Stream: Independent Submission
RFC: 8771
Category: Experimental
Published: 1 April 2020
ISSN: 2070-1721
Authors: A. Mayrhofer, *nic.at GmbH*
J. Hague, *Sinodun*

RFC 8771
The Internationalized Deliberately Unreadable Network NOtation (I-DUNNO)

Abstract

Domain Names were designed for humans, IP addresses were not. But more than 30 years after the introduction of the DNS, a minority of mankind persists in invading the realm of machine-to-machine communication by reading, writing, misspelling, memorizing, permuting, and confusing IP addresses. This memo describes the Internationalized Deliberately Unreadable Network NOtation ("I-DUNNO"), a notation designed to replace current textual representations of IP addresses with something that is not only more concise but will also discourage this small, but obviously important, subset of human activity.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8771.

Copyright Notice

Table of Contents

1. Introduction

In Section 2.3 of [RFC0791], the original designers of the Internet Protocol carefully defined names and addresses as separate quantities. While they did not explicitly reserve names for human consumption and addresses for machine use, they did consider the matter indirectly in their philosophical communal statement: "A name indicates what we seek." This clearly indicates that names rather than addresses should be of concern to humans.

The specification of domain names in [RFC1034], and indeed the continuing enormous effort put into the Domain Name System, reinforces the view that humans should use names and leave worrying about addresses to the machines. RFC 1034 mentions "users" several times, and even includes the word "humans", even though it is positioned slightly unfortunately, though perfectly

understandably, in a context of "annoying" and "can wreak havoc" (see Section 5.2.3 of [RFC1034]). Nevertheless, this is another clear indication that domain names are made for human use, while IP addresses are for machine use.

Given this, and a long error-strewn history of human attempts to utilize addresses directly, it is obviously desirable that humans should not meddle with IP addresses. For that reason, it appears quite logical that a human-readable (textual) representation of IP addresses was just very vaguely specified in Section 2.1 of [RFC1123]. Subsequently, a directed effort to further discourage human use by making IP addresses more confusing was introduced in [RFC1883] (which was obsoleted by [RFC8200]), and additional options for human puzzlement were offered in Section 2.2 of [RFC4291]. These noble early attempts to hamper efforts by humans to read, understand, or even spell IP addressing schemes were unfortunately severely compromised in [RFC5952].

In order to prevent further damage from human meddling with IP addresses, there is a clear urgent need for an address notation that replaces these "Legacy Notations", and efficiently discourages humans from reading, modifying, or otherwise manipulating IP addresses. Research in this area long ago recognized the potential in ab^H^Hperusing the intricacies, inaccuracies, and chaotic disorder of what humans are pleased to call a "Cultural Technique" (also known as "Script"), and with a certain inexorable inevitability has focused of late on the admirable confusion (and thus discouragement) potential of [UNICODE] as an address notation. In Section 4, we introduce a framework of Confusion Levels as an aid to the evaluation of the effectiveness of any Unicode-based scheme in producing notation in a form designed to be resistant to ready comprehension or, heaven forfend, mutation of the address, and so effecting the desired confusion and discouragement.

The authors welcome [RFC8369] as a major step in the right direction. However, we have some reservations about the scheme proposed therein:

- Our analysis of the proposed scheme indicates that, while impressively concise, it fails to attain more than at best a Minimum Confusion Level in our classification.
- Humans, especially younger ones, are becoming skilled at handling emoji. Over time, this will negatively impact the discouragement factor.
- The proposed scheme is specific to IPv6; if a solution to this problem is to be in any way timely, it must, as a matter of the highest priority, address IPv4. After all, even taking the regrettable effects of RFC 5952 into account, IPv6 does at least remain inherently significantly more confusing and discouraging than IPv4.

This document therefore specifies an alternative Unicode-based notation, the Internationalized Deliberately Unreadable Network NOtation (I-DUNNO). This notation addresses each of the concerns outlined above:

- I-DUNNO can generate Minimum, Satisfactory, or Delightful levels of confusion.
- As well as emoji, it takes advantage of other areas of Unicode confusion.
- It can be used with IPv4 and IPv6 addresses.

We concede that I-DUNNO notation is markedly less concise than that of RFC 8369. However, by permitting multiple code points in the representation of a single address, I-DUNNO opens up the full spectrum of Unicode-adjacent code point interaction. This is a significant factor in allowing I-DUNNO to achieve higher levels of confusion. I-DUNNO also requires no change to the current size of Unicode code points, and so its chances of adoption and implementation are (slightly) higher.

