmission, receive, playout and audio delivery delay. The playout delay is a function of the variable components of the network delay. Playout delay algorithms typically try to achieve a minimum delay where the number of packets that are discarded due to excessive delay is kept below a design threshold. This requires the receiving end system to estimate and predict the delay statistics of the network [8], typically on periods of one talkspurt. Short of actually simulating the playout delay algorithm based on trace data, the 95th-percentile statistic in RFC 2681 and RFC 2679 may turn out to be a reasonable algorithm-independent predictor of the playout delay. Another reasonable statistic may be a multiple of the delay variance plus the minimum delay.

If an alternate symmetric-delay audio path or a single radio source with known propagation statistics is available, it is possible to very accurately estimate end-to-end delays, from microphone to speaker, even without synchronized clocks. Using statististical correlation techniques on audio samples, it is possible to estimate the relative delay difference between the Internet path and the reference path, e.g., a PSTN connection. The latter can be assumed to have symmetric delays, which are readily estimated via an acoustic "ping" test.

Average available bulk-transfer bandwidth [12] is not a good indication of whether quality is acceptable, as short periods of high-loss that would normally just throttle back TCP connections, but lead to noticeable interruptions in audio service.

A better predictor of user-perceived quality-of-service are one-way loss metrics [13, 14], either averages [13] or, better, metrics that reflect the correlation between losses [14]. It remains an open problem which metric is best suited to reflect the perceptual impact of losses, taking into account the decay of prediction errors and the ability of audio and video loss concealment algorithms. It would appear to be helpful and relatively easy to extend loss pattern measurements by taking forward error correction into account [15, 16].

The caveat noted earlier about the applicability of Poisson sampling to the small-period periodic packet streams of typical audio streams applies here as well. Also, for access networks, some audio and video streams may no longer be usefully treated as mere "observers" of packet loss, but their addition may significantly increase packet loss.

In general, Internet telephony is far more sensitive to short-term disruptions of service than traditional data services. Thus, statements about low average delays and losses of Internet backbones or local area networks may not necessarily be indicative of their ability to support commercial-grade Internet telephony services.

1.7 Internet Impact

Beyond the requirements on reliability, quality of service and symmetry, Internet telephony is also likely to have other operational impacts on the Internet. In particular, Internet telephony packets at current audio bandwidth have to be short to minimize delays. For example, G.711 (64 kb/s) packets are usually no more than 200 bytes long, including headers. More highly compressed codecs have even shorter packets. For example, a low-latency codec, G.729, generates payloads of 20 bytes for a typical 20 ms packetization interval. Depending on the volume of voice traffic, this means that designing routers to handle line rates only for Ethernet-MTU-sized packets will not be feasible. In addition, packet header overheads imposed by encapsulation, IPsec and IPv6 are relatively more significant. Should video become more prevalent, long packets will again dominate, as decent-quality motion video requires about 256 kb/s or packets of about 1000 bytes.

On the positive side, voice streams are either periodic or periodic with (speech) pauses. In large multiplexing scenarios, utilizations of close to 90% [check this!] can be reached.

Internet telephony is also a likely user of small-group multicast solutions. Unlike traditional telephony, where adding a third party to an existing two-party call is relatively easy in modern PBXs, a "classical" Internet solution would have to transition from a standard unicast setup to a shared multicast address, with a host of currently unavailable machinery such as address allocation, multicast routing and the like. (It is not likely to be explainable to users that only users served by a certain long-distance company or ISP could be added into a three-way phone call . . .)

However, since many access facilities such as wireless will remain very bandwidth-limited, initial implementations seem to favor server-based mixing ("multipoint control units", MCUs) instead of end-system mixing, as even for silence-suppressed audio the simultaneous arrival of two or three audio streams is not unlikely.

2 Open Issues

The IETF Internet telephony and conferencing architecture is now reasonably complete to support basic telephony services, but a number of infrastructure requirements and services remain unaddressed.

2.1 Naming and Directories

Internet telephony signaling, both SIP and H.323, names end points similar to email, as "user@domain", expressed as a URL. Names are assigned based on providers, either communication providers or infrastructureless mapping services. In practice, this takes three forms: domains based on the employer or membership organization (such as IEEE or ACM), the Internet service provider or a "web email"-like service.

End points typically represent humans, but, as in email, "robots" of various kinds can also be named (such as a conference on a conference server) and a single name can correspond to a group of people or a functional designation ("postmaster", "support"), akin to email lists and standard functional names [17].

In addition, it has been suggested to use SIP URIs to identify stored voicemail messages [18].

(We'll ignore here that naming persons and resources is not fundamentally different, although in practice two rather different means have evolved, based on the origins in email and file-system-derived gopher and web servers, expressed as "user@domain" and "domain/name/name [...]". The email-like naming scheme tends to be shorter, about 20 characters on average, and more pronouncable, while hierarchical URLs work better for deeply-nested hierarchies. However, the use of email-style identifiers to name persons and robots approximating them for ad-hoc communication has become so entrenched that changing the approach is unlikely to be worth the user confusion.)

