Network Working Group
Internet-Draft
Updates: 5890, 5891, 5894 (if approved)
Intended status: Standards Track
Expires: March 1, 2020

J. Klensin
A. Freytag
ASMUS, Inc.

August 29, 2019

Internationalized Domain Names in Applications (IDNA): Registry Restrictions and Recommendations
draft-klensin-idna-rfc5891bis-05

Abstract

The IDNA specifications for internationalized domain names combine rules that determine the labels that are allowed in the DNS without violating the protocol itself and an assignment of responsibility, consistent with earlier specifications, for determining the labels that are allowed in particular zones. Conformance to IDNA by registries and other implementations requires both parts. Experience strongly suggests that the language describing those responsibilities was insufficiently clear to promote safe and interoperable use of the specifications and that more details and discussion of circumstances would have been helpful. Without making any substantive changes to IDNA, this specification updates two of the core IDNA documents (RFCs 5890 and 5891) and the IDNA explanatory document (RFC 5894) to provide that guidance and to correct some technical errors in the descriptions.

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 https://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 March 1, 2020.

Table of Contents

1. Introduction

Parts of the specifications for Internationalized Domain Names in Applications (IDNA) [RFC5890] [RFC5891] [RFC5894] (collectively known, along with RFC 5892 [RFC5892], RFC 5893 [RFC5893] and updates to them, as "IDNA2008" (or just "IDNA") impose a requirement that domain name system (DNS) registries restrict the characters they allow in domain name labels (see Section 2 below), and the contents and structure of those labels. That requirement and restriction are

consistent with the "duty to serve the community" described in the original specification for DNS naming and authority [RFC1591]. The restrictions are intended to limit the permitted characters and strings to those for which the registries or their advisers have a thorough understanding and for which they are willing to take responsibility.

That provision is centrally important because it recognized that historical relationships and variations among scripts and writing systems, the continuing evolution of those systems, differences in the uses of characters among languages (and locations) that use the same script, and so on make it impossible for a single list of characters and simple rules to be able to generate an "if we use these, we will be safe from confusion and various attacks" guideline.

Instead, the algorithm and rules of RFCs 5891 and 5892 eliminate many of the most dangerous and otherwise problematic cases, but cannot eliminate the need for registries and registrars to understand what they are doing and taking responsibility for the decisions they make.

The way in which the IDNA2008 specifications expressed these requirements may have under emphasized the intention that they actually are requirements. Section 2.3.2.3 of the Definitions document [RFC5890] mentions the need for the restrictions, indicates that they are mandatory, and points the reader to section 4.3 of the Protocol document [RFC5891], which in turn points to Section 3.2 of the Rationale document [RFC5894], with each document providing further detail, discussion, and clarification.

At the same time, the Internet has evolved significantly since the management assumptions for the DNS were established with RFC 1591 and earlier. In particular, the management and use of domain names have gone through several transformations. Recounting of those changes is beyond the scope of this document but one of them has had significant practical impact on the degree to which the requirement for registry knowledge and responsibility is observed in practice. When RFC 1591 was written, the assumption was that domains at all levels of the DNS would be operated in the best interest of the registrants in the domain and of the Internet as a whole. There were no notions about domains being operated for a profit, much less with a business model that made them more profitable the more names that could be registered (or even, under some circumstances, reserved and not registered). At the time RFC 1591 was written, there was also no notion that domains would be considered more successful based on the number of names registered and delegated from them. While rarely reflected in the DNS protocols, the distinction between domains operated in those ways and ones that are operated for, e.g., use within an enterprise or otherwise as a service have become very

important today. See Section 4 for a discussion on how those issues affect this specification.

This specification is intended to unify and clarify these requirements for registry decisions and responsibility and to emphasize the importance of registry restrictions at all levels of the DNS. It also makes a specific recommendation for character repertoire subsetting intermediate between the code points allowed by RFCs 5891 and 5892 and those allowed by individual registries. It does not alter the basic IDNA2008 protocols and rules themselves in any way.

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].

