Uniform Resource Names for Device Identifiers
draft-arkko-core-dev-urn-03

Abstract

   This memo describes a new Uniform Resource Name (URN) namespace for
   hardware device identifiers. A general representation of device
   identity can be useful in many applications, such as in sensor data
   streams and storage, or equipment inventories. A URN-based
   representation can be easily passed along in any application that
   needs the information.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

   This memo describes a new Uniform Resource Name (URN) [RFC2141]
   [RFC3406] namespace for hardware device identifiers. A general
   representation of device identity can be useful in many applications,
   such as in sensor data streams and storage, or equipment inventories
   [I-D.ietf-core-coap], [I-D.jennings-senml],
   [I-D.arkko-core-sleepy-sensors] [I-D.arkko-core-security-arch]. A
   URN-based representation can be easily passed along in any
   application that needs the information, as it fits in protocols
   mechanisms that are designed to carry URNs [RFC2616], [RFC3261],
   [I-D.ietf-core-coap]. Finally, URNs can also be easily carried and
   stored in formats such as XML [W3C.REC-xml-19980210] or JSON
   [I-D.jennings-senml] [RFC4627]. Using URNs in these formats is often
   preferable as they are universally recognized, self-describing, and
   therefore avoid the need for agreeing to interpret an octet string as
   a specific form of a MAC address, for instance.

   This memo defines identity URN types for situations where no such
   convenient type already exist. For instance, [I-D.farrell-decade-ni]
   defines cryptographic identifiers, [I-D.montemurro-gsma-imei-urn]
   defines International Mobile station Equipment Identity (IMEI)
   identifiers for use with 3GPP cellular systems, and
   [I-D.atarius-dispatch-meid-urn] defines Mobile Equipment Identity
   (MEID) identifiers for use with 3GPP2 cellular systems. Those URN
   types should be employed when such identities are transported; this
   memo does not redefine these identifiers in any way.

   Universally Unique IDentifier (UUID) URNs [RFC4122] are another
   alternative way for representing device identifiers, and already
   support MAC addresses as one of type of an identifier. However,
   UUIDs can be inconvenient in environments where it is important that
   the identifiers are as simple as possible and where additional
   requirements on stable storage, real-time clocks, and identifier
   length can be prohibitive. UUID-based identifiers are recommended
   for all general purpose uses when MAC addresses are available as
   identifiers. The device URN defined in this memo is recommended for
   constrained environments.

   Future device identifier types can extend the device device URN type
   defined here, or define their own URNs.

   The rest of this memo is organized as follows. Section 3 defines the
   "DEV" URN type, and Section 4 defines subtypes for IEEE MAC-48,
   EUI-48 and EUI-64 addresses and 1-wire device identifiers. Section 5
   gives examples. Section 6 discusses the security considerations of
   the new URN type. Finally, Section 7 specifies the IANA registration
   for the new URN type and sets requirements for subtype allocations

within this type.

2. Requirements language

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

3. DEV URN Definition

Namespace ID: "dev" requested

Registration Information: This is the first registration of this
namespace, 2011-08-27.

Registration version number: 1

Registration date: 2011-08-27

Declared registrant of the namespace: IETF and the CORE working
group. Should the working group cease to exist, discussion should be
directed to the general IETF discussion forums or the IESG.

Declaration of syntactic structure: The identifier is expressed in
ASCII (UTF-8) characters and has a hierarchical structure as follows:

```
devurn = "urn:dev:" body componentpart
body = macbody / owbody / otherbody
macbody = "mac:" hexstring
owbody = "ow:" hexstring
otherbody = subtype ":" identifier
subtype = ALPHA *(DIGIT / ALPHA)
identifier = 1*unreserved
componentpart = [ ";" component [ componentpart ]]
component = *1(DIGIT / ALPHA)
hexstring = hexbyte /
            hexbyte hexstring
hexbyte = hexdigit hexdigit
hexdigit = DIGIT / hexletter
hexletter = "a" / "b" / "c" / "d" / "e" / "f"
```

The above Augmented Backus-Naur Form (ABNF) uses the DIGIT and ALPHA
rules defined in [RFC5234], which are not repeated here. The rule
for unreserved is defined in Section 2.3 of [RFC3986].

The device identity namespace includes three subtypes, and more may

be defined in the future as specified in Section 7.

The optional components following the hexstring are strings depicting individual aspects of a device. The specific strings and their semantics are up to the designers of the device, but could be used to refer to specific interfaces or functions within the device.

Relevant ancillary documentation: See Section 4.

Identifier uniqueness considerations: Device identifiers are generally expected to be unique, barring the accidental issue of multiple devices with the same identifiers.