Even without the constraints of the current directoryless system, it appears difficult to come up with a scalable scheme that offers provider-independent naming based on a person's civil name. A possible outcome is that users will pick globally unique and permanent identifiers, akin to the current "handles" in IRC and on-line services. (The handle system [19] also falls into this category, although it is probably not suited for direct human use.) This system works reasonably well, however, only if name hoarding can be prevented. On-line services accomplish this implicitly by requiring a paid membership that discourages large-scale name grabbing. It may be possible to integrate global nick names for people in one of the friendly naming schemes, such as [20].

It is advantageous to reduce the number of different identifiers that a user has to employ for different communications services. Thus, it is expected that SIP addresses will be valid as email addresses, while the converse may not necessarily be true as not all domains will initially offer VoIP proxy or gatekeeper services. Having a common identity allows for simple authentication that confirms that a caller is reachable by that email address, as discussed in Section **??**.

As pointed out in Section **??**, identifiers are much more plentiful and much cheaper to acquire in Internet telephony than in the PSTN. Thus, the combination of media-neutrality and abudance makes it likely that a single individual will concurrently use many different identifiers, but based on her roles and privacy considerations, not the nature of the communications mechanism.

SIP also supports the mapping from a SIP URI to any number of other URIs, such as HTTP, `mailto` or LDAP URIs, offering a simple business card service that can be dynamically tailored to the caller, possibly programmatically. (This differs from standard directory services which tend to give the same answer, or no answer, to each query.)

The current LDAP Internet persona [21, 22] does not contain fields for Internet telephony; these should be added.

The rescap effort [23] allows a querier to obtain information about the capabilities associated with an RFC 822 address. While this functionality could also be used for Internet telephony, there appears to be less need, as, for example, the SIP OPTIONS request already offers this functionality. Also, as a directory service, rescap is likely to return fairly static information, independent of the querier, while Internet telephony responses commonly vary in detail and capabilities based on the current device used by the destination, time of day and identity of the querier.

It appears unlikely that large-scale white-pages directories of individuals outside closed organizations will become common, as even traditional telephone white pages directories are decreasing in usefulness due to the rapid increase in unlisted numbers and the diminishing detail provided in entries. Since it is hard to protect against abuse of this information by spammers and telemarketers, users have little incentive to list themselves. (Fortunately, unlike telephone numbers, alphanumeric identifiers are sufficiently sparse that simply dialing random numbers is not likely to be viable.)

Also, the current IETF directory mechanism, LDAP [24], seems ill-suited as an Internet telephony directory, as it is overly complex for simple, resource-constrained end systems. If there are directories, a simple query expressed as an HTTP cgi request, returning an XML object, would be far more likely to be implemented in real systems. Also, referral and identifying the starting point of the directory search remain difficult and cumbersome. IP telephony may find the on-going discussions about formalizing whois of some relevance for simple match-across-all-fields searches commonly found in the autocompletion feature of email clients.

The UDDI (Universal Description, Discovery and Integration) effort is aiming to create a SOAP-based infrastructure for white and yellow pages, with decentralized servers providing the illusion of a single directory. However, UDDI aims to allow applications to discover corporate web services, not typically invididuals. However, the same basic infrastructure may also be suitable for a more generic directory service.

2.2 Capability Negotiation

The current IETF IP telephony capability negotiation mechanism is based on (ab)using the Session Description Protocol (SDP) [25], originally designed for announcing the parameters of a multicast session. For IP telephony, each session description announces the media types and list of supported codecs.

SDP has a number of limitations for unicast sessions:

- SDP cannot express restrictions or preferences among media types or codecs. For example, the description cannot express that one side would like to send one of two different media streams, each with a different, partially-overlapping set of codecs.
- The sender of a session description has to be prepared to receive any one of the list of codecs, changed at any time, without being able to restrict the concurrent set. The latter ability is relevant primarily for embedded-systems implementations that can only hold a small number of codecs in code memory at any given point in time.
- Cross-media restrictions on codec choice cannot be expressed. For example, CPU constraints may limit a system to sending or receiving G.711 (uncompressed) audio if it uses a video codec with motion compensation.

- SDP has no ability to describe codecs for re-use across media sessions within a description.
- Differences between send and receive capabilities can be expressed only with peculiar syntax and by declaring a single RTP session as two sessions, one send-only and one receive-only. Labeling sessions [26] can make this relationship between multiple sessions more explicit.
- Since media entries are not labeled, adding and deleting media entries is error-prone in SDP, as media are associated by position between initiator and responder.

Efforts are under way to define a true negotiation and capability protocol, currently referred to as SDPng [27]. Its relationship to SMIL [28], a description of time-sequenced delivery of streaming media, remains to be explored.

SDP can describe recurring sessions, although this capability appears to be used little outside the realm of multicast session announcement. (From a user-interface perspective, it may be difficult to convey to a callee that she is being rung for a call that will actually start in three hours and repeat every other day.)

It remains to be decided whether multi-party negotiation is to be included in the description, where parties would somehow vote to arrive at a common set of stream parameters based on the preferences expressed. Such a capability is likely only to be of use when a multi-party conferences is negotiated ahead of time, rather than inviting participants one-by-one. (There seems to be general agreement that trying to renegotiate existing session parameters upon arrival of new members is not scalable.) It may be feasible to separate capabilities from the declaration of sending and receiving intent, as long as any voting algorithm is guaranteed to arrive at a consistent set of choices across all participants. However, for anything but very small groups, arriving at a consistent view of the membership and any-to-any communication is likely to be unreliable.

