INTERNET DRAFT
draft-ohta-notasip-04.txt

Masataka Ohta
Tokyo Institute of Technology
Kenji Fujikawa
Kyoto University

12 October 2001

Nothing Other Than A Simple Internet Phone (NOTASIP)

Status of this Memo

Abstract

This memo describes a simple protocol for port negotiation of bi-directional UDP steam. The primary target of the application of the protocol is Internet phone without QoS guarantee and the protocol is explained with Internet phone as the application using telephone terminology. A callee's host may accept multiple streams on a well known UDP port and redirect them to other ports.

1. The Architectural Principle of the Internet Phone

Fortunately enough, the Internet is the installed base of data communications. Other network technologies must find some way to interoperate with the Internet in order to survive a little longer.

However, in the world of phone communication today, POTS (Plain Old

Telephone Service) is the installed base. For the Internet to replace POTS within a few years, it is important that Internet telephony interoperate with POTS.

So, the primary requirement for Internet telephony at this early stage is that it should be able to interoperate with a dumb analog phone, which constitutes the installed base.

Historically, telephone companies in different nations have tried hard to ensure that their systems do not interoperate smoothly to protect their market. None the less, or as a result, protocols to interoperate POTS are well developed. The protocols must be constructed over voice, the only common transport over different phone systems. A notable protocol is the operator assisted call. However, as human intervention costs a lot, most POTS support tone dialing capability as a minimal digital communication capability.

Note that the complex capabilities of digital phones, which are disappearing, have nothing to do with the installed base and are ignored in this memo.

Possible complex capabilities of Internet phones such as multiparty teleconferencing, which are hard to operate over voice, are also ignored in this memo. It should be noted that there are multiparty teleconferencing services already available through POTS, simulation of which over Internet telephony is simple without using a complex Internet protocol.

POTS is the installed base worth considering. As it is difficult for a human being to generate or recognize IP packets over POTS with voice or dial tones, protocols that need a complex exchange of IP packets should be considered seriously only after we don't have to interoperate with POTS.

It is assumed that the operating system supports the notion of a connected UDP socket [UNIX].

While there are some protocol to partition a telephone device into a media gateway and a call agent (or a media controller), it is an obvious violation of the end to end architectural principle of the Internet. With NOTASIP, a telephone device or a telephone exchanger is an end system, a usual Internet host, with its own policy and control mechanism within it.

2. Caller Initiates the Call

The caller's host somehow (for example, through URL in callee's Web

page, as shown in Appendix A) finds the callee's IP address, UDP port number (with default port number of <to be assigned by IANA>) and desired encoding.

The caller's host opens a UDP socket and starts sending properly encoded UDP packets of voice.

3. Callee Accepts the Call

The callee's host receives a UDP packet from someone, and it opens a new connected UDP socket to the callers UDP port using a new source port number (chosen randomly) and the same IP address as the destination address of caller's packet.

The callee's host, then, start ringing the phone to notify the existence of a call to the callee. The ring back tone should be sent to the caller.

If the callee's host do not want to accept a simultaneous call, it may suspend the UDP port used to accept the call. Then, if the port is already connected to someone else, an ICMP port unreachable is returned, which makes the caller's host generate a busy tone to the caller.

4. Connection Established

If the callee takes his phone off hook, the callee's host should send the voice of the callee to the caller's host. The caller host receives a first packet and, confirming the source IP address of the packet is callee host's, connects its UDP port to the callee's sending port and the connection is established at the caller.

Then, the connection is established at the callee, if the callee receives a packet at the new socket.

It should be noted that the connection is established immediately after a three way exchange of packets between the caller and the callee, even if there are some packet losses.

5. Call Termination

To terminate the call, the caller's or callee's host closes the socket.

The same port number should not be used again until 256 (maximum IPv4

TTL) + 30 seconds passes.

6. Interoperation with PSTN

Interoperation with PSTN is performed over voice, the only common transport, with operator assistance, dial tone or anything. The exact protocol over the voice on natural language to/from telephone operators or on DTMF to/from server computers is service provider dependent and inappropriate for standardization.

7. Error Conditions

If the connected UDP socket cannot be created or the socket generates an error of ICMP port or protocol unreachable, the call terminates.

ICMP host or network unreachable should be ignored as a soft error.

If the callee's host receives a UDP packet on a well known UDP port from someone other than the callee's host or from the callee's host with source UDP port different from the original, it should be processed as a request of new connection.

If the callee's host receives a UDP packet on a new UDP port from someone other than the callee or from the callee with source UDP port different from the original, it should be ignored.

If the caller's host receives a UDP packet from someone other than the callee, it should be ignored.

If the caller's host receives a UDP packet from the callee's host with a source UDP port different from the original before a connection is established, it is a normal case to establish the connection.

If the caller's host receives a UDP packet from the callee's host with a source UDP port different from the original after the connection is established, it should be ignored.

If there is no valid packets received on a port for 30 seconds, the call terminates.

