

Network Working Group
Internet-Draft
Obsoletes: [3490](#), [3491](#)
(if approved)
Updates: [3492](#) (if approved)
Intended status: Standards Track
Expires: July 10, 2010

J. Klensin
January 6, 2010

**Internationalized Domain Names in Applications (IDNA): Protocol
draft-ietf-idnabis-protocol-18.txt**

Abstract

This document is the revised protocol definition for internationalized domain names (IDNs). The rationale for changes, the relationship to the older specification, and important terminology are provided in other documents. This document specifies the protocol mechanism, called Internationalized Domain Names in Applications (IDNA), for registering and looking up IDNs in a way that does not require changes to the DNS itself. IDNA is only meant for processing domain names, not free text.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on July 10, 2010.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1.	Introduction	4
1.1.	Discussion Forum	4
2.	Terminology	4
3.	Requirements and Applicability	5
3.1.	Requirements	5
3.2.	Applicability	5
3.2.1.	DNS Resource Records	6
3.2.2.	Non-domain-name Data Types Stored in the DNS	6
4.	Registration Protocol	6
4.1.	Input to IDNA Registration	7
4.2.	Permitted Character and Label Validation	7
4.2.1.	Input Format	7
4.2.2.	Rejection of Characters that are not Permitted	8
4.2.3.	Label Validation	8
4.2.4.	Registration Validation Requirements	9
4.3.	Registry Restrictions	9
4.4.	Punycode Conversion	9
4.5.	Insertion in the Zone	10
5.	Domain Name Lookup Protocol	10
5.1.	Label String Input	10
5.2.	Conversion to Unicode	10
5.3.	A-label Input	10
5.4.	Validation and Character List Testing	11
5.5.	Punycode Conversion	12
5.6.	DNS Name Resolution	13
6.	Security Considerations	13
7.	IANA Considerations	13
8.	Contributors	13
9.	Acknowledgments	14
10.	References	14
10.1.	Normative References	14
10.2.	Informative References	15
Appendix A.	Summary of Major Changes from IDNA2003	16
Appendix B.	Change Log	17
B.1.	Changes between Version -00 and -01 of draft-ietf-idnabis-protocol	17
B.2.	Version -02	17
B.3.	Version -03	18
B.4.	Version -04	18
B.5.	Version -05	18
B.6.	Version -06	18
B.7.	Version -07	19
B.8.	Version -08	19
B.9.	Version -09	19
B.10.	Version -10	20
B.11.	Version -11	20

Klensin

Expires July 10, 2010

[Page 3]

B.12.	Version -12	20
B.13.	Version -13	21
B.14.	Version -14	21
B.15.	Version -15	21
B.16.	Version -16	22
B.17.	Version -17	22
B.18.	Version -18	23
Author's Address		23

1. Introduction

This document supplies the protocol definition for internationalized domain names. Essential definitions and terminology for understanding this document and a road map of the collection of documents that make up IDNA2008 appear in [[IDNA2008-Defs](#)]. [Appendix A](#) discusses the relationship between this specification and the earlier version of IDNA (referred to here as "IDNA2003"). The rationale for these changes, along with considerable explanatory material and advice to zone administrators who support IDNs is provided in another document, [[IDNA2008-Rationale](#)].

IDNA works by allowing applications to use certain ASCII string labels (beginning with a special prefix) to represent non-ASCII name labels. Lower-layer protocols need not be aware of this; therefore IDNA does not change any infrastructure. In particular, IDNA does not depend on any changes to DNS servers, resolvers, or DNS protocol elements, because the ASCII name service provided by the existing DNS can be used for IDNA.

IDNA applies only to a specific subset of DNS labels. The base DNS standards [[RFC1034](#)] [[RFC1035](#)] and their various updates specify how to combine labels into fully-qualified domain names and parse labels out of those names.

This document describes two separate protocols, one for IDN registration ([Section 4](#)) and one for IDN lookup ([Section 5](#)). These two protocols share some terminology, reference data and operations.

1.1. Discussion Forum

[[RFC Editor: please remove this section.]]