2.3 Application Configuration

It appears likely that IP telephony systems, both software running on PCs and embedded systems ("IP telephones"), will be deployed in large numbers across organizations. Current implementations have chosen widely diverging ways of configuring upper-layer (SIP and RTP) protocol configuration information, ranging from several different tftp file formats to embedded web servers.

ACAP [29] may be suited for this purpose, but requires implementation of an additional non-trivial protocol and the resolution as to how end systems are identified.

Issues related to configuring IP telephony user agents are discussed in [30].

2.3.1 Phone Services

Some services that are easy to provide in PBXes and the PSTN are somewhat harder to provide in an Internet environment. In general, it may not be appropriate to try to emulate PSTN or PBX services exactly, as they are often derived from the centralized, single service nature of the PSTN. Some examples should suffice, [31] contains others.

The typical "party line" service found in residences and small businesses allows to connect a small number of phones to the same wires, which then ring together, stop ringing when the first receiver is picked up, but allow every member of the household to participate in the conversation, all without any support from the central office. A similar service in a VoIP environment requires either global multicast or a per-home media mixer for media distribution and a proxy server with parallel search. In order to emulate the ability to join an existing conversation, the end system has to remember current calls and has to have a mechanism to join the call at the mixer.

As discussed above, the trust model differs in the PSTN, with terminating operators considered trusted parties by the originating carrier. This makes it possible, for example, to have the terminating carrier perform call filtering without revealing the caller's number to the callee.

2.4 Event Notification, Presence and Instant Messaging

Currently, a major annoyance of the telephone service is that only a small fraction of call attempts actually succeed [32], with calls reaching either a busy line, no answer or an answering machine. With few exceptions such as call back services in PBX, the caller is limited to repeatedly calling (i.e., polling) the callee until it reaches the callee in the desired mode, i.e., a live human being or an answering device.

This is one example where a notification service would help. Instead of calling repeatedly, the caller would subscribe to the callee and could then make a call when the callee state indicates that either the line or the person is available. (Or, conversely, target the call such that it is likely to reach an answering machine if a less personal interaction is desired.) Traditional presence services [33] can offer this functionality, but they need to be tied to the communications device to avoid having the user manipulate her availability state manually.

Presence services can be viewed as a special case of the general event notification problem. Here, events are viewed as changes in state of some object in the network, where such state often reflects non-network conditions. Events are aperiodic, i.e., do not occur at regular intervals, asynchronous (not tied to a particular clock) and generally unpredictable. Thus, while an audio source sampled every 125 microseconds generates a state change, this is not an event according to this definition. (MIDI actions, on the other hand, could be treated as such.)

Event notification is only supported indirectly in the current Internet architecture. The Internet currently offers three basic standardized application-layer services: data retrieval (ftp, http, rcp, scp), asynchronous notification (email), and interactive login (telnet, ssh). However, it is missing a standardized generic synchronous event notification service. The event model has proven to be quite useful as a programming abstraction within operating systems, as it fits naturally for systems that are reactive, i.e., have to deal with external inputs such as user actions (mouse movements, keyboard input), network packets and external events such as sensor inputs. However, this service is not generally available across the Internet. Email, viewed as an asynchronous notification service, comes close, but since it requires a periodic polling by the user or application via POP or IMAP to determine whether an event has occurred, it is not suitable for events that require rapid reaction. Presence systems, such as the various proprietary presence and instant messaging systems, are similar, but limited in scope to narrow types of events, namely the change in the availability of a person. Also, many existing systems rely on a central server and thus do not scale easily to large numbers of users and events.

With signaling, event notification shares the need to reach abstract application-layer entities that may change network attachment points and even devices between event notifications.

A number of telephony services can be unified under a single mechanism once viewed as events. This includes call waiting services, voicemail/email notification [34], conference membership management and control, and supervised call transfer [35].

2.5 Number Portability

As indicated in Section 2.1, we appear to be stuck with identifiers tied to DNS names. While this is tolerable for names associated with organizations, as the validity of the name is often tied to the function performed by the individual addressed, this is less desirable for residential end users. With the use in Internet telephony, it ties consumers even more closely to a particular network service provider. This is a step backwards compared to the emerging number portability in the PSTN, where a customer can take her existing phone number to any local service provider, and, in the future, may even be able to keep her number when moving across large geographic distances (e.g., within a country or a supernational organization such as the European Union). There seem to be only two possible solutions: users acquire their own domain names, for example in top-level domains specifically geared towards personal names, or a purely random identifier, similar in principle to telephone numbers or social security numbers, is introduced, with a mapping to more evanescent domain names.

Directory services can, to some extent, mask changes in identifier, but for common names require a large amount of disambiguiting information that may raise privacy concerns. (However, these may be partially addressed by allowing searches for a particular combination of properties, without actually displaying anything but the identifier and the search criteria in the response.)

Number portability for E.164 numbers can be made easier by the ENUM service (see Section **??**), where the ENUM entry maps one phone number to another. However, this removes the distinction between dialed and ported ("physical") numbers and could lead to a chain of translations, resulting in inefficient use of the scarce numbering resource.

2.6 Mobility

The general problem of mobility can be described along several dimensions [36, 37, 38, 39]:

Roaming users: Roaming users connect to different network attachment points, but do not need to communicate while in transit. Unless the roaming user acts as a server, existing Internet mechanism are sufficient: DHCP for acquiring new addresses, appropriate authentication for any visited network and SIP registration for binding the current location to a permanent application-layer identifier.