2. Registry Restrictions in IDNA2008

As mentioned above, IDNA2008 specifies that the registries for each zone in the DNS that supports IDN labels are required to develop and apply their own rules to restrict the allowable labels, including limiting characters they allow to be used in labels in that zone. The chosen list MUST be a subset of the collection of code points specified as "PVALID", "CONTEXTJ", and "CONTEXTO" by the rules established by the protocols themselves. Labels containing any characters from the two CONTEXT categories or any characters that are normally part of a script written right to left [RFC5893] require that additional rules, specified in the protocols and known as "contextual rules" and "bidi rules", be applied. The entire collection of rules and restrictions required by the IDNA2008 protocols themselves are known as "protocol restrictions".

As mentioned above, registries may apply (and generally are required to apply) additional rules to further restrict the list of permitted code points, contextual rules (perhaps applied to normally PVALID code points) that apply additional restrictions, and/or restrictions on labels as distinct from code points. The most obvious of those restrictions include provisions for restricting suggested new registrations based on conflicts with labels already registered in the zone, so as to avoid homograph attacks and other issues. The specifications of what constitutes such conflicts, as well as the definition of "conflict" based on the properties of the labels in question, is the responsibility of each registry. They further include prohibitions on code points and labels that are not consistent with the intended function of the zone, the subtree in which the zone is embedded (see Section 3), or limitations on where allowable code points may be placed in a label.

These per-registry (or per-zone) rules are commonly known as "registry restrictions" to distinguish them from the protocol restrictions described above. By necessity, protocol restrictions are somewhat generic, having to cater both to the union of the needs for all zones as well as to the desires of the most permissive zones. In consequence, additional registry restrictions are essential to provide for the necessary security in the face of the tremendous variations and differences in writing systems, their ongoing evolution and development, as well as the human ability to recognize and distinguish characters in different scripts around the world and under different circumstances.

3. Progressive Subsets of Allowed Characters

The algorithm and rules of RFCs 5891 and 5892 determine the set of code points that are possible for inclusion in domain name labels; registries MUST NOT permit code points in labels unless they are part of that set. Labels that contain code points that are normally written from right to left MUST also conform to the requirements of RFC 5893. Each registry that intends to allow IDN registrations MUST then determine the strict subset of that set of code points that will be allowed by that registry. It SHOULD also consider additional rules, including contextual and whole label restrictions that provide further protection for registrants and users. For example, the widely-used principle that bars labels containing characters from more than one script is not an IDNA2008 requirement. It has been adopted by many registries but, as Section 4.4 of RFC 5890 indicates, there may be circumstances in which is it not required or appropriate.

In formulating their own rules, registries SHOULD normally consult carefully-developed consensus recommendations about global maximum repertoires to be used such as the ICANN Maximal Starting Repertoire 4 (MSR-4) for the Development of Label Generation Rules for the Root Zone [ICANN-MSR4] (or its successor documents). Additional recommendations of similar quality about particular scripts or languages exist, including, but not limited to, the RFCs for Cyrillic [RFC5992], Arabic Language [RFC5564], or script-based repertoires from the approved ICANN Root Zone Label Generation Rules (LGR-3) [ICANN-LGR3] (or its successor documents). Many of these recommendations also cover rules about relationships among code points that may be particularly important for complex scripts. They also interact with recommendations about how labels that appear to be the same or apparently the same should be handled.

It is the responsibility of the registry to determine which, if any, of those recommendations are applicable and to further subset or extend them as needed. For example, several of the recommendations

are designed for the root zone and therefore exclude digits and U+002D HYPHEN-MINUS; this restriction is not generally appropriate for other zones. On the other hand, some zones may be designed to not cater for all users of a given script, but perhaps only for the needs of selected languages, in which case a more selective repertoire may be appropriate.

In making these determinations, a registry SHOULD follow the IAB guidance in RFC 6912 [RFC6912]. Those guidelines include a number of principles for use in making decisions about allowable code points. In addition, that document notes that the closer a particular zone is to the root, the more restrictive the space of permitted labels should be. RFC 5894 provides some suggestions for any registry that may decide to reduce opportunities for confusion or attacks by constructing policies that disallow characters used in historic writing systems (whether these be archaic scripts or extensions of modern scripts for historic or obsolete orthographies) or characters whose use is restricted to specialized, or highly technical contexts. These suggestions were among the principles guiding the design of ICANN's Maximal Starting Repertoires (MSR) [LGR-Procedure].

A registry decision to allow only those code points in the full repertoire of the MSR (plus digits and hyphen) would already avoid a number of issues inherent in a more permissive policy such as "use anything permitted by IDNA2008", while still supporting the native languages and scripts for the vast majority of users today. However, it is unlikely, by itself, to fully satisfy the mandate set out above for three reasons.