8. Portability and Mobility

We discussed the way to call stationary hosts, in other words, hosts that do not support portability or mobility in the above. Now we will show how to call a particular host independent of its location

and how to call the host that a called user specifies to receive a call.

According to the terminology of the IP mobility WG, portability means limited mobility that allows the relocation of hosts that destroys existing connections.

Among several methods described in this section, only IP mobility allows the real mobility to allow relocation of hosts with all of the connections kept alive. That is, IP mobility, though not so popular today, is the way to go.

The following discussions show that there is no need to develop new protocols for portability and mobility in Internet telephony.

8.1. IP Mobility

IP mobility[RFC2002], which provides mobility on the L3 level, simply implements mobility in Internet telephony. The mobility in IP mobility enables a host to use the same IP address wherever it is located.

Of course, a host can also use the same IP address location-independently when the lower layer, i.e. L2, provides mobility (e.g. dial-up PPP using mobile phones). In this case, Internet telephony mobility can be easily achieved, although this may cost more.

8.2. DNS Update

Generally, a caller specifies a callee not by the callee's direct IP address but by the callee's DNS name, in or not in a URL. DNS dynamic update[RFC2136, RFC2137] attains the portability, though not all name servers support it.

8.3. Web Update

A user comes to know the callee's URL by looking at the callee's Web. Thus, dynamic updates of URLs in the calle's Web lead to portability in Internet telephony. CGI or Java is sufficient, so nothing is required to be standardized.

8.4. Forwarder

A forwarder runs on the host on which a user usually receives calls and forwards packets to another host statically specified by the user. (placing a file like ".forward" in his/her home directory). The forwarder applications can be written in a 10-to-20 line C program. Standardization is not required here either.

9. References

[RFC2022] Perkins C., ``IP Mobility Support,'' RFC 2022, October 1996.

[RFC2136] Vixie, P., Thomson, S., Rekhter, Y. and Bound, J., ``Dynamic Updates in the Domain Name System (DNS UPDATE),'' RFC 2136, April 1997.

[RFC2137] Eastlake, D., ``Secure Domain Name System Dynamic Update,'' RFC 2137, April 1997.

[UNIX] See UNIX manuals.

[IPHONE] FUJIKAWA K., and OHTA M., ``IPhone URL,'' Internet Draft draft-fujikawa-iphone-url-01.txt (work in progress), October 2001.

[DIR] KITAGAWA T., and FUJIKAWA K., ``Simple Directory Server for IP Telephony,'' Internet Draft draft-fujikawa-iptel-directory-00.txt (work in progress), October 2001.

10. Security Considerations

The security of POTS accounting is often based on 4 digit password or plain credit number and is quite poor. Moreover, it is, in general, impossible to know the phone number of the caller. However these are the accepted security of the phone system.

Best effort Internet phone is basically free (except for a flat rate portion) that no serious security consideration is necessary as a phone system. A possible denial of service attack can be based on forged caller source IP address, but is a lot more harmless than a similar attack through POTS with a unknown source telephone number.

How a portable/mobile node informs its location of its home in a secure manner is a serious problem in portability/mobility support. Security mechanisms in IP mobility, in DNS update or in Web update help without any modifications in NOTASIP.

With a forged source IP address in a request packet of a request-reply protocol, it is possible to use the server of the protcol for denial of service attack, which is unavoidable. However, with

NOTASIP, if a callee's host receives a packet with forged caller's IP address, the callee's host generates a reply stream of UDP packets to the address for 30 seconds, which badly amplifies the effectiveness of denial of service attack. Thus, before the connection is established and a packet is received to the new UDP port of the callee's host, the callee's host should generate at most one packet in the reply stream for each received packet from the caller.

Appendix A. How to Find Callee's Information

It is often misunderstood that, after callee's IP address is somehow known, callee's port number and encoding format must be known separately.

However, a mechanism to provide the IP address can often offer the port number and encoding format, too.

For example, if a call is initiated from a user browsing a web page, it is convenient if URL like this:

```
iphone://mohta.person.titech.ac.jp:10000/DVI4:8000
```

is embedded in a callee's home page. [IPHONE] From the URL, not only the IP address, but also the port number and encoding format can be obtained.

Thus, protocols to negotiate callee's port number and encoding format with callee's host is not so useful. Such protocols may be used in exceptional situations but should not be an integral part of telephone protocols.

Just as usual IP applications assume IP addresses or DNS names of the other side of communication are somehow provided, telephone applications should just assume that all the information to initiate calls is somehow provided.

Authors' Addresses

Masataka Ohta
Graduate School of Information Science and Engineering
Tokyo Institute of Technology
2-12-1, O-okayama, Meguro-ku, Tokyo 152-8552, JAPAN
Phone: +81-3-5734-3299
Fax: +81-3-5734-3299
EMail: mohta@necom830.hpcl.titech.ac.jp

Kenji Fujikawa
Graduate School of Informatics
Kyoto University
Yoshidahonmachi, Sakyo-Ku, Kyoto City, 606-8501, JAPAN
Phone: +81-75-753-5387
Fax: +81-75-751-0482
EMail: magician@kuis.kyoto-u.ac.jp