P2P-SIP BOF
Internet-Draft
Expires: August 31, 2006

J. Quittek
M. Stiemerling
T. Dietz
S. Niccolini
NEC

February 27, 2006

Problem Statement for SIP-signalled Peer-to-Peer Communication across Middleboxes
draft-quittek-p2p-sip-middlebox-00

Status of this Memo

Copyright Notice

Abstract

Middleboxes, particularly firewalls and network address translators, are essential components of today's Internet infrastructure. They are designed to support client-server applications well, but very often they are obstacles for peer-to-peer communication. This memo discusses middlebox traversal issues of SIP-based peer-to-peer

communication. Required communication steps are analyzed concerning their potential constraints caused by middleboxes. The requirements derived from this analysis are compared with the capabilites of currently available middlebox traversal methods and SIP signaling standards. The comparison identifies open issues that need to be considered when designing standards for a SIP-based peer-to-peer communication infrastructure.

Table of Contents

1. Introduction

Firewall and Network Address Translators (NATs) are essential components of today's Internet infrastructure. They belong to the class of middleboxes [RFC3234] and are widely used for protecting networks, connecting private networks to the public Internet, and further purposes. They are designed to support client-server applications well, but very often they are obstacles for peer-to-peer communication.

This memo analyzes middlebox traversal issues and problems of SIP-based peer-to-peer communication. The analysis concerns signaling across middleboxes as well as data streaming. It is conducted for each individual step that is expected to be required for SIP-based peer-to-peer communication. Required steps are derived from observation of existing peer-to-peer applications and from the general requirements for SIP-based peer-to-peer Internet telephony described in [I-D.baset-sipping-p2preq]. The analysis considers requirements for middlebox control as well as for SIP signaling required specifically for traversing middleboxes and it is performed for the terminal side as well as for the peer-to-peer infrastructure side.

The analysis in section Section 2 is followed by an overview of existing middlebox traversal techniques in section Section 3. Section Section 4 compares existing SIP capabilities and available common middlebox traversal methods against the requirements identified in section Section 2. The result describes four issues, three SIP-related, one SIP-unrelated, that should be considered when designing standards for a SIP-based peer-to-peer communication infrastructure.

2. Middlebox Issues of SIP-based Peer-to-Peer Communication

This section discusses steps of SIP-based peer-to-peer communication across middelboxes. Not all steps are required for all possible scenario variations and depending on design and implementation of SIP-based peer-to-peer communication infrastructures and terminals, the sequence of steps may be varied.

The steps discussed in the following subsections are

* middlebox detection,
* registration,
* search for relays,

* address lookup,
* call setup,
* call termination.

2.1. Step 1: Middlebox Detection

Detecting middleboxes in the signaling path or the data streaming path is a step that usually should be performed by a general SIP terminal each time it gets connected to an access network. The step is not required if the terminal receives this information by manual configurations or by interaction with a management system.

This step can be carried out explicitly and separately by using special protocols and/or servers designed and deployed for this particular purpose. But it can also be integrated in the steps described in the following subsections.

The information gathered by middlebox detection concerns available communication means and NAT parameters. Efficient methods may perform both checks at once.

Existing SIP terminals and peer-to-peer terminals for voice over IP use special protocols for this detection step, such as the STUN protocol [RFC3489]. The availability of detections techniques is discussed in section Section 3.3

2.1.1. Communication Means Detection

The terminal needs to detect the communication means that are available for

* registering with the peer-to-peer infrastructure,
* incoming and outgoing signaling within the peer-to-peer infrastructure,
* data streaming to and from other terminals or relays.

Depending on the requirements for the actions listed above, checks will include the availabilbity of

* sending and receiving UDP packets,
* the capability of opening incoming and outgoing TCP connections,
* the options to use certain fixed port numbers that are used within the peer-to-peer infrastructure,
* the option to relay or tunnel signaling messages and streamed data if the means for regular transport are too restrictive.

Many existing SIP solutions need UDP for signaling as well as for

data streaming. They do not need any support by TCP. However, this is expected to change. In order to protect SIP signaling from several kinds of attacks, SIP signaling over TLS [RFC2246] or other more secure protocols is currently considered. If the chosen secure solution is based on TCP or uses IPsec [RFC4301], then more than just UDP connectivity needs to be checked.

2.1.2. NAT Parameter Detection

If a NAT was detected between the terminal and the peer-to-peer infrastructure, then it may be necessary to explore the properties of the present network address translation. Particularly, the type of NAT and the assigned public IP address and port number need to be determined if a NAT is present.

Basic types of NATs are described in [RFC2663]. Further types are described in the terminology section of [RFC3489].