1. The MSR, like the set of code points permissible under IDNA2008 itself, was conceived merely as a boundary condition on permissible letter code points (it excludes digits and the hyphen). It was always intended to be used as a starting point for setting registry policy, with the expectation that some of the code points in the MSR would not be included in the final registry policy, whether for lack of actual usage, or for being inherently problematic.

2. It was recognized that many scripts require contextual rules for many more code points than are covered by CONTEXTO or CONTEXTJ rules defined in IDNA2008. This is particularly true for combining marks, typically used to encode diacritics, tone marks, vowel signs and the like. While, theoretically, any combining mark may occur in any context in Unicode, in practice rendering and other software that users rely on in viewing or entering labels will not support arbitrary combining sequences, or indeed arbitrary combinations of code points, in the case of complex scripts.

Contextual rules are needed in order to limit allowable code point sequences to those that can be expected to be rendered reliably. Identifying those requires knowledge about the way code points are used in a script, whence the mandate for registries to only support code points they understand. In this, some of the other recommendations, such as the Informational RFCs for specific scripts (e.g., Cyrillic [RFC5992]) or languages (e.g., Arabic [RFC5564] or Chinese [RFC4713]), or the Root Zone LGRs developed by ICANN, may provide useful guidance.

3. Third, because of the widely accepted practice of limiting any given label to a single script, a universal repertoire, such as the MSR, would have to be divided on a per-script basis into subrepertoires to make it useful, with some of those repertoires overlapping, for example, in the case of East Asian shared usage of the Han ideographs.

Registries choosing to make exceptions -- allow code points that recommendations such as the MSR do not allow -- should make such decisions only with great care and only if they have considerable understanding of, and great confidence in, their appropriateness. The obvious exception from the MSR would be to allow digits and the hyphen. Neither were allowed by the MSR, but only because they are not allowed in the Root Zone.

Nothing in this document permits a registry to allow code points or labels that are disallowed or otherwise prohibited by IDNA2008.

4. Considerations for For-Profit Domains

As discussed in the Introduction (Section 1), the distributed administrative structure of the DNS today can be described by dividing zones into two categories depending on how they are administered and for whom. These categories are not precise -- some zones may not fall neatly into one category or the other -- but are useful in understanding the practical applicability of this specification. They are:

Zones operating primarily or exclusively within an organization or enterprise and responsible to that organization or enterprise. DNS operations, including registrations and delegations, will typically occur in support of the purpose of that organization or enterprise rather than being its primary purpose.

Zones operating primarily on a for-profit basis in which most delegations of subdomains are to entities with little or no affiliation with the registry operator other than contractual agreements about operation of those subdomains. These zones are

often known as "public domains" or with similar terms, but those terms often have other semantics and may not cover all cases.

Rules requiring strict registry responsibility, including either thorough understanding of scripts and related issues in domain name labels being considered for registration or local naming rules that have the same effect, typically come naturally to registries for zones of the first type. Registration of labels that would prove problematic for any reason hurts the relevant organization or enterprise or its customers. More generally, there are strong incentives to be extremely conservative about labels that might be registered and few, if any, incentives favoring adventures into labels that might be considered clever, much less ones that are hard to type, render, or, where it is relevant to users, remember correctly.

By contrast, in a for-profit zone in which the profits are limited to selling names, there may be perceived incentives to register whatever names would-be registrants "want" or fears that any restrictions will cut into the available namespace. In such situations, restrictions are unlikely to be applied unless they meet at least one of two criteria: (i) they are easy to apply and can be applied algorithmically or otherwise automatically and/or (ii) there is clear evidence that the particular label would cause harm.

As suggested above, the two categories above are not precise. In particular, there may be domains that, despite being set up to operate at a profit, are sufficiently conservative about their operations to more closely resemble the first group in practice than the second one.

