Vehicle Identification Number-Based Unique Local IPv6 Unicast Addresses
                               (VULA)
                 draft-imadali-its-vinipv6-vula-00.txt

Abstract

   The Vehicle Identification Number (VIN) is standardized in ISO-3779
   and ISO-3780. The VIN is made of 17 alphanumeric characters code
   that uniquely identifies a vehicle worldwide. Some public
   information related to a vehicle can be obtained knowing its VIN
   code. This code may also be used to enable novel vehicular
   networking communications.

   RFC 4193 introduces a globally unique IPv6 unicast address format
   intended for local communications, usually inside of a site. These
   addresses (ULA) are not expected to be routable on the global
   Internet. This document introduces a method to build a VIN-based
   IPv6 Prefix that is intended for local communications involving more
   than one hop (VULA). The VIN-based generated prefix is assured to be
   unique among other VIN-based generated prefixes. Typically, in a
   scenario involving several vehicles, each single vehicle (in which
   one Mobile Router is in charge) is capable of generating its own
   unique infrastructure-independent globally-scoped VIN-based IPv6
   prefix. This document also describes some use cases where VULA could
   be exploited.

   This Internet-Draft will expire on August 19, 2013.

Copyright Notice

Table of Contents

1. Introduction and Problem

There exist currently several scenarios and requirements for IP in
Intelligent Transportation systems (its). A number of such scenarios
are described in [I-D.petrescu-its-scenarios-reqs]. Depending on the
typical topology where the communication happen, the draft groups the
communications into 4 categories: Intra-Vehicular communications (V),
Vehicle-to-Infrastructure (V2I), Vehicle-to-Vehicle (V2V) and
Vehicle-to-Vehicle-to-Infrastructure communications (V2V2I).

For example, in order to realize IP communications within a vehicle
(V), it is necessary to configure a distinct IP address on each IP-
enabled device deployed within a vehicle. This includes, and is not
limited to: passenger entertainment devices, navigation assistance
devices and, not least important, computers dedicated to electronics
control and safety messaging.

As another example, for Vehicle-to-Vehicle communications, it is
necessary to configure addresses on the devices within the two
vehicles in such a manner that it is possible to exchange IP packets
with a source address of a device in one vehicle and the destination
address a device in the other vehicle.

Finally, for Vehicle-to-Infrastructure communications (to Internet),
it is necessary that an address in one vehicle be globally unique and
topologically correct at the current point of attachment to the
Internet.

Whereas the last case is relatively solved with existing methods of
address auto-configuration (e.g. DHCPv6 or Stateless Address Auto-
configuration) the first two cases are more difficult. This
difficulty lies in the lack of authority of IP addressing in an
inherently distributed system formed by the vehicles. For example, a
DHCPv6 Server pre-configured with a pool of addresses to distribute
in one vehicle will allow for intra-vehicular communications (V).
But these addresses will be topologically incorrect with respect to
the current point of attachment to the Internet. This point of
attachment changes when a handover is performed re-connecting the
vehicle from GSM to WiFi access. In addition, these addresses will
be topologically incorrect with respect to the addresses of nearby
vehicles (the vehicle vicinity is highly dynamic).

The Mobile IP protocol could be employed to alleviate the problem of
topological correctness change upon inter-technology handovers; yet
this protocol has its inconvenients as well, such as the multi-
angular routing and the necessity of the presence of a Home Agent in
the infrastructure.

The prefix exchanges between moving networks could be implemented to
avoid the IP topological incorectness between vehicles.

For these reasons, several approaches exist to realize the address
auto-configuration without the help of a centralized management
scheme (DHCP nor ND no Mobile IP). Some approaches propose the use
of hardwired IP addresses in vehicles - at vehicle manufacturing
time, a number of addresses are hard-wired within a vehicle; in other
approaches the geographical coordinates may be used to form IP
addresses. Finally, certain approaches propose the use of the
vehicle's Vehicular Identification Number (VIN) in order to generate
IPv6 addresses related to the vehicle. It is along the lines of
these proposals that the current draft presents a method of forming a
ULA starting from the VIN ('VULA').

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 [RFC2119].