Terminal mobility: Terminal mobility allows a terminal to maintain connectivity while changing network attachment points. Depending on the level of transparency desired, terminal mobility can be implemented at either the network or application layer [39]. At the network layer, mobile IP is the standard solution, with open issues primarily having to do with AAA and low-latency hand offs. Application-layer mobility redirects existing streams to the new address, e.g., using SIP INVITE requests or RTSP REDIRECT requests. Application layer mobility requires changes in applications, but less cooperation from the network. Since many of the facilities for application-layer mobility are useful for general robustness (e.g., ftp restart, HTTP retry with range requests), it may serve as a backstop where network-layer mobility is not available.

Personal mobility: Personal mobility allows a user to maintain a single identity even when changing terminals. Also, it should be possible to have a many-to-one mapping from terminals to identities, so that all these terminals are considered suitable end points representing the identity. SIP addresses this aspect by the use of a global identifier under which any number of end systems can register. For example, tel:+1-415-555-9875, sip:alice@homeisp.com and mailto:alice@work.com, can all register as being represented by sip:alice@example.com.

Session mobility: Session mobility makes it possible for an active session to be moved to a different terminal. For example, a user may want to move an on-going call from a wireless device to a wired device when entering her office. SIP supports this by sending another INVITE request in mid-session, updating the end point information. However, call transfer [35] may be the more appropriate mechanism here.

Service mobility: Service mobility maintains the same set of services regardless of where the end point is located. For example, the same set of service-handling scripts, address books, and configuration information should be available regardless of the network attachment point or the end system used. Service mobility currently only works for scripting for inbound calls, with the remainder needing further work.

2.7 Quality of Service

Given that high-quality landline telephone services have become cheap, there is little incentive to replace it with Internet telephony unless Internet telephony can offer comparable quality of service (and additional functionality).

The separation of session and resource setup in VoIP, while otherwise advantageous, has the potential to cause call defects if only one of the two succeeds [40]. There are two choices: reserve network resources first or set up the session first. Setting up the session first and then reserve resources could cause the phone to ring, but then the call fails due to lack of resources. Setting up resources first is difficult since the IP destination address is generally known only after the session has been set up. Also, it is helpful to the data source to know the receive capabilities of the data sink, in order to set the appropriate reservation parameters. (This is less of an issue with receiver-based reservation, where the receiver can match the requested rate in the session description to that in the reservation request.)

Thus, it has been proposed [40] to interleave resource reservation and session setup, so that resource reservation happens after the callee has been contacted, but ringing is delayed until resource availability has been established. The mechanism proposed also addresses the issue of "split" reservations, where each side reserves resources locally (e.g., for the access link), but without an end-to-end reservation exchange. Unfortunately, such QoS-assured calls incur additional call setup delay and significantly increase the number of messages needed to set up a call.

Alternatively, session and resource setup can proceed in parallel. In the unlikely case that resource reservation fails, the caller or callee can be given the choice to continue the call at best-effort quality or terminate the call. The acceptability of such a decoupled operation depends on the probability of reservation failures and the quality available when using best-effort services.

2.8 Authentication, Authorization and Accounting (AAA)

One of the areas with the largest architectural uncertainties is the issue of AAA for VoIP services. It is beyond the scope of this survey to delve into details, but some of the questions are summarized below. It seems generally agreed that any VoIP-related resource reservations could make use of whatever AAA mechanism and protocols are available. (However, the architecture for authentication, authorization and accounting across multiple domains remains largely unexplored.)

However, special considerations apply for some VoIP services, since access to a number of additional resources, beyond just network transmission, need to be controlled, including

- Internet telephony gateways;

- special services, such as the equivalent of fee-based telephone services, e.g., 900-number services in the U.S. or "*traffic" in mobile networks;
- call services, such as call filtering, authentication, logging and anonymity;
- media recording, playback and storage ("unified messaging");

These services will often reside outside the user's home network. However, it is likely that current web-based registration will be sufficient for many of the point services.

An additional complication is that authorization for some transport services may depend on the destination dialed, for example, allowing emergency and operator calls.

Also, any AAA system needs to take into account that either caller or callee may authorize access to resources for the other side, to allow services similar to reversed charges, free-phone numbers and the like.

- Is it desirable to use AAA at all for the session setup part? It has been suggested that instead of having a SIP proxy server generate a AAA request by translating session setup parameters into the AAA protocol format, that it instead simply forward the SIP request itself to the entity that can make policy decisions.
- How are session-layer identities, such as SIP URIs and network access identifiers (NAI, [41]), coupled? (Both identifiers have roughly the same syntax, although escaping and the set of non-alphanumeric characters allowed differ.) Is the NAI carried in resource reservation requests or session setup requests and can its realm be used by visited networks to identify the home network?

It has been suggested [42] to use the Open Settlement Protocol (OSP) to exchange authentication and accounting information with a central clearinghouse.

2.9 Conferencing

Conferencing and Internet telephony are closely related, although the emphasis in conferencing tends to be on larger groups collaborating using multiple media, while Internet telephony stresses two- and three-party calls where audio plays the dominant role. However, with the emerging IETF Internet telephony architecture, the boundary between the two is fluid, as all protocols discussed here support sessions with multiple media and multiple participants.