The requirement of IDNA that is discussed at length elsewhere in this specification stands: IDNA (and IDNs generally) would work better and Internet users would be better protected and more secure if registries and registrars (of any type) confined their registrations to scripts and code point sequences that they understood thoroughly. While the IETF rarely gives advice to those who choose to violate IETF Standards, some advice to zones in the second category above may be in order. That advice is that significant conservatism in what is allowed to be registered, even for reservation purposes, and even more conservatism about what labels are actually entered into zones and delegated, is the best option for the Internet and its users. If practical considerations do not allow that much conservatism, then it is desirable to consult and utilize the many lists and tables that have been, and continue to be, developed to advise on what might be sensible for particular scripts and languages. These include ICANN's twin efforts of creating per-script Root Zone Label Generation Rules [RZ-LGR-3] and Second Level Reference Label Generation Rules

[SL-REF-LGR] (the latter of which may be per language). They also include other lists of code points or code point relationships that may be particularly problematic and that should be treated with extra caution or prohibited entirely such as the proposed "troublesome character" list [Freytag-troublesome]. See also Section 6 below.

5. Other corrections and updates

After the initial IDNA2008 documents were published (and RFC 5892 was updated for Unicode 6.0 by RFC 6452 [RFC6452]) several errors or instances of confusing text were noted. For the convenience of the community, the relevant corrections for RFCs 5890 and 5891 are noted below and update the corresponding documents. There are no errata for RFC 5893 or 5894 as of the date this document was published. Because further updates to RFC 5892 would require addressing other pending issues, the outstanding erratum for that document is not considered here. For consistency with the original documents, references to Unicode 5.0 are preserved in this document.

Readers should note that an update to RFC 5892 that is primarily concerned with the review process for new versions of Unicode but that makes some additional patches [ID.draft-klensin-idna-unicode-review] is in progress. Its status should be checked in conjunction with application of the present specification.

5.1. Updates to RFC 5890

The outstanding errata against RFC 5890 (Errata ID 4695, 4696, 4823, and 4824 [RFC-Editor-5890Errata]) are all associated with the same issue, the number of Unicode characters that can be associated with a maximum-length (63 octet) A-label. In retrospect and contrary to some of the suggestions in the errata, that value should not be expressed in octets because RFC 5890 and the other IDNA 2008 documents are otherwise careful to not specify Unicode encoding forms but, instead, work exclusively with Unicode code points. Consequently the relevant material in RFC 5890 should be corrected as follows:

Section 2.3.2.1

Old: expansion of the A-label form to a U-label may produce strings that are much longer than the normal 63 octet DNS limit (potentially up to 252 characters).

New: expansion of the A-label form to a U-label may produce strings that are much longer than the normal 63 octet DNS limit (See Section 4.2).

Comment: If the length limit is going to be a source of confusion or careful calculations, it should appear in only one place.

Section 4.2

Old: Because A-labels (the form actually used in the DNS) are potentially much more compressed than UTF-8 (and UTF-8 is, in general, more compressed that UTF-16 or UTF-32), U-labels that obey all of the relevant symmetry (and other) constraints of these documents may be quite a bit longer, potentially up to 252 characters (Unicode code points).

New: A-labels (the form actually used in the DNS) and the Punycode algorithm used as part of the process to produce them [RFC3492] are strings that are potentially much more compressed than any standard Unicode Encoding Form. A 63 octet A-label cannot represent more than 58 Unicode code points (four octet overhead and the requirement that at least one character lie outside the ASCII range) but implementations allocating buffer space for the conversion should allow significantly more space depending on the encoding form they are using.

5.2. Updates to RFC 5891

Errata ID 3969: Improve reference for combining marks. There is only one erratum for RFC 5891, Errata ID 3969 [RFC5891Erratum]. Combining marks are explained in the cited section, but not, as the text indicates, exactly defined.

Old: The Unicode string MUST NOT begin with a combining mark or combining character (see The Unicode Standard, Section 2.11 [UnicodeA] for an exact definition).

New: The Unicode string MUST NOT begin with a combining mark or combining character (see The Unicode Standard, Section 2.11 [Unicode] for an explanation and Section 3.6, definition D52) for an exact definition).

Comment: When RFC 5891 is actually updated, the references in the text should be updated to the current version of Unicode and the section numbers checked.