1.2. Abbreviations Used

VIN - Vehicle Identification Number

WMI - World Manufacturer Identifier

VDS - Vehicle Description Section

VIS - Vehicle Identification Section

2. VIN Overview

The VIN is a 17 characters alphanumeric code that uniquely identifies
a vehicle worldwide. Involved standard documents include :

o ISO 3833:1977, "Road vehicles -- Types -- Terms and definitions".
   This document defines terms relating to some types of road
   vehicles designated according to certain design and technical
   characteristics. It Applies to all vehicles designated for road
   circulation (with exception of agricultural tractors).

o ISO 4030:1983, "Road vehicles -- Vehicle identification number
   (VIN) -- Location and attachment". This document Specifies the
   requirements for the location and marking of the VIN on motor
   vehicles, trailers, motorcycles and mopeds as defined in ISO 3833.

o ISO 3779:2009, "Road vehicles -- Vehicle identification number
   (VIN) -- Content and structure". It specifies the content and
   structure of a VIN in order to establish, on a world-wide basis, a
   uniform identification numbering system for road vehicles. This
   document applies to motor vehicles, towed vehicles, motorcycles
   and mopeds as defined in ISO 3833.

o ISO 3780:2009, "Road vehicles -- World manufacturer identifier
   (WMI) code". It specifies the content and structure of an
   identifier in order to establish, on a world-wide basis, the
   identification of road vehicle manufacturers. The WMI constitutes
   the first section of the VIN described in ISO 3779. ISO 3780:2009
   applies to motor vehicles, towed vehicles, motorcycles and mopeds
   as defined in ISO 3833.

The VIN is used to uniquely identify a vehicle and therefore must
appear on each vehicle. Some public information related to a vehicle
can be obtained knowing its VIN code. This possibility is used in
thefts prevention by assisting law enforcement authorities in tracing
and recovering parts from stolen motor vehicles, or reporting vehicle
history to sell/buy a used car.

The VIN is written in 17 alphanumeric characters with a combination
of the 10 Arabic numbers and 23 capital letters of the Latin
alphabet. Excluded letters are I(i), O(o) and Q(q). This is to
avoid confusion between these characters and the numbers 0 and 1.

```
                    | 1 2 3 4 5 6 7 8 9 0 |
              |A B C D E F G H J K L M N P R S T U V W X Y Z|
```

        Figure 1: Alphabet for the generation of VIN codes

Although the definition of the code differs from USA, where the
National Highway Traffic Safety Administration (NHTSA) definition is
used, and Europe where the ISO is used, both VIN codes are compliant
and do not cause confusion. There are some other implementations of
the same code, for example within the Society of Automotive Engineers
(SAE) and Australian Design Rules (ADR) which are also compliant with
the European ISO format of the VIN. The VIN code contains three
sections.

```
           1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
          |   WMI   |       VDS       |         VIS           |
          +--------+-----------------+-----------------------+
```

                   Figure 2: VIN code sections

2.1. WMI

   The WMI is 3 digits long and uniquely designates the manufacturer's
   continent, country, and the unique national identifier. The 3 digit
   codes are defined within ISO-3780 and the database is maintained by
   the SAE. A WMI code can be revoked but cannot be used to designate
   another manufacturer for 30 years after its revocation date.

   ISO-3780 standard specifies the use of each bit in the WMI code.
   From the left to the right, the first bit value designates the region
   of the car manufacturer. Values [A-C] are reserved for Africa, [1-5]
   for North America, [8-9] for South America, [J-R] for Asia, [S-Z] for
   Europe and [6] for Oceania. Multiple alphanumeric values can be
   assigned for the same region, depending on the needs. Some values
   are reserved for future use.

   Second bit indicates a country in the region designated by the first
   bit. Multiple values are possible for the same country if needed.
   For example, if the first bit is V (in Europe) and the second bit is
   one the letters (F to R), then the designated country is France.
   Unique identity of a country is assured by the combination of these
   two values. The assigned codes by country are maintained by SAE and
   listed in ISO-3780.

   Third character of this section designates a national unique value
   for the car manufacturer, maintained by national authorities. It is
   possible to assign more than one value to the manufacturer if needed.
   Thus, a unique identification of the car manufacturer can be obtained
   by the combination of the three values.