This work is being discussed in the IETF IDNABIS WG and on the mailing list idna-update@alvestrand.no

2. Terminology

Terminology used as part of the definition of IDNA appears in [[IDNA2008-Defs](#)]. It is worth noting that some of this terminology overlaps with, and is consistent with, that used in Unicode or other character set standards and the DNS. Readers of this document are assumed to be familiar with [[IDNA2008-Defs](#)] and with the DNS-specific terminology in [RFC 1034](#) [[RFC1034](#)].

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 [BCP 14](#), [RFC 2119](#) [[RFC2119](#)].

3. Requirements and Applicability

3.1. Requirements

IDNA makes the following requirements:

1. Whenever a domain name is put into an IDN-unaware domain name slot (see [Section 2](#) and [[IDNA2008-Defs](#)]), it MUST contain only ASCII characters (i.e., its labels must be either A-labels or NR-LDH-labels), unless the DNS application is not subject to historical recommendations for "hostname"-style names (see [[RFC1034](#)] and [Section 3.2.1](#)).
2. Labels MUST be compared using equivalent forms: either both A-Label forms or both U-Label forms. Because A-labels and U-labels can be transformed into each other without loss of information, these comparisons are equivalent. A pair of A-labels MUST be compared as case-insensitive ASCII (as with all comparisons of ASCII DNS labels). U-labels MUST be compared as-is, without case-folding or other intermediate steps. Note that it is not necessary to validate labels in order to compare them and that successful comparison does not imply validity. In many cases, not limited to comparison, validation may be important for other reasons and SHOULD be performed.
3. Labels being registered MUST conform to the requirements of [Section 4](#). Labels being looked up and the lookup process MUST conform to the requirements of [Section 5](#).

3.2. Applicability

IDNA applies to all domain names in all domain name slots in protocols except where it is explicitly excluded. It does not apply to domain name slots which do not use the Letter/Digit/Hyphen (LDH) syntax rules.

Because IDNA uses the DNS, IDNA applies to many protocols that were specified before it was designed. IDNs occupying domain name slots in those older protocols MUST be in A-label form until and unless those protocols and their implementations are explicitly upgraded to be aware of IDNs. IDNs actually appearing in DNS queries or responses MUST be A-labels.

IDNA-aware protocols and implementations MAY accept U-labels,

Klensin

Expires July 10, 2010

[Page 6]

A-labels, or both as those particular protocols specify.

IDNA is not defined for extended label types (see [RFC 2671, Section 3 \[RFC2671\]](#)).

3.2.1. DNS Resource Records

IDNA applies only to domain names in the NAME and RDATA fields of DNS resource records whose CLASS is IN. See [RFC 1035 \[RFC1035\]](#) for precise definitions of these terms.

The application of IDNA to DNS resource records depends entirely on the CLASS of the record, and not on the TYPE except as noted below. This will remain true, even as new types are defined, unless a new type defines type-specific rules. Special naming conventions for SRV records (and "underscore labels" more generally) are incompatible with IDNA coding as discussed in [\[IDNA2008-Defs\]](#), especially [Section 2.3.2.3](#). Of course, underscore labels may be part of a domain that uses IDN labels at higher levels in the tree.

3.2.2. Non-domain-name Data Types Stored in the DNS

Although IDNA enables the representation of non-ASCII characters in domain names, that does not imply that IDNA enables the representation of non-ASCII characters in other data types that are stored in domain names, specifically in the RDATA field for types that have structured RDATA format. For example, an email address local part is stored in a domain name in the RNAME field as part of the RDATA of an SOA record (hostmaster@example.com would be represented as hostmaster.example.com). IDNA does not update the existing email standards, which allow only ASCII characters in local parts. Even though work is in progress to define internationalization for email addresses [\[RFC4952\]](#), changes to the email address part of the SOA RDATA would require action in, or updates to, other standards, specifically those that specify the format of the SOA RR.

4. Registration Protocol

This section defines the model for registering an IDN. The model is implementation independent; any sequence of steps that produces exactly the same result for all labels is considered a valid implementation.

Note that, while the registration (this section) and lookup protocols ([Section 5](#)) are very similar in most respects, they are not identical and implementers should carefully follow the steps described in this

specification.