2.2. Step 2: Registration

Once the terminal has sufficient information about middleboxes between itself and other parts of the peer-to-peer infrastructure, it can register itself using a communication means that has proven to be available.

Typically, a registration would be performed as a client-server operation that is permitted by most NAT and firewall configurations. However, in highly restrictive environments, there might be a need for relaying or tunneling the registration, because direct access to registration servers or the agreed port number for this purpose is blocked. In such a case, the search for a relay that is described in the following section needs to be made before registering.

The registration step includes up to four actions

* authentication of the user
  The authentication can be integrated into SIP signaling or into another protocol that is used within the peer-to-peer infrastructure. In general, also peer-to-peer infrastructures without authentication could be used, but the practical use might be limited in today's Internet. It is recommendable that the user authenticates itself at the peer-to-peer infrastructure. However, in oder to increase security (and avoid potential man-in-the-middle attacks) it is desirable that the peer that receives the user authentication authenticates itself at the user.

* notification of communication capability and willingness
  The user need to notify the peer that receives its registration about its communication capabilities and its willingness to receive session invitations.
  If during the middlebox detection step it was found that a relay is needed for communication with other terminals, then it might be required for the registration to indicate that the terminal is not at all able to communicate. If this problem can be solved by connecting to a relay in step 3, the registration of communication capabilities can be updated.

* registration of contact parameters
  The user needs to register its contact parameters, such as IP address and port numbers at which it can receive invitations to calls. These values might depend on the result of the middlebox detection performed in step 1. If a relay is required for signaling or for data streaming, then this information cannot be provided before the relay has been found and connected.

* notification of service provisioning capability and willingness
  The user need to notify the peer that receives its registration about its capabilities and willingness to offer services to other users of the peer-to-peer infrastructure, such as authentication relaying, signaling relaying, lookup service provisioning, data stream relaying, etc.

These actions can be performed separately using different protocols, but in an efficient solution, they are all fully integrated into SIP signaling.

2.3. Step 3: Search and Connect Relay

In case the middlebox detection in step 1 indicated that there is a need for relaying signaling and/or data streaming, the terminal can request suggestions for or assignment of a relay of the peer-to-peer infrastructure In implementations this step can be integrated with the registration, but logically, it is a separate action.

There are several ways of finding a suited relay, but since relays are considered parts of the peer-to-peer infrastructure, the selection of the relay to use will probably be based on a list of suggested relays that the terminal receives from the peer at which it registered or via a lookup service that was made available by the registration. The actual choice of a relay may depend on the results of the middlebox detection in step 1.

The requirements for SIP-based peer-to-peer Internet telephony in

[I-D.baset-sipping-p2preq] suggest that a relay should be located such that the communication delay is not significantly increased. This may imply that different relays are chosen for calls to different other peers which requires a call-by-call search for a relay.

After a suited relay has been found, the terminal connects to it and receives from the relay the address parameters (IP address and port number) that the relay provides for the terminal. With this information, the terminal can update its initial registration and register itself as ready to receive invitations to sessions.

2.4. Step 4: Address Lookup

After registering with the peer-to-peer infrastructure and before intiating a call, the terminal needs to lookup addresses of concrete and potential callees. There are two common ways to do so: per call look-up of the callee's address or call-independent lookup of the addresses for a predefined "buddy list".

In both cases, the peer-to-peer infrastructure needs to be contacted and a lookup of a single address or a list of addresses needs to be initialized. Because of the client-server nature of this lookup procedure, it should be permitted by most NAT and firewall configurations. However, in case it is not, relaying is required also for this step.

2.5. Step 5: Connection Establishment and termination

After step 1-4 have been performed (as far as necessary), the terminal should be prepared to establish connections across the detected middleboxes either by using direct peer-to-peer communication or via a relay server.

When exchanging addresses for data streaming with another peer, addresses and port numbers provided by a NAT or relay need to be considered.

In some cases direct communication with middleboxes is required for this step. For example, if the terminal assumes that affected middleboxes support the NSIS NATFW NSLP [I-D.ietf-nsis-nslp-natfw], then signaling between terminal and firewall would be required at call establishment and at termination.

3. Middlebox Traversal Methods

This section discusses standardized and/or well known existing

middlebox traversal methods. They are grouped into tunneling methods, network-initiated methods and terminal-initiated methods.

Tunneling methods work around existing firewall and NAT configurations that intend to block certain kinds of traffic. Therefore, they should be used only, if there is no doubt that their use is appropriate and does not violate any service agreement.

Network-initiated methods use explicit signaling between a controlling entity in the network, such as a network or service management system or a "call agent", and one or more middleboxes.