   ISO-3780 distinguishes manufacturers building more or less than 500
   cars per year. If less, 3rd digit of WMI is set to '9' and positions
   12 to 14 (included) form an extended unique manufacturer identifier
   that identifies all the manufacturers that build less than 500 cars a
   year in the same country. If the manufacturer builds more than 500
   cars a year, it has one or more identification number(s) depending on
   the needs. For example, "1FA" and "3FA" identifies Ford in USA and
   Mexico, respectively. "VF3" is one of the WMI codes of French
   manufacturer Peugeot.

2.2. VDS

   The VDS is the second part of the VIN code. It is 6 characters long
   and gives more information about the vehicle. The description of the
   vehicle is not unique and each manufacturer has its own mapping table
   for this section; that is, a same character may have different
   meanings depending on the manufacturer, and sometimes differ upon the
   vehicle model.

The information given by this section may relate to the vehicle
weight, the model, the engine type, the body style or the engine
power, for example. It is also possible for the manufacturer to fill
this section with "dummy" information if it does not want to use it,
as spaces (blanks) are not allowed. Therefore, the manufacturer may
not rely on this section to complete the unique identification of a
vehicle.

Last position of this section (9th digit) is called the check digit.
Like the TCP checksum, the check digit is the result of a standard
algorithm where the values of other positions are used to generate
this value. Possible values for the check digit are numbers 0 to 9
and the letter X [CHECK].

2.3. VIS

The VIS is the third section of the VIN code. It is 8 characters
long and, combined with the VDS section, uniquely identifies a
vehicle within a car manufacturer for 30 years. The combination of
the VIS and the WMI section allows to uniquely identify a vehicle
worldwide. This section goes from the 10th digit to the 17th.

Digit number 10 designates the year model. It is the year during
which the vehicle has been manufactured, or the vehicle model year
depending on the manufacturer choice. For this digit, the allowed
values are 1 to 9 (0 forbidden) and uppercase Latin alphabet letters
except I, O, Q, U and Z. Years from 1980 to 2000 are coded with
letters from A to Y; years from 2001 to 2009 are represented with
digits from 1 to 9; year 2010 and later are represented with letters
from A to Y. This representation gives a cycle of 30 years during
which a VIS code is guaranteed to be unique.

11th digit of the VIN designates the plant of the manufacturer where
the vehicle has been assembled. For example, 'E' refers to 'Kentucky
Truck' plant (Jefferson county, Kentucky) and 'T' refers to 'Otosan
Kocaeli Assembly' plant (Kocaeli, Turkey), both plants belong to the
manufacturer 'Ford'. The same value may represent different plants
depending on the manufacturer.

Positions 12 to 17 represent the sequential identification part which
is unique on the production line and assigned by the car
manufacturer. Digits from 12 to 14 (if digit 3 of WMI is '9')
represent the rest of the WMI code of the car manufacturer that
produces less than 500 cars a year. Otherwise (general case),
positions 12 to 17 are considered as a whole and generally considered
as a sequential number. ISO-3779:2009 specifies that the last 4
positions must be numeric, which is applied in Europe. In North
America (Canada and USA), the last 5 digits must be numeric for some

kind of vehicles and only the last 4 digits for the rest.

```
        1 2 3 4 5 6 7  8   9    10    11   12  13  14   15  16  17
   +------+-+-+-+-+-+-+-+--+-----+-----+-----+---+---+----+---+---+--+
   | ISO  |      |          |     |                                  |
   | 3779 | WMI  |   VDS    |     |               VIS                |
   +------+-----+----------+-----+-----+-----+----------------------+
   | >500 |     | Vehicle  |Check|Model|Plant|                      |
   |cars a| WMI |Attributes|digit|Year |Code |    Sequential Number |
   | year |     |          |     |     |     |                      |
   +------+-----+----------+-----+-----+-----+-----------+----------+
   | <500 |     | Vehicle  |Check|Model|Plant|Manufacturer|Sequential|
   |cars a| WMI |Attributes|digit|Year |Code | IDentifier |  Number  |
   | year |     |          |     |     |     |            |          |
   +------+-----+----------+-----+-----+-----+-----------+----------+
```

           Figure 3: Summary of VIN code, sections definition and use

3. Mapping method

The VIN code can be split into 3 parts: WMI, VDS, and VIS. Each part
gives a partial description of one vehicle's attributes: manufacturer
ID, vehicle's description and vehicle's serial ID.

