CCNx Publisher Serial Versioning
draft-mosko-icnrg-ccnxserialversion-00

Abstract

This document specifies using a serial number to indicate versions in name name segments. A serial number is an increasing unsigned integer with an increment of 1. Therefore, given one name with a serial version number, one may compute the next version number.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This document specifies using a serial number in a CCNx name as a version identifier. It specifies a new name segment label and a TLV encoding. The use of a serial number in a name to denote a version is limited to coordination among publishers if an attempt is made to use a serial number as a distributed ordering.

Packets are represented as 32-bit wide words using ASCII art. Because of the TLV encoding and optional fields or sizes, there is no concise way to represent all possibilities. We use the convention that ASCII art fields enclosed by vertical bars "|" represent exact bit widths. Fields with a forward slash "/" are variable bitwidths, which we typically pad out to word alignment for picture readability.

TODO -- we have not adopted the Requirements Language yet.

1.1. Requirements Language

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

2. Protocol Description

A serial number in a CCNx Name segment indicates an ordering on names based on the unsigned integer. The serial number is encoded in network byte order using the minimum number of bytes. The value of "0" is represented as the single byte %x00.

The serial number must be incremented by 1 for consecutive versions of the prior name name segments. There is no maximum serial number, it is limited only by the number of bytes put in to the name component.

A "GONE" PayloadType means that this version is a terminal version. All prior versions should be interpreted as deleted. A user, however, may publish more "DATA" after the terminal version, if he decides to un-delete it.

Type	Name
'Serial'	Serial number, displayed as an Integer.

Table 1: Labeled Content Information Types

Type	Symbol	Name	Description
%x0013	T_SERIAL	Serial Number	Serial Number, incrementing by +1.

Table 2: CCNx Name Types

3. Acknowledgements

4. IANA Considerations

TODO: Work with IANA to define the type space for CCNx Name segment types.

All drafts are required to have an IANA considerations section (see Guidelines for Writing an IANA Considerations Section in RFCs (Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs," May 2008.) [RFC5226] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

5. Security Considerations

All drafts are required to have a security considerations section. See RFC 3552 (Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations," July 2003.) [RFC3552] for a guide.

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997.

6.2. Informative References

[CCNx]   PARC, Inc., "CCNx Open Source," 2007.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations," BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs," BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008.

Author's Address

Marc Mosko
PARC
Palo Alto, California 94304
USA
Phone: +01 650-812-4405
Email: marc.mosko@parc.com