4.1. Input to IDNA Registration

Registration processes, especially processing by entities (often called "registrars") who deal with registrants before the request actually reaches the zone manager ("registry") are outside the scope of this definition and may differ significantly depending on local needs. By the time a string enters the IDNA registration process as described in this specification, it **MUST** be in Unicode and in Normalization Form C (NFC [[Unicode-UAX15](#)]). Entities responsible for zone files ("registries") **MUST** accept only the exact string for which registration is requested, free of any mappings or local adjustments. They **MAY** accept that input in any of three forms:

1. As a pair of A-label and U-label.
2. As an A-label only.
3. As a U-label only.

The first two of these forms are **RECOMMENDED** because the use of A-labels avoids any possibility of ambiguity. The first is normally preferred over the second because it permits further verification of user intent (see [Section 4.2.1](#)).

4.2. Permitted Character and Label Validation

4.2.1. Input Format

If both the U-label and A-label forms are available, the registry **MUST** ensure that the A-label form is in lower case, perform a conversion to a U-label, perform the steps and tests described below on that U-label, and then verify that the A-label produced by the step in [Section 4.4](#) matches the one provided as input. In addition, the U-label that was provided as input and the one obtained by conversion of the A-label **MUST** match exactly. If, for some reason, these tests fail, the registration **MUST** be rejected.

If only an A-label was provided and the conversion to a U-label is not performed, the registry **MUST** still verify that the A-label is superficially valid, i.e., that it does not violate any of the rules of Punycode [[RFC3492](#)] encoding such as the prohibition on trailing hyphen-minus, appearance of non-basic characters before the delimiter, and so on. Strings that appear to be A-labels (e.g., they start with "xn--") and strings that are supplied to the registry in a context (such as a field in a form to be filled out) reserved for A-labels, but that are not valid A-labels as described in this

paragraph, MUST NOT be placed in DNS zones that support IDNA.

If only an A-label is provided, the conversion to a U-label is not performed, and the superficial tests described in the previous paragraph are performed, registration procedures MAY, and usually will, bypass the tests and actions in the balance of [Section 4.2](#) and in [Section 4.3](#) and [Section 4.4](#).

[4.2.2.](#) Rejection of Characters that are not Permitted

The candidate Unicode string MUST NOT contain characters that appear in the "DISALLOWED" and "UNASSIGNED" lists specified in [\[IDNA2008-Tables\]](#).

[4.2.3.](#) Label Validation

The proposed label (in the form of a Unicode string, i.e., a string that at least superficially appears to be a U-label) is then examined, performing tests that require examination of more than one character. Character order is considered to be the on-the-wire order, not the display order.

[4.2.3.1.](#) Hyphen Restrictions

The Unicode string MUST NOT contain "--" (two consecutive hyphens) in the third and fourth character positions and MUST NOT start or end with a "-" (hyphen).

[4.2.3.2.](#) Leading Combining Marks

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

[4.2.3.3.](#) Contextual Rules

The Unicode string MUST NOT contain any characters whose validity is context-dependent, unless the validity is positively confirmed by a contextual rule. To check this, each code-point marked as CONTEXTJ or CONTEXTO in [\[IDNA2008-Tables\]](#) MUST have a non-null rule. If such a code-point is missing a rule, it is invalid. If the rule exists but the result of applying the rule is negative or inconclusive, the proposed label is invalid.

[4.2.3.4.](#) Labels Containing Characters Written Right to Left

If the proposed label contains any characters that are written from right to left it MUST meet the BIDI criteria [\[IDNA2008-BIDI\]](#).

[4.2.4.](#) Registration Validation Requirements

Strings that contain at least one non-ASCII character, have been produced by the steps above, whose contents pass all of the tests in [Section 4.2.3](#), and are 63 or fewer characters long in ACE form (see [Section 4.4](#)), are U-labels.

To summarize, tests are made in [Section 4.2](#) for invalid characters, invalid combinations of characters, for labels that are invalid even if the characters they contain are valid individually, and for labels that do not conform to the restrictions for strings containing right to left characters.