Note that the use of I-DUNNO in the reverse DNS system is currently out of scope. The occasional human-induced absence of the magical one-character sequence U+002E is believed to cause sufficient disorder there.

Media Access Control (MAC) addresses are totally out of the question.

2. Terminology

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Additional terminology from [RFC6919] MIGHT apply.

3. The Notation

I-DUNNO leverages UTF-8 [RFC3629] to obfuscate IP addresses for humans. UTF-8 uses sequences between 1 and 4 octets to represent code points as follows:

Char. number range (hexadecimal)	UTF-8 octet sequence (binary)
0000 0000 - 0000 007F	0xxxxxxx
0000 0080 - 0000 07FF	110xxxxx 10xxxxxx
0000 0800 - 0000 FFFF	1110xxxx 10xxxxxx 10xxxxxx
0001 0000 - 0010 FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

Table 1

I-DUNNO uses that structure to convey addressing information as follows:

3.1. Forming I-DUNNO

In order to form an I-DUNNO based on the Legacy Notation of an IP address, the following steps are performed:

1. The octets of the IP address are written as a bitstring in network byte order.
2. Working from left to right, the bitstring (32 bits for IPv4; 128 bits for IPv6) is used to generate a list of valid UTF-8 octet sequences. To allocate a single UTF-8 sequence:
   a. Choose whether to generate a UTF-8 sequence of 1, 2, 3, or 4 octets. The choice OUGHT TO be guided by the requirement to generate a satisfactory Minimum Confusion Level (Section 4.1) (not to be confused with the minimum Satisfactory Confusion Level (Section 4.2)). Refer to the character number range in Table 1 in order to identify which octet sequence lengths are valid for a given bitstring. For example, a 2-octet UTF-8 sequence requires the next 11 bits to have a value in the range 0080-07ff.
   b. Allocate bits from the bitstring to fill the vacant positions 'x' in the UTF-8 sequence (see Table 1) from left to right.
   c. UTF-8 sequences of 1, 2, 3, and 4 octets require 7, 11, 16, and 21 bits, respectively, from the bitstring. Since the number of combinations of UTF-8 sequences accommodating exactly 32 or 128 bits is limited, in sequences where the number of bits required does not exactly match the number of available bits, the final UTF-8 sequence **MUST** be padded with additional bits once the available address bits are exhausted. The sequence may therefore require up to 20 bits of padding. The content of the padding **SHOULD** be chosen to maximize the resulting Confusion Level.
3. Once the bits in the bitstring are exhausted, the conversion is complete. The I-DUNNO representation of the address consists of the Unicode code points described by the list of generated UTF-8 sequences, and it **MAY** now be presented to unsuspecting humans.

3.2. Deforming I-DUNNO

This section is intentionally omitted. The machines will know how to do it, and by definition humans **SHOULD NOT** attempt the process.

4. I-DUNNO Confusion Level Requirements

A sequence of characters is considered I-DUNNO only when there's enough potential to confuse humans.

Unallocated code points **MUST** be avoided. While they might appear to have great confusion power at the moment, there's a minor chance that a future allocation to a useful, legible character will reduce this capacity significantly. Worse, in the (unlikely, but not impossible -- see Section 3.1.3 of [RFC5894]) event of a code point losing its DISALLOWED property per IDNA2008 [RFC5894], existing I-DUNNOs could be rendered less than minimally confusing, with disastrous consequences.

The following Confusion Levels are defined:

4.1. Minimum Confusion Level

As a minimum, a valid I-DUNNO **MUST**:

- Contain at least one UTF-8 octet sequence with a length greater than one octet.
- Contain at least one character that is DISALLOWED in IDNA2008. No code point left behind! Note that this allows machines to distinguish I-DUNNO from Internationalized Domain Name labels.

I-DUNNOs on this level will at least puzzle most human users with knowledge of the Legacy Notation.

4.2. Satisfactory Confusion Level

An I-DUNNO with Satisfactory Confusion Level **MUST** adhere to the Minimum Confusion Level, and additionally contain two of the following:

- At least one non-printable character.
- Characters from at least two different Scripts.
- A character from the "Symbol" category.

The Satisfactory Confusion Level will make many human-machine interfaces beep, blink, silently fail, or any combination thereof. This is considered sufficient to discourage most humans from deforming I-DUNNO.