Terminal-initiated methods use explicit or implicit signaling between the terminal and one or more middleboxes.

3.1. Tunneling

Tunneling is a highly controversial means for traversing firewalls and NATs. It is discussed, because it exists, is well known and - unfortunately - used by common applications for Internet telephony.

Most problematic is tunneling using well known port numbers used for other services, such as DNS and HTTP. Most commonly, these port numbers are open to be used for DNS and HTTP services. Using them for other purposes might be considered as illegal use of the network. But for obvious reasons, many tunneling tools concentrate on port numbers that are most commonly open.

A more legitimate tunneling technology is Traversal Using Relay NAT (TURN) [I-D.rosenberg-midcom-turn]. TURN does not mis-use fixed port numbers, but provide an network address translation for terminals behind a NAT that is not peer-to-peer-friendly. TURN is particularly helpful in the presence of symmetric NATs. But also if more than one terminals of a session are located behind a NAT, TURN can be very helpful. This case is described in more detail in [I-D.srisuresh-behave-p2p-state].

3.2. Network-initiated Middlebox Signaling

Network-initiated middlebox signaling is based on the assumption that there is an entity in the network that manages sessions. Such a system could be a service management system or a special call agent. This system directly controls middleboxes in order to enable sessions by using a control protocol, such as SNMP [RFC3410] and the MIDCOM MIB [I-D.ietf-midcom-mib] or the SIMCO protocol [I-D.stiemerling-midcom-simco]. Both provide sufficient means to an authenticated entity to configure the middlebox such that sessions can be established across them.

3.3. Terminal-initiated Middlebox Signaling

Terminal-initiated signaling is performed between the terminal and one or more middleboxes. Terminal-initiated signaling is suggested to be used for SIP-based Internet telephony by [I-D.baset-sipping-p2preq].

Signaling may be performed explicitly or implicitly. Explicit signaling requires specific support at the middleboxes for the used signaling protocol and method, for example, the NSIS NATFW NSLP [I-D.ietf-nsis-nslp-natfw], while implicit signaling configures the firewall rather by a side-effect as, for example, the STUN protocol [RFC3489] does it.

Network-initiated signaling methods exist for middlebox detection purposes as well as for middlebox configuration.

The following methods are discussed commonly used or currently under standardization:

* STUN
* UPnP
* SOCKS and RSIP
* NSIS

3.3.1. STUN

The Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) [RFC3489], also referred to as STUN, gives the terminal the ability to test NATs on the path for their UDP properties and to determine the terminal's publicly reachable IP address and port number. It is limited to UDP flows only. This technique is widely used by deployed SIP user agents to determine their reachability.

3.3.2. UPnP

UPnP [UPNP] is a link broadcast technique used to locate services in a small office/home office network. One of the services is home gateway detection, i.e., locating your home gateway/NAT. UPnP is well integrated into the Windows operating system and may home gateways. It is not well supported on other operating system platforms.

3.3.3. SOCKS and RSIP

SOCKS [RFC1928] enables authenticated NAT traversal for terminals. The protocol enables terminals located behind a single NAT to

authenticate itself and to request NAT traversal service. SOCKS has been partially deployed in the past, especially in enterprise networks, but is not so much used by today anymore. UDP support is very rudimentary at best.

RSIP [RFC3103] is a tunnel mechanism, where a RSIP-aware terminal can request allocation of a publicly reachable IP address and other parameters at RSIP-aware firewalls and NATs. The other parameters are UDP/TCP ports, IPsec SPIs, etc. A tunnel for the requested flow is created between the terminal and the NAT. As for SOCKS, RSIP supports only a single NAT on one side of the network.

3.3.4. NSIS NATFW NSLP

A recent protocol proposal for locating and configuring NATs and firewalls is the NAT and firewall NSIS Signaling Layer Protocol (NATFW NSLP) [I-D.ietf-nsis-nslp-natfw] defined by the IETF NSIS working group. NAT/firewall signaling shares a property known from Quality of Service (QoS) signaling. The signaling of both must reach any device on the data path that is involved in QoS or NATFW treatment of data packets. Therefore, the NATFW NSLP signaling follows the data path and configures any NATFW NSLP-enabled firewall. This protocol requires at least one side of the network to support NSLP signaling, but works end-to-end if supported by both ends.

4. Open Issues for SIP-based Peer-to-Peer Communication

This section lists open issues of middlebox traversal for SIP-based peer-to-peer communication. The issues are grouped into issues that concern SIP signaling and issues that concern middlebox traversal methods outside of SIP.

4.1. SIP-related Issues

This section describes three SIP-related issues of middlebox traversal. They concern missing capabilities of SIP for signaling

* terminal reachability,
* communication service requirements,
* communication service offers.