There remain two primary axes of describing sessions, namely how session members become part of a session, through announcement or explicit invitation, and how media data is distributed. Internet telephony focuses on explicit invitation of individuals, but combinations such as inviting members to an announcement-based session are easily implemented, as described above. Announcement-based sessions generally do not attempt to maintain a fully accurate roster of participants, leading to the designation as a light-weight or loose conference model [43]. Only servers managing all membership changes via a central server can guarantee that membership information is always accurate.

Data distribution can take place in a number of ways [44]: In the simplest case, a host replicates or mixes media streams. For RTP streams, it acts as a translator or mixer. (Generally, audio streams are mixed and video streams replicated, but graphical composition is feasible for video and audio may need to be replicated if the host does not have access to the encryption key or cannot decode the incoming audio stream.) This host can either be a designated entity or a conference participant ("end system mixing"). Mixers can also be arranged in a hierarchy or mesh, approximating application-layer versions of multicast routing. However, in the absence of a routing protocol, such setup is likely to be manual, suboptimal and error prone. Simple topologies, where a mixer is located on each side of an expensive link such as a transoceanic fiber, may however be manageable and nearly as effective as network-layer multicast. In addition, application-layer mixing avoids the problem that a single receiver may receive multiple simultaneous media streams, possibly overloading access links and thus rendering all streams unintelligible.

Scaling of this architecture is limited by the CPU resources required to encode and decode media streams, as well as egress bandwidth. Instead of single system handling media replication, each source of data can also directly send copies of the media via unicast or xcast (see below) to each participant.

There are three basic mechanism for explicit session setup: Each participant can explicitly set up a session with all other participants ("mesh"), the mixer or single controller can invite participants (in telephony, known as "dial-out") or the participants can invite themselves to the mixer ("dial-in"). Dial-in and dial-out require no extensions to SIP or MGCP/Megaco, while mesh extensions for SIP are under discussion. (See also Section 2.13.)

Table 1 summarizes the possible combinations for media distribution and signaling. Combinations of session types are possible; for example, some users may be connected to the conference via a mixer which also multicasts data to the remaining participants.

While small conferences are generally best considered symmetric, with each participant either taking turns on short notice, for audio, or a set of participants sending video. (Often, in video-follows-audio mode, video is sent from recent speakers, possibly enhanced with low-frame-rate video from others, to convey their stage of wakefulness.)

data distribution	mesh	session setup mixer invites	part. invite
multicast	yes	yes	N/A
mixer	N/A	dial-out	dial-in
mesh	yes	N/A	yes

Figure 1: Conferencing models

Recently, difficulties in deploying classical IP multicast across wide-area networks have led to the development of single-source multicast (SSM). While SSM is well-suited to content-distribution, it is less appropriate for conferences where activity can shift rapidly between sources. Thus, unless multicast trees can be set up in a matter of a few hundred milliseconds on demand, SSM can only be used by having all participants send media to a single host, which then multicasts it to the group.

Explicit or small-group multicast (xcast or SGM) [45], which enumerates destination addresses, may be well-suited for modest-sized groups. (For example, [46] is limited to nine destinations.) Since applications will not always be able to predict the maximum group size, systems will incur complexity if they need the ability to transition from small-group multicast to more scalable versions. However, this problem is similar to those faced by the mesh and end-system mixing solutions described above. Combinations are feasible, where the mixer or end system sends several xcast packets. Given the high relative packet header overhead for packet audio, reducing the packet count via SGM is particularly beneficial for packet voice.

2.10 Call-by-Call Provider Selection

In the PSTN, most countries that have deregulated their telecommunications sector offer consumers the ability to choose a long-distance provider on a call-by-call basis, typically by dialing a prefix, e.g., 10 in the United States. Indeed, in some countries, a large fraction of all calls are made in this fashion. This system works primarily because it defines the local network as the interconnection point where any local carrier can communicate with any long-distance carrier. It is unlikely that such a simple architecture is applicable to the Internet.

For IP-connected hosts, there are several approaches to approximate carrier selection:

Global routing: Larger organizations can acquire an AS number and a globally routed IP address block. This does not allow per-user carrier selection, but does support at least approximate choices of carriers for the organization on timescales of tens of minutes, due to BGP convergence times.

Multiple IP addresses: Each device is assigned multiple IP addresses, obtained from different carriers; calls are routed through the appropriate interface. This approach is more manageable for IP telephony, since the binding of addresses to external identifiers via SIP registration hides the change of IP addresses, without involving dynamic DNS. The sequential use of multiple dial-up providers is a variation on this concept. The need for large number of IP addresses makes this concept applicable to IPv6 only.

Network address translation: End systems use a constant local address, which is then translated by the NAT into one of several carrier-specific addresses. This approach conserves addresses, but has the usual NAT problems [47].

Note that this only influences how packets are routed *to* the caller, not which carrier is used to transport data to the callee. If volume-based charging were to be implemented for QOS-assured traffic, some form of carrier selection may well be required, if only to prevent effectively random charges depending on the callee.

For SIP services, having multiple incoming or outgoing *signaling* service providers is relatively easy, simply by maintaining several different external identifiers, in a manner similar to how users currently maintain several different email providers.

2.11 Phone Spam or Unsolicited Commercial Phone Calls

See Security Considerations section.