This section specifies the algorithm to perform in order to set an
IPv6 [RFC4291] Unique Local Unicast Address out of a VIN code (VULA).
ULA addresses [RFC4193] are specified for local communications. The
method used is based on two short and powerful assumptions that allow
for the compression of a maximum number of VIN digits in a minimum
number of bits.

The generation method conserves the uniqueness property of VIN codes
hence the uniqueness of generated prefixes. This collision-free
derivation method is also infrastructure-less, and allows for up to
32 subnets (5 bits) with the same VULA prefix.

3.1. Initial assumption

The unique identity of a vehicle is set through two sections of the
VIN: the WMI (digits 1 to 3, included) and VIS (digits 12 to 17,
included). According to ISO 3779:2009, this information may not be
enough, but it is certainly mandatory. Specifically, the cases where
dummy values are filled in the VDS section (allowed by ISO 3779:2009)
makes these two sections the only relevant information known about
the vehicle in order to have its unique identity .

The VDS section can be recovered in the cases when WMI and VIS
sections are known, using a multi-key query on a local database owned
by the manufacturer. A reasonable assumption statement (Assumption
1) based on the previous arguments is summarized as follows:

Assumption 1: WMI and VIS sections of a VIN code uniquely identify a
vehicle

3.2. Detailed algorithm

In order to convert the alphanumeric VIN into a binary ULA, an
initial and straightforward two-steps approach consists in following
the operation of mapping (transliteration) by the operation of
conversion. The VIN alphabet previously described, leaves 33
possible values for every single VIN digit. The mapping operation
consists in representing a VIN digit in decimal and the conversion
phase consists in converting this decimal to a binary number. Each
resulting binary is 6 bits long in order to cover all the
possibilities.

This initial method is highly inefficient regarding the number of
digits converted versus the number of bits used (which is limited to
64 for ULA). For instance, converting WMI and VIS sections (a total
of 11 digits) would require 66 bits (6*11) which would not fit in the
ULA address format.

The current proposal specifies the use of a VIN-specific numeral
system (cf. Appendix) using only allowed values; that is Arabic
numerals (0 to 9) and Latin letters (A to Z) excluding the exceptions
(I, O, and Q) in order to generate Base-VIN numbers. As a
comparison, Base-36 (radix 36) contains numerals (0-9) and letters
(A-Z). Numbers of this system can be converted to other bases (Hex,
decimal, octal and binary). Digits are ordered as follows: 0 < 1 ...
< 9 < A ... < Y < Z. Base-VIN can be considered as Base-36 system
tailored for VIN-related uses. From this proposal derives the below
work hypothesis (Assumption 2) that allows us to convert more than a
single VIN digit at a time.

Assumption 2: VIN codes are numbers written in the Base-VIN system.

```
            1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         |    A    |      B      | C| D|    E    |    F     |
         +--------+-------------+--+--+--------+----------+
```
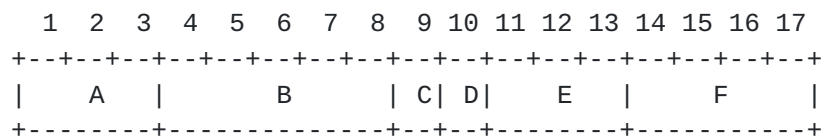
Figure 4: Redesign of the VIN code according to sections definitions

The compression gain is achieved by converting a value read in Base-
VIN to the binary base, rather than mapping/converting separate
digits. The restricted set of allowed values for VIN sections helps
reducing the number of bits besides preserving the VIN uniqueness
property. Sections A, D, E, and F are extracted from the VIN code
and converted accordingly. Sections A and E are read in Base-VIN
while sections D and F in decimal.