6. Related Discussions

This document is one of a series of measures that have been suggested to address IDNA issues raised in other documents, including mechanisms for dealing with combining sequences and single-code point characters with the same appearance that normalization neither

combines nor decomposes as IDNA2008 assumed [IDNA-Unicode], including the IAB response to that issue [IAB-2015], and to take a higher-level view of issues, demands, and proposals for new uses of the DNS. Those documents also include a discussion of issues with IDNA and character graphemes for which abstractions exist in Unicode in precomposed form but that can be generated from combining sequences and a suggested registry of code points known to be problematic [Freytag-troublesome]. The discussion of combining sequences and non-decomposing characters is intended to lay the foundation for an actual update to the IDNA code points document [RFC5892]. Such an update will presumably also address the existing errata against that document.

7. Security Considerations

As discussed in IAB recommendations about internationalized domain names [RFC4690], [RFC6912], and elsewhere, poor choices of strings for DNS labels can lead to opportunities for attacks, user confusion, and other issues less directly related to security. This document clarifies the importance of registries carefully establishing design policies for the labels they will allow and that having such policies and taking responsibility for them is a requirement, not an option. If that clarification is useful in practice, the result should be an improvement in security.

8. Acknowledgments

Many thanks to Patrik Faltstrom who provided an important review on the initial version, to Jaap Akkerhuis, Don Eastlake, Barry Leiba, and Alessandro Vesely who did reviews that improved the text and to Pete Resnick who acted as document shepherd and did an additional careful review.

9. IANA Considerations

[[CREF1: RFC Editor: Please remove this section before publication.]]

This memo includes no requests to or actions for IANA. In particular, it does not contain any provisions that would alter any IDNA-related registries or tables.

10. References

10.1. Normative References

[ICANN-LGR3]
ICANN, "Root Zone Label Generation Rules (LGR-1)", July 2019, <https://www.icann.org/news/announcement-2-2019-04-25-en>.

[ICANN-MSR4]
ICANN, "Maximal Starting Repertoire Version 4 (MSR-4) for the Development of Label Generation Rules for the Root Zone", January 2019, <https://www.icann.org/news/announcement-2019-02-07-en>.

[RFC1591] Postel, J., "Domain Name System Structure and Delegation", RFC 1591, DOI 10.17487/RFC1591, March 1994, <https://www.rfc-editor.org/info/rfc1591>.

[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>.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <https://www.rfc-editor.org/info/rfc5890>.

[RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, DOI 10.17487/RFC5891, August 2010, <https://www.rfc-editor.org/info/rfc5891>.

[RFC5891Erratum]
"RFC 5891, "Internationalized Domain Names in Applications (IDNA): Protocol"", Errata ID 3969, April 2014, <http://www.rfc-editor.org/errata_search.php/doc/html/rfc5891>.

[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>.

10.2. Informative References

[Freytag-troublesome]
Freytag, A., Klensin, J., and A. Sullivan, "Those Troublesome Characters: A Registry of Unicode Code Points Needing Special Consideration When Used in Network Identifiers", June 2017, <draft-freytag-troublesome-characters-01>.

[IAB-2015]
Internet Architecture Board (IAB), "IAB Statement on Identifiers and Unicode 7.0.0", February 2015, <https://www.iab.org/documents/correspondence-reports-documents/2015-2/iab-statement-on-identifiers-and-unicode-7-0-0/>.

[ID.draft-klensin-idna-unicode-review]
Klensin, J. and P. Faltstrom, "IDNA Review for New Unicode Versions", June 2019, <https://datatracker.ietf.org/doc/draft-klensin-idna-unicode-review/>.

[IDNA-Unicode]
Klensin, J. and P. Faltstrom, "IDNA Update for Unicode 7.0.0", September 2017, <draft-klensin-idna-5892upd-unicode70-05>.

[LGR-Procedure]
Internet Corporation for Assigned Names and Numbers (ICANN), "Procedure to Develop and Maintain the Label Generation Rules for the Root Zone in Respect of IDNA Labels", March 2013, <https://www.icann.org/en/system/files/files/draft-lgr-procedure-20mar13-en.pdf>.

[RFC-Editor-5890Errata]
RFC Editor, "RFC Errata: RFC 5890, "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", August 2010", Note to RFC Editor: Please figure out how you would like this referenced and make it so., Captured 2017-09-10, 2016, <https://www.rfc-editor.org/errata_search.php/doc/html/rfc5890>.

[RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, DOI 10.17487/RFC3492, March 2003, <https://www.rfc-editor.org/info/rfc3492>.