Identifier persistence considerations: This URN type SHOULD only be used for persistent identifiers, such as hardware-based identifiers or cryptographic identifiers based on keys intended for long-term usage.

Process of identifier assignment: The process for identifier assignment is dependent on the used subtype, and documented in the specific subsection under Section 4.

Process for identifier resolution: The device identities are not expected to be globally resolvable. No identity resolution system is expected. Systems may perform local matching of identities to previously seen identities or configured information, however.

Rules for Lexical Equivalence: The lexical equivalence of the DEV URN is defined as an exact and case sensitive string match. Note that the two subtypes defined in this document use only lower case letters, however. Future types might use identifiers that require other encodings that require a more full-blown character set (such as BASE64), however.

Conformance with URN Syntax: The string representation of the device identity URN and of the MEID sub namespace is fully compatible with the URN syntax.

Validation Mechanism: Specific subtypes may be validated through mechanisms discussed in Section 4.

Scope: DEV URN is global in scope.

4. DEV URN Subtypes

4.1. MAC Addresses

DEV URNs of the "mac" subtype are based on the EUI-64 identifier
[IEEE.EUI64] derived from a device with a built-in 64-bit EUI-64.
The EUI-64 is formed from 24 or 36 bits of organization identifier
followed by 40 or 28 bits of device-specific extension identifier
assigned by that organization.

In the DEV URN "mac" subtype the hexstring is simply the full EUI-64
identifier represented as a hexadecimal string. It is always exactly
16 characters long.

MAC-48 and EUI-48 identifiers are also supported by the same DEV URN
subtype. To convert a MAC-48 address to an EUI-64 identifier, The
OUI of the Ethernet address (the first three octets) becomes the
organization identifier of the EUI-64 (the first three octets). The
fourth and fifth octets of the EUI are set to the fixed value FFFF
hexadecimal. The last three octets of the Ethernet address become
the last three octets of the EUI-64. The same process is used to
convert an EUI-48 identifier, but the fixed value FFFE is used
instead.

Identifier assignment for all of these identifiers rests within the
IEEE.

4.2. 1-Wire Device Identifiers

The 1-Wire* system is a device communications bus system designed by
Dallas Semiconductor Corporation. 1-Wire devices are identified by a
64-bit identifier that consists of 8 byte family code, 48 bit
identifier unique within a family, and 8 bit CRC code [OW].

   *) 1-Wire is a registered trademark.

In DEV URNs with the "ow" subtype the hexstring is a representation
of the full 64 bit identifier as a hexadecimal string. It is always
exactly 16 characters long. Note that the last two characters
represent the 8-bit CRC code. Implementations MAY check the validity
of this code.

Family code and identifier assignment for all 1-wire devices rests
with the manufacturers.

5. Examples

The following three examples provide examples of MAC-based, 1-Wire,
and Cryptographic identifiers:

```
    urn:dev:mac:0024befffe804ff1              # The MAC address of
                                              # Jari's laptop

    urn:dev:ow:10e2073a01080063               # The 1-Wire temperature
                                              # sensor in Jari's
                                              # kitchen

    urn:dev:ow:264437f5000000ed;humidity      # The laundry sensor's
                                              # humidity part

    urn:dev:ow:264437f5000000ed;temperature # The laundry sensor's
                                              # temperature part
```

6. Security Considerations

   On most devices, the user can display device identifiers. Depending
   on circumstances, device identifiers may or may not be modified or
   tampered by the user. An implementation of the DEV URN MUST NOT
   change these properties from what they were intended. In particular,
   a device identifier that is intended to be immutable should not
   become mutable as a part of implementing the DEV URN type. More
   generally, nothing in this memo should be construed to override what
   the relevant device specifications have already said about the
   identifiers.

   Other devices in the same network may or may not be able to identify
   the device. For instance, on Ethernet network, the MAC address of a
   device is visible to all other devices.

7. IANA Considerations

   Additional subtypes for DEV URNs can be defined through IETF Review
   or IESG Approval [RFC5226].

8. References

8.1. Normative References

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

   [RFC2141] Moats, R., "URN Syntax", RFC 2141, May 1997.

   [RFC3406] Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom,

"Uniform Resource Names (URN) Namespace Definition
                Mechanisms", BCP 66, RFC 3406, October 2002.

   [RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
                Resource Identifier (URI): Generic Syntax", STD 66,
                RFC 3986, January 2005.

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

   [RFC5234]   Crocker, D. and P. Overell, "Augmented BNF for Syntax
                Specifications: ABNF", STD 68, RFC 5234, January 2008.