The resulting number of bits after conversion is 51 (over 66 with the
previous method). Using ULA bitmap, 51 bits (40 + 11 bits) would
require using the 40 global ID section and 11 extra bits from the
subnet ID. The new prefix ID is 5 bits long which leaves up to 32
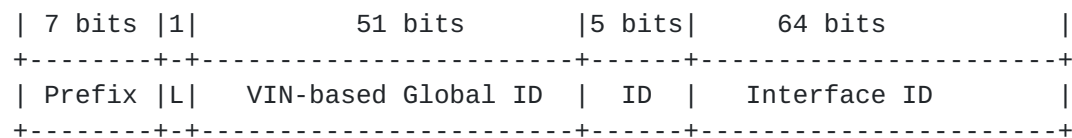distinct subnets with the initial VIN-extracted Global ID.

```
| 7 bits |1|          51 bits       |5 bits|      64 bits            |
+--------+-+-----------------------+------+-----------------------+
| Prefix |L|   VIN-based Global ID  | ID  |    Interface ID        |
+--------+-+-----------------------+------+-----------------------+
```

                    Figure 5: VIN-based ULA bitmap

In details, section A (WMI) after conversion is 16 bits long (log2
(33^3)); section D (year model) is 5 bits long (log2(30)); section E
(first alphanumeric part of the VIS) is 16 bits long (log2 (33^3));
and section F (second numeric part of the VIS) is 14 bits long
(log2(10^4)). This algorithm is reversible: from the converted
values (binary), it is possible to compute the initial coded sections
(VMI and VIS). This property is the result of the bijectiveness of
used operations for conversion between different numeral systems.
Useful scenarios are discussed below and privacy consequences are
addressed in the security considerations section.

3.3. Examples of use

The Mobile Router (MR) that executes the previous algorithm, is
capable of announcing the generated prefix on one or several internal
interfaces and configure one or several external interfaces,
depending on the scenario. For instance, under some conditions,
knowing the VIN code of a vehicle, one can deduce the internally
advertised prefix and remotely access a well-known internal device
(with a certain MAC address). This access might be possible for
vehicle manufacturers in order to perform remote diagnostic, or other
car rental companies depending on the application.

Using the vehicle's VIN and the above method, the MR can deduce the
same VULA prefix and advertise it internally or use it to configure
its own external interfaces. The use of MIPv6/NEMO protocol
[RFC3963] on the MR in order to guarantee the use of the same home

prefix is no longer necessary and the remote access to internal
devices might be addressed using other techniques.

Possible use cases involve classical vehicular IPv6 networking
scenarios: in-vehicle communications (Machine-type), and inter-
vehicles communications (distributed games, social communications
enabler). In order to enable vehicle-to-Infrastructure
communications using VULA (remote monitoring), which is not in the
initial scope of ULA addresses, a special IANA request should be made
about reserving a special code for VULA prefixes, or the
(re)definition of the L bit in the current ULA bitmap format
[RFC4193].

4. VIN-based Unique Local IPv6 Unicast Addresses

ULA addresses are used interchangeably with globally-scoped IPv6
addresses and communication between a limited set of sites is
possible. In particular, RFC 4193 defines the functionality of a
border router that prevents packets with ULA destination to cross
towards the global Internet. In a vehicular setting, a border router
could be the MR if provided with access to the Infrastructure.

Known limitations of RFC 4193 are due to the pseudo-random generation
method of these addresses. Depending on the algorithm, a collision
is highly unlikely but theoretically possible with a low probability.
In the VIN-based scenario, a ULA prefix is generated at the MR level
(independently from the ISP/RIR) and advertised inside the vehicular
network, to allow in-vehicle network to network communications if the
necessary routes are installed in respective MRs (border routers).

4.1. Method 1: RFC 4193 compliant Unique Local IPv6 Unicast Address
     generation

The VIN is, according to section 3.2.2 of RFC 4193, a suitably unique
identifier that could be used locally to the MR for the generation of
an IPv6 ULA prefix and can thus be used in the algorithm described in
the same section. Basically, step 2 of the aforementioned algorithm
is transformed in order to take the local VIN code as input. The
resulting ULA prefix is advertised on MR's ingress interface, or used
to configure any other local interface.

Since RFC 4193 algorithm relies on a pseudo-random generation method
for the ULA prefix, and introduces, for example, the timestamp at the
moment of the execution, two different instances of the same
algorithm given the same VIN code, will result in two different
prefixes. When this method has obvious benefits for privacy concerns
or collision avoidance, it fails in the requirements described in

Section 3.3. In particular, in-vehicle advertised prefix changes
with the timestamp, which makes our remote access scenario based on
the knowledge of a vehicle's VIN almost impossible to realize without
MIPv6/NEMO running on the MR. Also, the VIN code is hashed and
partly present in the Global ID, which makes it impossible to guess
from a given ULA prefix.