[RFC4690] Klensin, J., Faltstrom, P., Karp, C., and IAB, "Review and Recommendations for Internationalized Domain Names (IDNs)", RFC 4690, DOI 10.17487/RFC4690, September 2006, <https://www.rfc-editor.org/info/rfc4690>.

[RFC4713] Lee, X., Mao, W., Chen, E., Hsu, N., and J. Klensin, "Registration and Administration Recommendations for Chinese Domain Names", RFC 4713, DOI 10.17487/RFC4713, October 2006, <https://www.rfc-editor.org/info/rfc4713>.

[RFC5564] El-Sherbiny, A., Farah, M., Oueichek, I., and A. Al-Zoman, "Linguistic Guidelines for the Use of the Arabic Language in Internet Domains", RFC 5564, DOI 10.17487/RFC5564, February 2010, <https://www.rfc-editor.org/info/rfc5564>.

[RFC5892] Faltstrom, P., Ed., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)", RFC 5892, DOI 10.17487/RFC5892, August 2010, <https://www.rfc-editor.org/info/rfc5892>.

[RFC5893] Alvestrand, H., Ed. and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)", RFC 5893, DOI 10.17487/RFC5893, August 2010, <https://www.rfc-editor.org/info/rfc5893>.

[RFC5992] Sharikov, S., Miloshevic, D., and J. Klensin, "Internationalized Domain Names Registration and Administration Guidelines for European Languages Using Cyrillic", RFC 5992, DOI 10.17487/RFC5992, October 2010, <https://www.rfc-editor.org/info/rfc5992>.

[RFC6452] Faltstrom, P., Ed. and P. Hoffman, Ed., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA) - Unicode 6.0", RFC 6452, DOI 10.17487/RFC6452, November 2011, <https://www.rfc-editor.org/info/rfc6452>.

[RFC6912] Sullivan, A., Thaler, D., Klensin, J., and O. Kolkman, "Principles for Unicode Code Point Inclusion in Labels in the DNS", RFC 6912, DOI 10.17487/RFC6912, April 2013, <https://www.rfc-editor.org/info/rfc6912>.

[RZ-LGR-3]
Internet Corporation for Assigned Names and Numbers, "Root Zone Label Generation Rules - LGR-3: Overview and Summary, Version 3", July 2019, <https://www.icann.org/sites/default/files/lgr/lgr-3-overview-10jul19-en.pdf>.

[SL-REF-LGR]
Internet Corporation for Assigned Names and Numbers (ICANN), "Second Level Label Generation Rules", 2019, <https://www.icann.org/resources/pages/second-level-lgr-2015-06-21-en>.

[UnicodeA]
The Unicode Consortium, "The Unicode Standard, Version 12.1", May 2019.

Section 2.11

Appendix A. Change Log

RFC Editor: Please remove this appendix before publication.

A.1. Changes from version -00 (2017-03-11) to -01

- Added Acknowledgments and adjusted references.
- Filled in Section 5 with updates to respond to errata.
- Added Section 6 to discuss relationships to other documents.
- Modified the Abstract to note specifically updated documents.
- Several small editorial changes and corrections.

A.2. Changes from version -01 (2017-09-12) to -02

After a pause of nearly 34 months due to inability to get this draft processed, including nearly a year waiting for a new directorate to actually do anything of substance about fundamental IDNA issues, the -02 version was posted in the hope of getting a new start. Specific changes include:

- Added a new section, Section 4, and some introductory material to address the very practical issue that domains run on a for-profit basis are unlikely to follow the very strict "understand what you are registering" requirement if they support IDNs at all and expect to profit from them.
- Added a pointer to draft-klensin-idna-unicode-review to the discussion of other work.
- Editorial corrections and changes.

A.3. Changes from version -02 (2019-07-06) to -03

- Minor editorial changes in response to shepherd review.
- Additional references.

A.4. Changes from version -03 (2019-07-22) to -04

- Editorial changes after AD review and some additional changes to improve clarity.

A.5. Changes from version -04 (2019-08-02) to -05

- Small editorial corrections, many to correct glitches found during IETF Last Call.
- Updated acknowledgments, particularly to reflect reviews in Last Call.

Authors' Addresses

John C Klensin
1770 Massachusetts Ave, Ste 322
Cambridge, MA 02140
USA

Phone: +1 617 245 1457
Email: john-ietf@jck.com

Asmus Freytag
ASMUS, Inc.

Email: asmus@unicode.org