        Guidelines for Choosing RTP Control Protocol (RTCP) Canonical Names
                                    (CNAMEs)
                         draft-ietf-avt-rtp-cnames-03

Abstract

   The RTP Control Protocol (RTCP) Canonical Name (CNAME) is a
   persistent transport-level identifier for an RTP endpoint. While the
   Synchronization Source (SSRC) identifier of an RTP endpoint may
   change if a collision is detected, or when the RTP application is
   restarted, its RTCP CNAME is meant to stay unchanged, so that RTP
   endpoints can be uniquely identified and associated with their RTP
   media streams. For proper functionality, RTCP CNAMEs should be
   unique within the participants of an RTP session. However, the
   existing guidelines for choosing the RTCP CNAME provided in the RTP
   standard are insufficient to achieve this uniqueness. This memo
   updates these guidelines to allow endpoints to choose unique RTCP
   CNAMEs.

Status of this Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on June 25, 2011.

Table of Contents

1. Introduction

   In Section 6.5.1 of the RTP specification, [RFC3550], there are a
   number of recommendations for choosing a unique RTCP CNAME for an RTP
   endpoint. However, in practice, some of these methods are not
   guaranteed to produce a unique RTCP CNAME. This memo updates
   guidelines for choosing RTCP CNAMEs, superceding those presented in
   Section 6.5.1 of [RFC3550].

2. Requirements Notation

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in [RFC2119].

3. Deficiencies with Earlier Guidelines for Choosing an RTCP CNAME

   The recommendation in [RFC3550] is to generate an RTCP CNAME of the
   form "user@host" for multiuser systems, or "host" if the username is
   not available. The "host" part is specified to be the fully
   qualified domain name (FQDN) of the host from which the real-time
   data originates. While this guidance was appropriate at the time
   [RFC3550] was written, FQDNs are no longer necessarily unique, and
   can sometimes be common across several endpoints in large service
   provider networks. This document replaces the use of FQDN as an RTCP
   CNAME by alternative mechanisms.

   IPv4 addresses are also suggested for use in RTCP CNAMEs in
   [RFC3550], where the "host" part of the RTCP CNAME is the numeric
   representation of the IPv4 address of the interface from which the
   RTP data originates. As noted in [RFC3550], the use of private
   network address space [RFC1918] can result in hosts having network
   addresses that are not globally unique. Additionally, this shared
   use of the same IPv4 address can also occur with public IPv4
   addresses if multiple hosts are assigned the same public IPv4 address
   and connected to a Network Address Translation (NAT) device
   [RFC3022]. When multiple hosts share the same IPv4 address, whether
   private or public, using the IPv4 address as the RTCP CNAME leads to
   RTCP CNAMEs that are not necessarily unique.

   It is also noted in [RFC3550] that if hosts with private addresses
   and no direct IP connectivity to the public Internet have their RTP
   packets forwarded to the public Internet through an RTP-level
   translator, they may end up having non-unique RTCP CNAMEs. The
   suggestion in [RFC3550] is that such applications provide a
   configuration option to allow the user to choose a unique RTCP CNAME,

and puts the burden on the translator to translate RTCP CNAMEs from
private addresses to public addresses if necessary to keep private
addresses from being exposed. Experience has shown that this does
not work well in practice.

4. Choosing an RTCP CNAME

   It is difficult, and in some cases impossible, for a host to
   determine if there is a NAT between itself and its RTP peer.
   Furthermore, even some public IPv4 addresses can be shared by
   multiple hosts in the Internet. Using the numeric representation of
   the IPv4 address as the "host" part of the RTCP CNAME is NOT
   RECOMMENDED.

4.1. Persistent RTCP CNAMEs vs. Per-Session RTCP CNAMEs

   The RTCP CNAME can either be persistent across different RTP sessions
   for an RTP endpoint, or it can be unique per session, meaning that an
   RTP endpoint chooses a different RTCP CNAME for each RTP session.

   An RTP endpoint that is emitting multiple related RTP streams that
   require synchronization at the other endpoint(s) MUST use the same
   RTCP CNAME for all streams that are to be synchronized. This
   requires a short-term persistent RTCP CNAME that is common across
   several RTP flows, and potentially across several related RTP
   sessions. A common example of such use occurs when lip-syncing audio
   and video streams in a multimedia session, where a single participant
   has to use the same RTCP CNAME for its audio RTP session and for its
   video RTP session. Another example might be to synchronize the
   layers of a layered audio codec, where the same RTCP CNAME has to be
   used for each layer.

   A longer-term persistent RTCP CNAME is sometimes useful to facilitate
   third-party monitoring. One such use might be to allow network
   management tools to correlate the ongoing quality of service for a
   participant across multiple RTP sessions for fault diagnosis, and to
   understand long-term network performance statistics. Other, less
   benign, uses can also be envisaged. An implementation that wishes to
   discourage this type of third-party monitoring can generate a unique
   RTCP CNAME for each RTP session, or group of related RTP sessions,
   that it joins. Such a per-session RTCP CNAME cannot be used for
   traffic analysis, and so provides some limited form of privacy (note
   that there are non-RTP means that can be used by a third-party to
   correlate RTP sessions, so the use of per-session RTCP CNAMEs will
   not prevent a determined traffic analyst).

   This memo defines several different ways by which an implementation

can choose an RTCP CNAME. It is possible, and legitimate, for
independent implementations to make different choices of RTCP CNAME
when running on the same host. This can hinder third-party
monitoring, unless some external means is provided to configure a
persistent choice of RTCP CNAME for those implementations.

Note that there is no backwards compatibility issue (with [RFC3550]-
compatible implementations) introduced in this memo, since the RTCP
CNAMEs are opaque strings to remote peers.