2.12 Control of Multimedia End Systems

Most commercial implementations of multimedia systems are monolithic, i.e., a single application deals with signaling and media. It also provides the user interface for all these components. As an alternative, a number of systems have been built [48, 49] where media and control are separated into separate components (processes), typically running on the same host. Each media type typically runs on its own media agent. These components communicate with each other via a local RPC mechanism or host-internal multicast, called a conference bus [50]. Multicast communication makes it easy to implement communication mechanisms based on subscribe-notify, i.e., where a number of entities all are informed of status changes. This approach has the advantage that it is easier to add new media types to conferences and to upgrade individual media components without changing the basic signaling and user interface aspects. To be equivalent in functionality to monolithic components, care has to be taken that all functionality of a media agent can be changed both when initially setting up the call and when changing media or network properties in mid-call. It would be desirable to allow new media agents to advertise their controllable parameters, so that user interfaces can automatically provide the appropriate controls or without having to provide a separate per-media interface.

A similar, but more general, control mechanism is found in the master-slave control protocols such as MGCP and Megaco. Here, a signaling process receives and generates signaling information, and then issues MGCP/Megaco requests to media agents. Since MGCP/Megaco are unicast-based, a discovery mechanism is needed so that the controller can detect when new components are available and how they are reachable. MGCP and Megaco are designed for use in a local area network, not necessarily on a single host, so they offer the necessary reliability mechanisms, but also use default ports, making running several instances on a host more tedious. Some of the issues are similar to those for conference buses, e.g., the ability to discover the controllable parameters of new media agents.

2.13 Conference Control

There is currently no IETF-standardized mechanism for "conference control". Part of the problem is that this service is ill-defined. Generally, it is taken to include floor control, including chair selection, and membership management. Floor control controls access to a single shared resource such as the audio channel, a common pointer or a token to allow changing a shared resource. Generally, conferences with such mechanisms are referred to as tightly-controlled sessions [43], as opposed to the loosely controlled multicast sessions pioneered in the Mbone.

Floor control may be simplified by building upon a reliable multicast mechanism [51]. If floor access is handled by a distinguished participant rather than in some mechanical, say, first-come-first-served, order, a mechanism for chair selection needs to be included, including a mechanism to elect a new chair if the old one has been disconnected from the network and thus cannot hand off control to her successor.

Membership management includes the ability to determine accurately the set of participants and to include or exclude certain members. For multicast sessions, RTCP and its extensions to large groups [52] provides a probabilistic estimate of the group membership. The maximum group size is not limited, but the learning rate is constrained to about four members a second for a conference with a media rate of 64 kb/s. Active (sending) participants are identified at a much higher rate, so that this may not be a problem in practice. If higher bandwidths are tolerable to gain membership information, either reliable multicast or central-server-based approaches may be used, but no such mechanism has been specified so far. For moderate group sizes and sessions mixed by a central server (MCU), the MCU could send out event notifications to let participants know that new users have arrived. In addition, normal RTCP sender reports can include this information.

For groups of modest size, say, up to a few dozen participants, it may be feasible to build a full mesh between all participants, flooding membership information in a manner similar to OSPF link-state flooding.

One of the missing IETF protocol pieces for multimedia conferences is the ability to share workstation applications, improving the rather inflexible service of T.120 [53]. It may be possible to build upon systems such as VNC [54].

2.14 Emergency Services

One of primary motivations for subscribing to telephone services is the ability to summon emergency help. In the PSTN, enhanced emergency (E911) systems provide three components: a common, widely known emergency number, a network-internal mapping function that translates this number to the nearest public safety answering point (PSAP) and finally a mechanism that allows this PSAP to determine reliably where the caller is located, even if the caller is too confused or young to provide this information verbally. As discussed in more detail in [55], current Internet protocols are not equipped to provide this functionality. Note that the issues only arise if there are IP terminals, not if IP is used as a trunking transport mechanism.

There are three basic approaches to providing emergency services to Internet telephones:

Local gateway: The simplest approach is that all emergency calls terminate on a PSTN gateway that is close to the caller, e.g., in the same building or neighborhood. As long as the gateway can insert the appropriate caller identification, the PSAP will treat this call like any other emergency call. If the number is simply the modem pool number, the ambulance will show up at the POP instead of at the home of the person making the emergency call. There are cases when this fails; for example, with VPNs, a person could be working at home, but appear to be part of the corporate network.

Central call center: In so-called telematics applications, cars are equipped with mobile phones that call a designated call center if, for example, the air bag inflates. The call center determines the location of the caller and then contacts the appropriate emergency call center via a normal phone call. This approach could also work for IP phones, at greater cost and increased call setup delay.

IP-enabled PSAPs: Longer term, PSAPs may themselves be able to handle IP-based emergency calls. This offers additional functionality, such as multimedia calls, telemetry, integration with web content and simpler use by the hearing-impaired.

It is likely that widespread use of Internet telephony cannot happen until at least basic emergency calling functionality is offered. However, the two basic services of service location and user location information are also useful for a wide variety of non-emergency services. On the other hand, the privacy considerations that apply for location-based services are not relevant for emergency calls.

Emergency services should be available across communication mechanism, including email, IRC, instant messaging and voice communication.

As a first step, it has been proposed [56] to define a common SIP address, “sip:sos@domain” as the common emergency identifier, in addition to local emergency numbers such as “tel:911” or “tel:112”.

