Internet Engineering Task Force
INTERNET-DRAFT
ietf-sip-locating-00.ps

SIP WG
Schulzrinne/Rosenberg
Columbia U./dynamicsoft

October 2, 2000
Expires: January 2000

SIP: Session Initiation Protocol – Locating SIP Servers

Status of this Memo

Copyright Notice

Abstract

This document describes how a SIP client locates a SIP server based on the Request-URI or a pre-configured outbound proxy server. This document updates the process described in RFC 2543.

1 Introduction

This document updates Sections 1.3 and 1.4.2 and supercedes Appendix D of RFC 2543 [1]. Inter alia, it defines the term outbound proxy and replaces references to the obsoleted RFC 2052 with current references to RFC 2782.

1.1 Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [2] and indicate requirement levels for compliant SIP implementations.

1.2 Definitions

Outbound proxy: A *proxy* that is located near the originator of requests. It receives all outgoing requests **from** a particular UAC, including those requests whose Request-URIs identify a host other than the outbound proxy. The outbound proxy sends these requests, after any local processing, to the address indicated in the request-URI. (All other proxy servers are simply referred as proxies, not *inbound* proxies.)

2 Locating a SIP Server

When a client wishes to send a request, the client either sends it to a locally configured SIP proxy server, the so-called *outbound proxy*, independent of the Request-URI, or sends it to the IP address and port corresponding to the Request-URI. The outbound proxy can be configured by any mechanism, including DHCP [3].

If the Request-URI is used, the client needs to determine the protocol, port and IP address of a server to which to send the request. A client SHOULD follow the steps below to obtain this information.

At each step, unless stated otherwise, the client SHOULD try to contact a server at the port number listed in the Request-URI. If no port number is present in the Request-URI, the client uses port 5060. If the Request-URI specifies a protocol, the client contacts the server using that protocol. If no protocol is specified, the client tries UDP (if UDP is supported). If the attempt fails with an ICMP error of "destination unreachable", code "port unreachable" or "protocol unreachable" or a time out, or if the client doesn't support UDP but supports other protocols, it tries those protocols in some unspecified order.

A client SHOULD be able to interpret explicit network notifications (such as ICMP messages) which indicate that a server is not reachable, rather than relying solely on timeouts. (For example, in socket-based programs, `connect()` for TCP returns `ECONNREFUSED` if the client could not connect to a server at that address. For UDP, the socket needs to be bound to the destination address using `connect()` rather than `sendto()` or similar so that a second `write()` or `send()` fails with `ECONNREFUSED` if there is no server listening.) If the client finds the server is not reachable at a particular address, it SHOULD behave as if it had received a 400-class error response to that request.

The client tries to find one or more addresses for the SIP server by querying DNS. If a step elicits no addresses, the client continues to the next step. However if a step elicits one or more addresses, but no SIP server at any of those addresses responds, then the client concludes the server is down and does not continue on to the next step.

The service identifier for DNS SRV records [4] is "_sip".

1. If the maddr SIP URI parameter exists, it becomes the *destination address* used below; if not, the host element in the Request-URI is the destination address.

2. If the destination address is an IP address, the client contacts the server at the given address and the port number specified in the Request-URI or, if none is specified, the default port and ignores the remaining steps.

3. The Request-URI is examined. If it contains no port number or port 5060, the transport parameter is inspected:

   (a) There are three cases: the Request-URI does not specify a transport protocol, it specifies a client-supported transport protocol, or it specifies a protocol that is not supported by the client. We discuss these cases below in turn.

   If the Request-URI does not specify a transport protocol, DNS SRV records are retrieved according to RFC 2782 [4]. The results of the query or queries are merged and ordered based on priority, keeping only records with transport protocols that the client supports. Then, the searching technique outlined in RFC 2782 [4] is used to select servers in order. Server selection across requests is independent of previous choices, except as noted below for stateless proxies. The client attempts to contact each server in the order listed, at the port number specified in the SRV

record. If none of the servers can be contacted, the client gives up. If there are no SRV records (with any transport protocol), DNS address records are used, as described below.

If the Request-URI specifies a transport protocol and the transport protocol is supported by the client, the procedure in the paragraph above is used, limited to DNS resource records with the transport protocol specified in the Request-URI.

If the Request-URI specifies a transport protocol that is not supported by the client, the client gives up.

If the Request-URI contains a port number other than 5060 or if there are no SRV records, the client queries the DNS server for address records for the destination address. Address records include A RR's, AAAA RR's, or other similar records, chosen according to the client's network protocol capabilities. If the DNS server returns no address records, the client gives up.

Within a transaction, a stateless proxy MUST always select the same destination within the set of hosts with the same priority. This can be accomplished, for example, by using the modulo N of a hash of the Call-ID value or some other combination of transaction-identifying headers as the uniform random number described in the weighting algorithm of RFC 2782. Here, N is the sum of weights within the priority class.

A client MAY cache the list of DNS query results if one of the addresses was contacted successfully. Request for the same transaction SHOULD be sent to the same network address. Other requests from the same client select a server from the list of addresses cached, using the SRV load-balancing mechanism if applicable. The client must invalidate this list and retry the DNS query according to the rules in RFC1035 [5].

A client MAY omit attempting to reach a server which it had failed to reach for a previous request.

The results of the DNS lookup operation do not, in general, lead to a modification of the Request-URI.

A proxy is free to modify the Request-URI to any value desired, but the DNS lookups are usually based on the Request-URI obtained from a location server.

If the DNS time-to-live value exceeds a few minutes, servers generating a large number of requests are probably well advised to retry failed servers every few minutes.

3 Security Considerations

The security considerations in RFC 2543 [1] apply.

4 Authors' Addresses

Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
USA
electronic mail: schulzrinne@cs.columbia.edu

Jonathan Rosenberg
dynamicsoft

72 Eagle Rock Ave
East Hanover, NJ 07936
USA
electronic mail: jdrosen@dynamicsoft.com

References

[1] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, “SIP: session initiation protocol,” Request for Comments 2543, Internet Engineering Task Force, Mar. 1999.

[2] S. Bradner, “Key words for use in RFCs to indicate requirement levels,” Request for Comments 2119, Internet Engineering Task Force, Mar. 1997.

[3] G. Nair and H. Schulzrinne, “DHCP option for SIP servers,” Internet Draft, Internet Engineering Task Force, Apr. 2000. Work in progress.

[4] A. Gulbrandsen, P. Vixie, and L. Esibov, “A DNS RR for specifying the location of services (DNS SRV),” Request for Comments 2782, Internet Engineering Task Force, Feb. 2000.

[5] P. V. Mockapetris, “Domain names - implementation and specification,” Request for Comments 1035, Internet Engineering Task Force, Nov. 1987.

Full Copyright Statement