4.2. Requirements

RTP endpoints will choose to generate RTCP CNAMEs that are persistent
or per-session. An RTP endpoint that wishes to generate a persistent
RTCP CNAME MUST use one of the following two methods:

o To produce a long-term persistent RTCP CNAME, an RTP endpoint MUST
   generate and store a Universally Unique IDentifier (UUID)
   [RFC4122] for use as the "host" part of its RTCP CNAME. The UUID
   MUST be version 1, 2 or 4 described in [RFC4122], with the
   "urn:uuid:" stripped, resulting in a 36-octet printable string
   representation.

o To produce a short-term persistent RTCP CNAME, an RTP endpoint
   MUST use either (a) the numeric representation of the layer-2
   (MAC) address of the interface that is used to initiate the RTP
   session as the "host" part of its RTCP CNAME or (b) generate an
   identifier by following the procedure described in Section 5. In
   either case, the procedure is performed once per initialization of
   the software. After obtaining a identifier by doing (a) or (b),
   the least significant 48 bits are converted to the standard colon-
   separated hexadecimal format, e.g., "00:23:32:af:9b:aa", resulting
   in a 17-octet printable string representation.

In the two cases above, the "user@" part of the RTCP CNAME MAY be
omitted on single-user systems, and MAY be replaced by an opaque
token on multi-user systems, to preserve privacy.

An RTP endpoint that wishes to generate a per-session RTCP CNAME MUST
use the following method:

o For every new RTP session, a new CNAME is generated following the
   procedure described in Section 5. After performing that
   procedure, the significant 96 bits are used to generate an
   identifier (to compromise between packet size and security) which
   is converted ASCII using Base64 encoding [RFC4648]. This results
   in a 16-octet string representation. The RTCP CNAME cannot change
   over the life of an RTP session [RFC3550], hence, only the initial

     SSRC value chosen by the endpoint is used. The "user@" part of
     the RTCP CNAME is omitted when generating per-session RTCP CNAMEs.

   It is believed that obtaining uniqueness is an important property
   that requires careful evaluation of the method. This document
   provides a number of methods, at least one of which would be suitable
   for all deployment scenarios. This document therefore does not
   provide for the implementor to define and select an alternative
   method.

   A future specification might define an alternative method for
   generating RTCP CNAMEs as long as the proposed method has appropriate
   uniqueness, and there is consistency between the methods used for
   multiple RTP sessions that are to be correlated. However, such a
   specification needs to be reviewed and approved before deployment.

5. Procedure to Generate a Unique Identifier

   The algorithm described below is intended to be used for locally-
   generated unique identifier.

   1. Obtain the current time of day in 64-bit NTP format [RFC5905].

   2. Obtain an EUI-64 identifier from the system running this
       algorithm. If an EUI-64 does not exist, one can be created from
       a 48-bit MAC address as specified in [RFC4291]. If an EUI-64
       cannot be obtained or created, a suitably unique identifier,
       local to the node, should be used (e.g., system serial number).

   3. Concatenate the time of day with the system-specific identifier
       in order to create a key.

   4. If generating a per-session CNAME, also concatenate RTP
       endpoint's initial SSRC, the source and destination IP addresses,
       and ports to the key.

   5. Compute an SHA-1 digest on the key as specified in [RFC4634],
       which outputs 160 bits.

6. Security Considerations

   The security considerations of [RFC3550] apply to this memo.

   In some environments, notably telephony, a fixed RTCP CNAME value
   allows separate RTP sessions to be correlated and eliminates the
   obfuscation provided by IPv6 privacy addresses [RFC4941] or IPv4 NAPT

[RFC3022]. Secure RTP (SRTP) [RFC3711] can help prevent such
correlation by encrypting Secure RTCP (SRTCP) but it should be noted
that SRTP only mandates SRTCP integrity protection (not encryption).
Thus, RTP applications used in such environments should consider
encrypting their SRTCP or generate a per-session RTCP CNAME as
discussed in Section 4.1.

7. IANA Considerations

   No IANA actions are required.

8. Acknowledgments

   Thanks to Marc Petit-Huguenin who suggested to use UUIDs in
   generating RTCP CNAMEs.

9. References

9.1. Normative References

   [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, July 2003.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy
              Extensions for Stateless Address Autoconfiguration in
              IPv6", RFC 4941, September 2007.

   [RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally
              Unique IDentifier (UUID) URN Namespace", RFC 4122,
              July 2005.

   [RFC4634] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms
              (SHA and HMAC-SHA)", RFC 4634, July 2006.

   [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data
              Encodings", RFC 4648, October 2006.

   [RFC5905] Mills, D., Martin, J., Burbank, J., and W. Kasch, "Network
              Time Protocol Version 4: Protocol and Algorithms
              Specification", RFC 5905, June 2010.

   [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing
              Architecture", RFC 4291, February 2006.

9.2. Informative References

   [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and
              E. Lear, "Address Allocation for Private Internets",
              BCP 5, RFC 1918, February 1996.

   [RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network
              Address Translator (Traditional NAT)", RFC 3022,
              January 2001.

   [RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K.
              Norrman, "The Secure Real-time Transport Protocol (SRTP)",
              RFC 3711, March 2004.

Authors' Addresses

   Ali Begen
   Cisco
   181 Bay Street
   Toronto, ON M5J 2T3
   CANADA

   Email: abegen@cisco.com

   Colin Perkins
   University of Glasgow
   School of Computing Science
   Glasgow,   G12 8QQ
   UK

   Email: csp@csperkins.org

   Dan Wing
   Cisco Systems, Inc.
   170 West Tasman Dr.
   San Jose, CA 95134
   USA

   Email: dwing@cisco.com