A general wide-area service for determining the nearest PSAP is needed, based on the caller’s geographic location, as calls may be handled by the home proxy instead of in the network visited by a SIP terminal. The difficulty depends primarily on whether a replicated database can be used, e.g., LDAP or whois, or a hierarchical delegation mechanisms needs to be used. Delegation would likely need to be done according to civil boundaries instead of geographic location. Current civil coordinates, such as city and street address, may however not be known to the caller requesting this service.

A separate issue is the use of Internet telephony for emergency communication, e.g., in case of natural disasters. Here, the emphasis is primarily on prioritizing emergency communications traffic, regardless of the type of information (audio, video, data) carried. For voice and multimedia communications, the existing Priority header field in SIP can be extended as needed to conform to traditional PSTN naming.

3 Acknowledgements

Brian Carpenter, Gonzalo Camarillo and Jonathan Lennox provided detailed comments. The discussion of the IESG and IAB influenced the description of SIP and TRIP.

References

[1] C. Labovitz, A. Ahuja, R. Wattenhofer, and S. Venkatachary, "The impact of internet policy and topology on delayed routing convergence," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, (Anchorage, Alaska), Apr. 2001.

[2] H. Schulzrinne, "DHCP option for SIP servers," Internet Draft, Internet Engineering Task Force, Mar. 2002. Work in progress.

[3] A. Williams, "Zeroconf IP host requirements," Internet Draft, Internet Engineering Task Force, May 2002. Work in progress.

[4] K. Sollins, "The TFTP protocol (revision 2)," RFC 1350, Internet Engineering Task Force, July 1992.

[5] J. Kuthan and J. Rosenberg, "Firewall control protocol framework and requirements," Internet Draft, Internet Engineering Task Force, Nov. 2000. Work in progress.

[6] T. Hain, "Architectural implications of NAT," RFC 2993, Internet Engineering Task Force, Nov. 2000.

[7] J. Rosenberg, J. Weinberger, and H. Schulzrinne, "NAT friendly SIP," Internet Draft, Internet Engineering Task Force, July 2001. Work in progress.

[8] J. Rosenberg, L. Qiu, and H. Schulzrinne, "Integrating packet FEC into adaptive voice playout buffer algorithms on the Internet," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, (Tel Aviv, Israel), Mar. 2000.

[9] International Telecommunication Union (ITU), "Transmission systems and media, general recommendation on the transmission quality for an entire international telephone connection; one-way transmission time," Recommendation G.114, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Mar. 1993.

[10] G. Almes, S. Kalidindi, and M. Zekauskas, "A round-trip delay metric for IPPM," RFC 2681, Internet Engineering Task Force, Sept. 1999.

[11] G. Almes, S. Kalidindi, and M. Zekauskas, "A one-way delay metric for IPPM," RFC 2679, Internet Engineering Task Force, Sept. 1999.

[12] M. Mathis and M. Allman, "A framework for defining empirical bulk transfer capacity metrics," Internet Draft, Internet Engineering Task Force, Apr. 2001. Work in progress.

[13] G. Almes, S. Kalidindi, and M. Zekauskas, "A one-way packet loss metric for IPPM," RFC 2680, Internet Engineering Task Force, Sept. 1999.

[14] R. Koodli and R. Ravikanth, "One-way loss pattern sample metrics," Internet Draft, Internet Engineering Task Force, Apr. 2002. Work in progress.

[15] C. Perkins and O. Hodson, “Options for repair of streaming media,” RFC 2354, Internet Engineering Task Force, June 1998.

[16] J. Rosenberg and H. Schulzrinne, “An RTP payload format for generic forward error correction,” RFC 2733, Internet Engineering Task Force, Dec. 1999.

[17] D. Crocker, “Mailbox names for common services, roles and functions,” RFC 2142, Internet Engineering Task Force, May 1997.

[18] B. Campbell and R. Sparks, “Control of service context using SIP request-URI,” Internet Draft, Internet Engineering Task Force, Feb. 2001. Work in progress.

[19] S. Sun, S. Reilly, L. Lannom, and J. Shi, “Handle system protocol (ver 2.0) specification,” Internet Draft, Internet Engineering Task Force, Apr. 2001. Work in progress.

[20] N. Popp, M. Mealling, and M. Moseley, “Common name resolution protocol (CNRP),” Internet Draft, Internet Engineering Task Force, Feb. 2002. Work in progress.

[21] M. Smith, “Definition of the inetOrgPerson LDAP object class,” RFC 2798, Internet Engineering Task Force, Apr. 2000.

[22] M. Wahl, “A summary of the X.500(96) user schema for use with LDAPv3,” RFC 2256, Internet Engineering Task Force, Dec. 1997.

[23] P. Hoffman, “The rescap resolution protocol,” Internet Draft, Internet Engineering Task Force, June 1999. Work in progress.

[24] W. Yeong, T. Howes, and S. Kille, “Lightweight directory access protocol,” RFC 1777, Internet Engineering Task Force, Mar. 1995.

[25] M. Handley and V. Jacobson, “SDP: session description protocol,” RFC 2327, Internet Engineering Task Force, Apr. 1998.

[26] G. Camarillo, J. Holler, G. Eriksson, and H. Schulzrinne, “Grouping of m lines in SDP,” Internet Draft, Internet Engineering Task Force, Feb. 2002. Work in progress.