A similar RFC4193-compliant approach for the prefix generation has
been presented at IETF-83 (Paris, France) [BLESS]. This method
employs the VIS (or VIN7) part only of the VIN in a pseudo-random
generation algorithm. It can also be used to derive an IPv6 address
more deterministically, and yet preserve privacy.

4.2. Method 2: VIN-based Unique Local IPv6 Unicast Address generation

The conversion method described in Section 3.2 defines a new VULA
prefix format (as depicted in Figure 5) which is guaranteed unique
amongst the VIN-based generated prefixes. Knowing a VIN code, it is
possible to derive the related ULA prefix and use this information
for a remote access.

This subtype of ULA prefixes which has enhanced uniqueness guarantees
may be defined in a separate category that requests specific /8
prefix (for example) that are expected to be globally routed.
Another solution might be the (re)definition of the value 0 of the
bit L illustrated in the ULA prefix bitmap.

There are obvious privacy and security concerns if the method 2 is
used. Some of these considerations are discussed in Section 7.

5. Acknowledgements

This work has been performed in the framework of the ICT project ICT-
5-258512 EXALTED, which is partly funded by the European Union. The
organisations on the source list [CEA] would like to acknowledge the
contributions of their colleagues to the project, although the views
expressed in this contribution are those of the authors and do not
necessarily represent the project.

Roland Bless provided useful comments helping to clarify the
presentation.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

   Collision happens if at least two interfaces try to set the same IPv6
   address in the same subnet. The 5 remaining bits using our method
   can then be used to fix the collision by pulling another adjacent
   prefix (32 possibilities). The proposed mapping/conversion method
   strictly applied to ISO-3779 definitions makes a collision highly
   unlikely but theoretically possible with the use of revoked or forged
   VIN codes. We recommend the collision avoidance using the 5
   remaining bits to complete assumption 1 (uniqueness conservation) in
   order to use a vehicle's VIN to generate VULA even in these rare
   exceptions. If VIN codes cannot be forged (using tamper-proof
   devices for example) the 5 bits could be used for subnets inside a
   VIN identified vehicle.

   Privacy is an important requirement for in-vehicle network-to-network
   communications. State of the art vehicular applications proposals
   usually disclose accurate information about geographical position,
   velocity and identity in safety beacons. Even if these
   advertisements are signed according to IEEE 1609.2, the above VULA
   proposal does not fit in this category and does not provide
   mechanisms to protect the in-vehicle network privacy. The focus of
   this work is to enable in-vehicle networks to exchange packets with
   VIN-based IPv6 addresses. Uniqueness and collision-avoidance are
   thus top list requirements for this objective. In order not to
   exacerbate privacy concerns, VIN-based IPv6 communications should
   take precautions in advance. For instance, a Caesar ciphering method
   would map into Base-VIN with substituted values, before VULA
   generation. A compromise between collisions and privacy expectations
   should be made in this case.

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.

   [RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A., and P.
              Thubert, "Network Mobility (NEMO) Basic Support Protocol",
              RFC 3963, January 2005.

   [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast
              Addresses", RFC 4193, October 2005.

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

8.2. Informative References

[BLESS]   Bless, R., "A Secure IPv6-based Architecture for Internal
            Automotive Networks", March 2012,
            <http://www.lara.prd.fr/ietf-its>.

[CHECK]   US Code of Federal Regulations, "PART 565--VEHICLE
            IDENTIFICATION NUMBER REQUIREMENTS", June 1996, <http://
            www.nhtsa.gov/DOT/NHTSA/Rulemaking/Rules/
            Associated%20Files/VIN_Final_Rule_April_08.pdf>.

[I-D.petrescu-its-scenarios-reqs]
            Petrescu, A., Janneteau, C., Boc, M., and W. Klaudel,
            "Scenarios and Requirements for IP in Intelligent
            Transportation Systems",
            draft-petrescu-its-scenarios-reqs-01 (work in progress),
            July 2012.