4.3. Delightful Confusion Level

An I-DUNNO with Delightful Confusion Level **MUST** adhere to the Satisfactory Confusion Level, and additionally contain at least two of the following:

- Characters from scripts with different directionalities.
- Character classified as "Confusables".
- One or more emoji.

An I-DUNNO conforming to this level will cause almost all humans to U+1F926, with the exception of those subscribed to the idna-update mailing list.

(We have also considered a further, higher Confusion Level, tentatively entitled "BReak EXaminatIon or Twiddling" or "BREXIT" Level Confusion, but currently we have no idea how to go about actually implementing it.)

5. Example

An I-DUNNO based on the Legacy Notation IPv4 address "198.51.100.164" is formed and validated as follows: First, the Legacy Notation is written as a string of 32 bits in network byte order:

```
                    11000110001100110110010010100100
```

Since I-DUNNO requires at least one UTF-8 octet sequence with a length greater than one octet, we allocate bits in the following form:

```
                      seq1  |   seq2  |   seq3  |   seq4
                    --------+---------+---------+------------
                    1100011 | 0001100 | 1101100 | 10010100100
```

This translates into the following code points:

Bit Seq.	Character Number (Character Name)
1100011	U+0063 (LATIN SMALL LETTER C)
0001100	U+000C (FORM FEED (FF))
1101100	U+006C (LATIN SMALL LETTER L)
10010100100	U+04A4 (CYRILLIC CAPITAL LIGATURE EN GHE)

Table 2

The resulting string **MUST** be evaluated against the Confusion Level Requirements before I-DUNNO can be declared. Given the example above:

- There is at least one UTF-8 octet sequence with a length greater than 1 (U+04A4) .
- There are two IDNA2008 DISALLOWED characters: U+000C (for good reason!) and U+04A4.
- There is one non-printable character (U+000C).
- There are characters from two different Scripts (Latin and Cyrillic).

Therefore, the example above constitutes valid I-DUNNO with a Satisfactory Confusion Level. U+000C in particular has great potential in environments where I-DUNNOs would be sent to printers.

6. IANA Considerations

If this work is standardized, IANA is kindly requested to revoke all IPv4 and IPv6 address range allocations that do not allow for at least one I-DUNNO of Delightful Confusion Level. IPv4 prefixes are more likely to be affected, hence this can easily be marketed as an effort to foster IPv6 deployment.

Furthermore, IANA is urged to expand the Internet TLA Registry [RFC5513] to accommodate Seven-Letter Acronyms (SLA) for obvious reasons, and register 'I-DUNNO'. For that purpose, U+002D ("-", HYPHEN-MINUS) **SHALL** be declared a Letter.

7. Security Considerations

I-DUNNO is not a security algorithm. Quite the contrary -- many humans are known to develop a strong feeling of insecurity when confronted with I-DUNNO.

In the tradition of many other RFCs, the evaluation of other security aspects of I-DUNNO is left as an exercise for the reader.

8. References

8.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <https://www.rfc-editor.org/info/rfc3629>.

[RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale", RFC 5894, DOI 10.17487/RFC5894, August 2010, <https://www.rfc-editor.org/info/rfc5894>.

[RFC6919] Barnes, R., Kent, S., and E. Rescorla, "Further Key Words for Use in RFCs to Indicate Requirement Levels", RFC 6919, DOI 10.17487/RFC6919, April 2013, <https://www.rfc-editor.org/info/rfc6919>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

8.2. Informative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, DOI 10.17487/RFC1123, October 1989, <https://www.rfc-editor.org/info/rfc1123>.

[RFC1883] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, DOI 10.17487/RFC1883, December 1995, <https://www.rfc-editor.org/info/rfc1883>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC5513] Farrel, A., "IANA Considerations for Three Letter Acronyms", RFC 5513, DOI 10.17487/RFC5513, April 2009, <https://www.rfc-editor.org/info/rfc5513>.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <https://www.rfc-editor.org/info/rfc5952>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8369] Kaplan, H., "Internationalizing IPv6 Using 128-Bit Unicode", RFC 8369, DOI 10.17487/RFC8369, April 2018, <https://www.rfc-editor.org/info/rfc8369>.

[UNICODE] The Unicode Consortium, "The Unicode Standard (Current Version)", 2019, <http://www.unicode.org/versions/latest/>.

Authors' Addresses

Alexander Mayrhofer
nic.at GmbH
Email: alexander.mayrhofer@nic.at
URI: https://i-dunno.at/

Jim Hague
Sinodun
Email: jim@sinodun.com
URI: https://www.sinodun.com/