[27] D. Kutscher, J. Ott, and C. Bormann, “Session description and capability negotiation,” Internet Draft, Internet Engineering Task Force, Mar. 2002. Work in progress.

[28] P. Hoschka, “Synchronized multimedia integration language (SMIL) 1.0 specification,” W3C Recommendation REC-smil-19980615, World Wide Web Consortium (W3C), June 1998. Available at http://www.w3.org/TR/REC-smil/.

[29] C. Newman and J. G. Myers, “ACAP – application configuration access protocol,” RFC 2244, Internet Engineering Task Force, Nov. 1997.

[30] D. Petrie, “A framework for SIP user agent configuration,” Internet Draft, Internet Engineering Task Force, Nov. 2001. Work in progress.

[31] J. Lennox and H. Schulzrinne, “Feature interaction in Internet telephony,” in *Proc. of Feature Interaction in Telecommunications and Software Systems VI*, (Glasgow, United Kingdom), May 2000.

[32] F. P. Duffy and R. A. Mercer, "A study of network performance and customer behavior during-direct-distance-dialing call attempts in the USA," *Bell System Technical Journal*, vol. 57, no. 1, pp. 1–33, 1978.

[33] M. Day, J. Rosenberg, and H. Sugano, "A model for presence and instant messaging," RFC 2778, Internet Engineering Task Force, Feb. 2000.

[34] R. Mahy and I. Slain, "SIP extensions for message waiting indication," Internet Draft, Internet Engineering Task Force, July 2001. Work in progress.

[35] R. Sparks, "SIP call control - transfer," Internet Draft, Internet Engineering Task Force, July 2001. Work in progress.

[36] J. G. Hemmady, J. R. Maymir, and D. J. Meyers, "Network evolution to support personal communications services," in *globecom*, (San Francisco, California), pp. 710–714, Nov. 1994.

[37] P. E. Wirth, "Teletraffic implications of database architectures in mobile and personal communications," *Computer Networks*, vol. 28, pp. 613–618, Mar. 1996.

[38] V. K. Garg, E. L. Sneed, and W. E. Gooding, "Subscriber data management in personal communications services networks," *IEEE Personal Communications Magazine*, vol. 4, June 1997.

[39] H. Schulzrinne and E. Wedlund, "Application-layer mobility using SIP," *Mobile Computing and Communications Review*, vol. 4, pp. 47–57, July 2000.

[40] W. Marshall, G. Camarillo, and J. Rosenberg, "Integration of resource management and SIP," Internet Draft, Internet Engineering Task Force, Apr. 2002. Work in progress.

[41] B. Aboba and M. Beadles, "The network access identifier," RFC 2486, Internet Engineering Task Force, Jan. 1999.

[42] H. Sinnreich, D. Rawlins, A. Johnston, S. Donovan, and S. Thomas, "AAA usage for IP telephony with QoS," Internet Draft, Internet Engineering Task Force, July 2000. Work in progress.

[43] M. Handley, J. Crowcroft, C. Bormann, and J. Ott, "The internet multimedia conferencing architecture," Internet Draft, Internet Engineering Task Force, July 2000. Work in progress.

[44] J. Rosenberg and H. Schulzrinne, "Models for multi party conferencing in SIP," Internet Draft, Internet Engineering Task Force, July 2001. Work in progress.

[45] D. Ooms, "Taxonomy of xcast/sgm proposals," Internet Draft, Internet Engineering Task Force, July 2000. Work in progress.

[46] D. Helder and S. Jamin, "IPv4 option for somecast," Internet Draft, Internet Engineering Task Force, July 2000. Work in progress.

[47] M. Holdrege and P. Srisuresh, "Protocol complications with the IP network address translator," RFC 3027, Internet Engineering Task Force, Jan. 2001.

[48] H. Schulzrinne, "A comprehensive multimedia control architecture for the Internet," in *Proc. International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, (St. Louis, Missouri), May 1997.

[49] S. McCanne and V. Jacobson, “vic: A flexible framework for packet video,” in *Proc. of ACM Multimedia ’95*, Nov. 1995.

[50] J. Ott, C. Perkins, and D. Kutscher, “Requirements for local conference control,” Internet Draft, Internet Engineering Task Force, Dec. 1999. Work in progress.

[51] A. Mankin, A. Romanow, S. Bradner, and V. Paxson, “IETF criteria for evaluating reliable multicast transport and application protocols,” RFC 2357, Internet Engineering Task Force, June 1998.

[52] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A transport protocol for real-time applications,” Internet Draft, Internet Engineering Task Force, Nov. 2001. Work in progress.

[53] International Telecommunication Union, “Data protocols for multimedia conferencing,” Recommendation T.120, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, July 1996.

[54] T. Richardson, Q. Stafford-Fraser, K. R. Wood, and A. Hopper, “Virtual network computing,” *IEEE Internet Computing*, vol. 2, pp. 33–38, January/February 1998.

[55] H. Schulzrinne, “Providing emergency call services for SIP-based internet telephony,” Internet Draft, Internet Engineering Task Force, Mar. 2001. Work in progress.

[56] H. Schulzrinne, “Universal emergency address for SIP-based internet telephony,” Internet Draft, Internet Engineering Task Force, Feb. 2002. Work in progress.