[4.3.](#) Registry Restrictions

In addition to the rules and tests above, there are many reasons why a registry could reject a label. Registries at all levels of the DNS, not just the top level, are expected to establish policies about label registrations. Policies are likely to be informed by the local languages and the scripts that are used to write them and may depend on many factors including what characters are in the label (for example, a label may be rejected based on other labels already registered). See [[IDNA2008-Rationale](#)] [Section 3.2](#) for a discussion and recommendations about registry policies.

The string produced by the steps in [Section 4.2](#) is checked and processed as appropriate to local registry restrictions. Application of those registry restrictions may result in the rejection of some labels or the application of special restrictions to others.

[4.4.](#) Punycode Conversion

The resulting U-label is converted to an A-label (defined in [Section 2.3.2.1](#) of [[IDNA2008-Defs](#)]). The A-label is the encoding of the U-label according to the Punycode algorithm [[RFC3492](#)] with the ACE prefix "xn--" added at the beginning of the string. The resulting string must, of course, conform to the length limits imposed by the DNS. This document does not update or alter the Punycode algorithm specified in [RFC 3492](#) in any way. That document does make a non-normative reference to the information about the value and construction of the ACE prefix that appears "in [RFC 3490](#) or Nameprep [[RFC3491](#)]". For consistency and reader convenience, IDNA2008 effectively updates that reference to point to this document. That change does not alter the prefix itself. The prefix, "xn--", is the same in both sets of documents.

With the exception of the maximum string length test on Punycode output, the failure conditions identified in the Punycode encoding

procedure cannot occur if the input is a U-label as determined by the steps in [Section 4.1](#) through [Section 4.3](#) above.

4.5. Insertion in the Zone

The label is registered in the DNS by inserting the A-label into a zone.

5. Domain Name Lookup Protocol

Lookup is different from registration and different tests are applied on the client. Although some validity checks are necessary to avoid serious problems with the protocol, the lookup-side tests are more permissive and rely on the assumption that names that are present in the DNS are valid. That assumption is, however, a weak one because the presence of wild cards in the DNS might cause a string that is not actually registered in the DNS to be successfully looked up.

5.1. Label String Input

The user supplies a string in the local character set, for example by typing it or clicking on, or copying and pasting, a resource identifier, e.g., a URI [[RFC3986](#)] or IRI [[RFC3987](#)] from which the domain name is extracted. Alternately, some process not directly involving the user may read the string from a file or obtain it in some other way. Processing in this step and that specified in [Section 5.2](#) are local matters, to be accomplished prior to actual invocation of IDNA.

5.2. Conversion to Unicode

The string is converted from the local character set into Unicode, if it is not already in Unicode. Depending on local needs, this conversion may involve mapping some characters into other characters as well as coding conversions. Those issues are discussed in [[IDNA2008-Mapping](#)] and the mapping-related sections (Sections [4.4](#), [6](#), and [7.3](#)) of [[IDNA2008-Rationale](#)]. The result MUST be a Unicode string in NFC form.

5.3. A-label Input

If the input to this procedure appears to be an A-label (i.e., it starts in "xn--", interpreted case-insensitively), the lookup application MAY attempt to convert it to a U-label, first ensuring that the A-label is entirely in lower case (converting it to lower case if necessary), and apply the tests of [Section 5.4](#) and the conversion of [Section 5.5](#) to that form. If the label is converted to

Klensin

Expires July 10, 2010

[Page 11]

Unicode (i.e., to U-label form) using the Punycode decoding algorithm, then the processing specified in those two sections **MUST** be performed, and the label **MUST** be rejected if the resulting label is not identical to the original. See Section 8.1 of [\[IDNA2008-Rationale\]](#) for additional discussion on this topic.

Conversion from the A-label and testing that the result is a U-label **SHOULD** be performed if the domain name will later be presented to the user in native character form (this requires that the lookup application be IDNA-aware). If those steps are not performed, the lookup process **SHOULD** at least test to determine that the string is actually an A-label, examining it for the invalid formats specified in the Punycode decoding specification. Applications that are not IDNA-aware will obviously omit that testing; others **MAY** treat the string as opaque to avoid the additional processing at the expense of providing less protection and information to users.