   [IEEE.EUI64]
                IEEE, "Guidelines For 64-bit Global Identifier (EUI-64)",
                IEEE ,
                <http://standards.ieee.org/db/oui/tutorials/EUI64.html>.

   [OW]   IEEE, "Overview of 1-Wire(R) Technology and Its Use",
                MAXIM
                http://www.maxim-ic.com/app-notes/index.mvp/id/1796,
                <http://www.maxim-ic.com/app-notes/index.mvp/id/1796>.

8.2. Informative References

   [RFC2616]   Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,
                Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext
                Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

   [RFC3261]   Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston,
                A., Peterson, J., Sparks, R., Handley, M., and E.
                Schooler, "SIP: Session Initiation Protocol", RFC 3261,
                June 2002.

   [RFC3971]   Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure
                Neighbor Discovery (SEND)", RFC 3971, March 2005.

   [RFC3972]   Aura, T., "Cryptographically Generated Addresses (CGA)",
                RFC 3972, March 2005.

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

   [RFC4627]   Crockford, D., "The application/json Media Type for
                JavaScript Object Notation (JSON)", RFC 4627, July 2006.

[W3C.REC-xml-19980210]
          Sperberg-McQueen, C., Bray, T., and J. Paoli, "XML 1.0
          Recommendation", World Wide Web Consortium
          FirstEdition REC-xml-19980210, February 1998,
          <http://www.w3.org/TR/1998/REC-xml-19980210>.

[I-D.ietf-core-coap]
          Shelby, Z., Hartke, K., Bormann, C., and B. Frank,
          "Constrained Application Protocol (CoAP)",
          draft-ietf-core-coap-06 (work in progress), May 2011.

[I-D.jennings-senml]
          Jennings, C., "Media Type for Sensor Markup Language
          (SENML)", draft-jennings-senml-05 (work in progress),
          March 2011.

[I-D.farrell-decade-ni]
          Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B.,
          Keranen, A., and P. Hallam-Baker, "Naming Things with
          Hashes", draft-farrell-decade-ni-09 (work in progress),
          July 2012.

[I-D.arkko-core-sleepy-sensors]
          Arkko, J., Rissanen, H., Loreto, S., Turanyi, Z., and O.
          Novo, "Implementing Tiny COAP Sensors",
          draft-arkko-core-sleepy-sensors-01 (work in progress),
          July 2011.

[I-D.arkko-core-security-arch]
          Arkko, J. and A. Keranen, "CoAP Security Architecture",
          draft-arkko-core-security-arch-00 (work in progress),
          July 2011.

[I-D.montemurro-gsma-imei-urn]
          Montemurro, M., "A Uniform Resource Name Namespace For The
          GSM Association (GSMA) and the International Mobile
          station Equipment Identity(IMEI)",
          draft-montemurro-gsma-imei-urn-01 (work in progress),
          February 2007.

[I-D.atarius-dispatch-meid-urn]
          Atarius, R., "A Uniform Resource Name Namespace for the
          Device Identity and the Mobile Equipment Identity (MEID)",
          draft-atarius-dispatch-meid-urn-01 (work in progress),
          August 2011.

Appendix A. Changes from Previous Version

   Version -02 introduced several changes. The biggest change is that
   with the NI URNs [I-D.farrell-decade-ni], it was no longer necessary
   to define cryptographic identifiers in this specification. Another
   change was that we incorporated a more generic syntax for future
   extensions; non-hexstring identifiers can now also be supported, if
   some future device identifiers for some reason would, for instance,
   use BASE64. As a part of this change, we also changed the component
   part separator character from '-' to ';' so that the general format
   of the rest of the URN can employ the unreserved characters
   [RFC3986].

Appendix B. Acknowledgments

   The authors would like to thank Ari Keranen, Stephen Farrell,
   Christer Holmberg, Peter Saint-Andre, Wouter Cloetens, and Ahmad
   Muhanna for interesting discussions in this problem space. We would
   also like to note prior documents that focused on specific device
   identifiers, such as [I-D.montemurro-gsma-imei-urn] or
   [I-D.atarius-dispatch-meid-urn].

Authors' Addresses

   Jari Arkko
   Ericsson
   Jorvas 02420
   Finland

   Email: jari.arkko@piuha.net

   Cullen Jennings
   Cisco
   170 West Tasman Drive
   San Jose, CA 95134
   USA

   Phone: +1 408 421-9990
   Email: fluffy@cisco.com

Zach Shelby
Sensinode
Kidekuja 2
Vuokatti 88600
FINLAND

Phone: +358407796297
Email: zach@sensinode.com