Appendix A. Base-VIN Numeral System Specification

A.1. Definitions

The Base-VIN is the numeral system where all VIN codes belong. A
number in Base-VIN is convertible in other numeral systems (decimal,
binary, hexadecimal, etc) by simple multiplication operations and
vice-versa, with simple division operations. The Base-VIN numeral
system contains 33 different ordered digits. Their value in the
decimal system goes from 0 to 32, and the symbols used are those
defined in the VIN description: One of the letters in the set
[ABCDEFGHJKLMNPRSTUVWXYZ] or a numeral in the set [0123456789]. The
ordered set of Base-VIN numerals is defined below.

```
[ Decimal|0|1|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16|17]
[Base-VIN|0|1|2|3|4|5|6|7|8|9| A| B| C| D| E| F| G| H]

[ Decimal|18|19|20|21|22|23|24|25|26|27|28|29|30|31|32]
[Base-VIN| J| K| L| M| N| P| R| S| T| U| V| W| X| Y| Z]
```

                Figure 6: Base-VIN ordered numbering

A.2. Conversion from decimal to Base-VIN

A decimal number can be written in the Base-VIN following simple
rules using simple division operations. The algorithm is similar to
conversion from decimal to other numeral systems (binary for example)
and it is defined as an extension of it. Here are the necessary

steps: Suppose we have to convert X from decimal system to VIN
system. The resulting number is Y

1) Divide X by 33 while the quotient is greater than 33

2) Write the remainder in the Base-VIN numeral system

3) Stop division when the quotient is less than 33 and write it in
     Base-VIN

4) Read Y backwards from last quotient to first remainder, putting
     the last quotient in the most significant bit (MSB) and the first
     remainder in the least significant bit (LSB). Y is now in the
     Base-VIN numeral system

Example: Let us convert 1200 from decimal system to VIN system.

o   1200/33 = 36 + 12[33] ---> 12 is the first remainder. It is C in
     Base-VIN

o   36 is greater than 33, so we shall continue

o   36/33 = 1 + 3[33] ---> 3 is the second remainder. It is 3 in
     Base-VIN

o   1 is less than 33, so this is the end of our algorithm. 1 is 1 in
     Base-VIN

o   The result of the conversion is '13C'

A.3. Conversion from Base-VIN to decimal

A Base-VIN number can be written in the decimal system following
simple rules using simple multiplication and power operations. The
algorithm is similar to conversion from other numeral systems to
decimal (binary for example) and it defined as an extension of it.
Here are the necessary steps: Suppose we have to convert 'X' from
Base-VIN numeral system to decimal. The resulting number is 'Y'.
Let 'n' be the number of positions in X and X(i) the Base-VIN digit
of X in ieth position starting from 1, reading the number from the
right (LSB) and converted to decimal (from the table above). Then:

Y = 0;

for (i = n-1; i == 0; i--)

```
   Y+=X(i+1)*(33^i);
```

   Example: Let us reconvert the result from the previous example, from
   Base-VIN to decimal. X = 13C, n = 3, LSB = C and MSB = 1.

```
   Y = 1*(33^2) + 3*(33^1) + C*(33^0)

     = 1*(1089) + 3* (33) + 12*(1)

     = 1200
```

A.4. Number of positions

   It is possible to know how many positions are necessary to write a
   number in the Base-VIN numeral system before doing the conversion
   from decimal. It is necessary for that to have the immediate power
   of 33 that is greater than or equal to the decimal number to convert.
   The number of necessary positions is this power of 33. For example,
   suppose that you have to convert 1200 from decimal numeral system to
   Base-VIN. 1200 is immediately less than 33^3 and greater than 33^2.
   So 1200 will be written in 3 Base-VIN positions after conversion (We
   saw above that 1200 is actually '13C' in Base-VIN).

Authors' Addresses

   Sofiane Imadali
   CEA
   CEA, LIST, Communicating Systems Laboratory, Point Courrier 173
   Gif-sur-Yvette, Essonne F-91191
   France

   Phone: +33 0169080727
   Email: sofiane.imadali@cea.fr

   Alexandru Petrescu
   CEA
   CEA, LIST, Communicating Systems Laboratory, Point Courrier 173
   Gif-sur-Yvette, Essonne F-91191
   France

   Phone: +33 0169089223
   Email: alexandru.petrescu@cea.fr

   Christophe Janneteau
   CEA
   CEA, LIST, Communicating Systems Laboratory, Point Courrier 173
   Gif-sur-Yvette, Essonne F-91191
   France

   Phone: +33 0169089182
   Email: christophe.janneteau@cea.fr