5.4. Validation and Character List Testing

As with the registration procedure described in [Section 4](#), the Unicode string is checked to verify that all characters that appear in it are valid as input to IDNA lookup processing. As discussed above and in [\[IDNA2008-Rationale\]](#), the lookup check is more liberal than the registration one. Labels that have not been fully evaluated for conformance to the applicable rules are referred to as "putative" labels as discussed in Section 2.3.2.1 of [\[IDNA2008-Defs\]](#). Putative labels with any of the following characteristics **MUST** be rejected prior to DNS lookup:

- o Labels that are not in NFC [\[Unicode-UAX15\]](#).
- o Labels containing "--" (two consecutive hyphens) in the third and fourth character positions.
- o Labels whose first character is a combining mark (see The Unicode Standard, [Section 2.11](#) [\[Unicode\]](#)).
- o Labels containing prohibited code points, i.e., those that are assigned to the "DISALLOWED" category of [\[IDNA2008-Tables\]](#).
- o Labels containing code points that are identified in [\[IDNA2008-Tables\]](#) as "CONTEXTJ", i.e., requiring exceptional contextual rule processing on lookup, but that do not conform to those rules. Note that this implies that a rule must be defined, not null: a character that requires a contextual rule but for which the rule is null is treated in this step as having failed to conform to the rule.

Klensin

Expires July 10, 2010

[Page 12]

- o Labels containing code points that are identified in [[IDNA2008-Tables](#)] as "CONTEXT0", but for which no such rule appears in the table of rules. Applications resolving DNS names or carrying out equivalent operations are not required to test contextual rules for "CONTEXT0" characters, only to verify that a rule is defined (although they MAY make such tests to provide better protection or give better information to the user).
- o Labels containing code points that are unassigned in the version of Unicode being used by the application, i.e., in the UNASSIGNED category of [[IDNA2008-Tables](#)].

This requirement means that the application must use a list of unassigned characters that is matched to the version of Unicode that is being used for the other requirements in this section. It is not required that the application know which version of Unicode is being used; that information might be part of the operating environment in which the application is running.

In addition, the application SHOULD apply the following test.

- o Verification that the string is compliant with the requirements for right to left characters, specified in [[IDNA2008-BIDI](#)].

This test may be omitted in special circumstances, such as when the lookup application knows that the conditions are enforced elsewhere, because an attempt to look up and resolve such strings will almost certainly lead to a DNS lookup failure except when wild cards are present in the zone. However, applying the test is likely to give much better information about the reason for a lookup failure -- information that may be usefully passed to the user when that is feasible -- than DNS resolution failure information alone.

For all other strings, the lookup application MUST rely on the presence or absence of labels in the DNS to determine the validity of those labels and the validity of the characters they contain. If they are registered, they are presumed to be valid; if they are not, their possible validity is not relevant. While a lookup application may reasonably issue warnings about strings it believes may be problematic, applications that decline to process a string that conforms to the rules above (i.e., does not look it up in the DNS) are not in conformance with this protocol.

5.5. Punycode Conversion

The string that has now been validated for lookup is converted to ACE form by applying the Punycode algorithm to the string and then adding the ACE prefix.

5.6. DNS Name Resolution

The A-label resulting from the conversion in [Section 5.5](#) or supplied directly (see [Section 5.3](#)) is combined with other labels as needed to form a fully-qualified domain name which is then looked up in the DNS, using normal DNS resolver procedures. The lookup can obviously either succeed (returning information) or fail.

6. Security Considerations

Security Considerations for this version of IDNA are described in [[IDNA2008-Defs](#)], except for the special issues associated with right to left scripts and characters. The latter are discussed in [[IDNA2008-BIDI](#)].

In order to avoid intentional or accidental attacks from labels that might be confused with others, special problems in rendering, and so on, the IDNA model requires that registries exercise care and thoughtfulness about what labels they choose to permit. That issue is discussed in [Section 4.3](#) of this document which, in turn, points to a somewhat more extensive discussion in [[IDNA2008-Rationale](#)].