4.1.1. Terminal Reachability

The relevance of this open issue depends on design decisions concerning the registration and relay selection process. If a terminal in an restricted environment needs to register before it can select a relay, then it needs to express within its registration that

it is currently unreachable. Currently, the SIP protocol does not provide explicit means for signaling such a state.

4.1.2. Communication Service Requirement

A terminal in a restricted environment needs to be able to express its needs for communication services provided by other peers, such as relaying signaling messages, lookup requests, and/or data streams. Currently, the SIP protocol does not provide explicit means for signaling such requirements.

4.1.3. Communication Service Offering

A terminal in an unrestricted environment (or just slightly restricted) environment might be able (and the user willing) to offer services to other peers, such as relay services and lookup services. Currently, the SIP protocol does not provide explicit means for signaling such offers.

4.2. SIP-unrelated Issues

There is only one SIP-unrelated issue identified by this document. It concerns middlebox detection for protocols other than UDP.

4.2.1. Middleobox Detection Beyond UDP

As stated in section Section 2.1, it will probably not be sufficient for secure SIP-based peer-to-peer communication to just detect NAT properties for the UDP protocol, but TCP checks or further checks for IPsec will be necessary. Currently, there are no commonly known tools available for performing these checks.

5. Security considerations

Obviously, securing access to firewall and NAT configuration is extremely important for maintaining network security. Whatever means are applied for enabling SIP-based peer-to-peer communication across firewalls and NATs, it should be ensured that the security policy that was used for configuring the firewall and/or NAT is not violated by this action.

Particularly tunneling over port numbers that are assigned to other services, such as the HTTP or DNS port numbers can often be considered a violation of the policy that was set up by the access network operator. Furthermore, such tunnels may be mis-used by attackers as entry points into otherwise well protected network.

If explicit methods for firewall and NAT configuration are used, the security considerations section of the respective standards document should be considered.

6. Acknowledgements

Martin Stiemerling is partly funded by Ambient Networks, a research project supported by the European Commission under its Sixth Framework Program. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Ambient Networks project or the European Commission.

7. Informative References

[RFC1928] Leech, M., Ganis, M., Lee, Y., Kuris, R., Koblas, D., and L. Jones, "SOCKS Protocol Version 5", RFC 1928, March 1996.

[RFC2246] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, January 1999.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[RFC3103] Borella, M., Grabelsky, D., Lo, J., and K. Taniguchi, "Realm Specific IP: Protocol Specification", RFC 3103, October 2001.

[RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, February 2002.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC3489] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[I-D.rosenberg-midcom-turn]
Rosenberg, J., "Traversal Using Relay NAT (TURN)", draft-rosenberg-midcom-turn-08 (work in progress), September 2005.

[I-D.srisuresh-behave-p2p-state]
Srisuresh, P., "State of Peer-to-Peer(P2P) communication across Network Address Translators(NATs)", draft-srisuresh-behave-p2p-state-01 (work in progress), October 2005.

[I-D.baset-sipping-p2preq]
Baset, S., "Requirements for SIP-based Peer-to-Peer Internet Telephony", draft-baset-sipping-p2preq-00 (work in progress), November 2005.

[I-D.stiemerling-midcom-simco]
Stiemerling, M., "Simple Middlebox Configuration (SIMCO) Protocol Version 3.0", draft-stiemerling-midcom-simco-08 (work in progress), December 2005.

[I-D.ietf-midcom-mib]
Quittek, J., "Definitions of Managed Objects for Middlebox Communication", draft-ietf-midcom-mib-06 (work in progress), January 2006.

[I-D.ietf-nsis-nslp-natfw]
Stiemerling, M., "NAT/Firewall NSIS Signaling Layer Protocol (NSLP)", draft-ietf-nsis-nslp-natfw-09 (work in progress), February 2006.

[UPNP] UPNP Web Site, "Universal Plug and Play Web Site", Web Site http://www.upnp.org/, February 2006.

Authors' Addresses

Juergen Quittek
Network Laboratories, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 905 11 15
Email: quittek@netlab.nec.de
URI: http://www.netlab.nec.de

Martin Stiemerling
Network Laboratories, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 905 11 13
Email: stiemerling@netlab.nec.de
URI: http://www.netlab.nec.de

Thomas Dietz
Network Laboratories, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 905 11 28
Email: dietz@netlab.nec.de
URI: http://www.netlab.nec.de

Saverio Niccolini
Network Laboratories, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 905 11 18
Email: saverio.niccolini@netlab.nec.de
URI: http://www.netlab.nec.de

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.