7. IANA Considerations

IANA actions for this version of IDNA are specified in [[IDNA2008-Tables](#)] and discussed informally in [[IDNA2008-Rationale](#)]. The components of IDNA described in this document do not require any IANA actions.

8. Contributors

While the listed editor held the pen, the original versions of this document represent the joint work and conclusions of an ad hoc design team consisting of the editor and, in alphabetic order, Harald Alvestrand, Tina Dam, Patrik Faltstrom, and Cary Karp. This document draws significantly on the original version of IDNA [[RFC3490](#)] both conceptually and for specific text. This second-generation version would not have been possible without the work that went into that first version and especially the contributions of its authors Patrik Faltstrom, Paul Hoffman, and Adam Costello. While Faltstrom was actively involved in the creation of this version, Hoffman and Costello were not and should not be held responsible for any errors or omissions.

9. Acknowledgments

This revision to IDNA would have been impossible without the accumulated experience since [RFC 3490](#) was published and resulting comments and complaints of many people in the IETF, ICANN, and other communities, too many people to list here. Nor would it have been possible without [RFC 3490](#) itself and the efforts of the Working Group that defined it. Those people whose contributions are acknowledged in [RFC 3490](#), [[RFC4690](#)], and [[IDNA2008-Rationale](#)] were particularly important.

Specific textual changes were incorporated into this document after suggestions from the other contributors, Stephane Bortzmeyer, Vint Cerf, Lisa Dusseault, Paul Hoffman, Kent Karlsson, James Mitchell, Erik van der Poel, Marcos Sanz, Andrew Sullivan, Wil Tan, Ken Whistler, Chris Wright, and other WG participants and reviewers including Martin Duerst, James Mitchell, Subramanian Moonesamy, Peter Saint-Andre, Margaret Wasserman, and Dan Winship who caught specific errors and recommended corrections. Special thanks are due to Paul Hoffman for permission to extract material from his Internet-Draft to form the basis for [Appendix A](#).

10. References

10.1. Normative References

[IDNA2008-BIDI]

Alvestrand, H. and C. Karp, "An updated IDNA criterion for right-to-left scripts", August 2009, <<https://datatracker.ietf.org/drafts/draft-ietf-idnabis-bidi/>>.

[IDNA2008-Defs]

Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", August 2009, <<https://datatracker.ietf.org/drafts/draft-ietf-idnabis-defs/>>.

[IDNA2008-Tables]

Faltstrom, P., "The Unicode Codepoints and IDNA", August 2009, <<https://datatracker.ietf.org/drafts/draft-ietf-idnabis-tables/>>.

A version of this document is available in HTML format at <http://stupid.domain.name/idnabis/draft-ietf-idnabis-tables-06.html>

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities",

STD 13, [RFC 1034](#), November 1987.

- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, [RFC 1035](#), November 1987.
- [RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, [RFC 1123](#), October 1989.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", [RFC 3492](#), March 2003.

[Unicode-UAX15]

The Unicode Consortium, "Unicode Standard Annex #15: Unicode Normalization Forms", 2006,
<<http://www.unicode.org/reports/tr15/>>.

10.2. Informative References

- [ASCII] American National Standards Institute (formerly United States of America Standards Institute), "USA Code for Information Interchange", ANSI X3.4-1968, 1968.

ANSI X3.4-1968 has been replaced by newer versions with slight modifications, but the 1968 version remains definitive for the Internet.
- [IDNA2008-Mapping] Resnick, P. and P. Hoffman, "Mapping Characters in IDNA", September 2009, <<https://datatracker.ietf.org/drafts/draft-ietf-idnabis-mapping/>>.
- [IDNA2008-Rationale] Klensin, J., Ed., "Internationalized Domain Names for Applications (IDNA): Issues, Explanation, and Rationale", February 2009, <<https://datatracker.ietf.org/drafts/draft-ietf-idnabis-rationale>>.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", [RFC 2181](#), July 1997.
- [RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", [RFC 2671](#), August 1999.
- [RFC3490] Faltstrom, P., Hoffman, P., and A. Costello,

"Internationalizing Domain Names in Applications (IDNA)",
[RFC 3490](#), March 2003.

- [RFC3491] Hoffman, P. and M. Blanchet, "Nameprep: A Stringprep Profile for Internationalized Domain Names (IDN)",
[RFC 3491](#), March 2003.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66,
[RFC 3986](#), January 2005.
- [RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", [RFC 3987](#), January 2005.
- [RFC4690] Klensin, J., Faltstrom, P., Karp, C., and IAB, "Review and Recommendations for Internationalized Domain Names (IDNs)", [RFC 4690](#), September 2006.
- [RFC4952] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", [RFC 4952](#), July 2007.
- [Unicode] The Unicode Consortium, "The Unicode Standard, Version 5.0", 2007.

Boston, MA, USA: Addison-Wesley. ISBN 0-321-48091-0.
This printed reference has now been updated online to reflect additional codepoints. For codepoints, the reference at the time this document was published is to Unicode 5.2.

Appendix A. Summary of Major Changes from IDNA2003

1. Update base character set from Unicode 3.2 to Unicode version-agnostic.
2. Separate the definitions for the "registration" and "lookup" activities.
3. Disallow symbol and punctuation characters except where special exceptions are necessary.
4. Remove the mapping and normalization steps from the protocol and have them instead done by the applications themselves, possibly in a local fashion, before invoking the protocol.
5. Change the way that the protocol specifies which characters are allowed in labels from "humans decide what the table of

codepoints contains" to "decision about codepoints are based on Unicode properties plus a small exclusion list created by humans".

6. Introduce the new concept of characters that can be used only in specific contexts.
7. Allow typical words and names in languages such as Dhivehi and Yiddish to be expressed.
8. Make bidirectional domain names (delimited strings of labels, not just labels standing on their own) display in a less surprising fashion whether they appear in obvious domain name contexts or as part of running text in paragraphs.
9. Remove the dot separator from the mandatory part of the protocol.
10. Make some currently-valid labels that are not actually IDNA labels invalid.

[Appendix B](#). Change Log

[[RFC Editor: Please remove this appendix.]]

[B.1](#). Changes between Version -00 and -01 of [draft-ietf-idnabis-protocol](#)

- o Corrected discussion of SRV records.
- o Several small corrections for clarity.
- o Inserted more "open issue" placeholders.

[B.2](#). Version -02

- o Rewrote the "conversion to Unicode" text in [Section 5.2](#) as requested on-list.
- o Added a comment (and reference) about EDNS0 to the "DNS Server Conventions" section, which was also retitled.
- o Made several editorial corrections and improvements in response to various comments.
- o Added several new discussion placeholder anchors and updated some older ones.

[B.3.](#) Version -03

- o Trimmed change log, removing information about pre-WG drafts.
- o Incorporated a number of changes suggested by Marcos Sanz in his note of 2008.07.17 and added several more placeholder anchors.
- o Several minor editorial corrections and improvements.
- o "Editor" designation temporarily removed because the automatic posting machinery does not accept it.

[B.4.](#) Version -04

- o Removed Contextual Rule appendices for transfer to Tables.
- o Several changes, including removal of discussion anchors, based on discussions at IETF 72 (Dublin)
- o Rewrote the preprocessing material (former [Section 5.3](#)) somewhat -- see [Appendix B.14](#).

[B.5.](#) Version -05

- o Updated part of the A-label input explanation ([Section 5.3](#)) per note from Erik van der Poel.

[B.6.](#) Version -06

- o Corrected a few typographical errors.
- o Incorporated the material (formerly in Rationale) on the relationship between IDNA2003 and IDNA2008 as an appendix and pointed to the new definitions document.
- o Text modified in several places to recognize the dangers of interaction between DNS wild cards and IDNs.
- o Text added to be explicit about the handling of edge and failure cases in Punycode encoding and decoding.
- o Revised for consistency with the new Definitions document and to make the text read more smoothly.

B.7. Version -07

- o Multiple small textual and editorial changes and clarifications.
- o Requirement for normalization clarified to apply to all cases and conditions for preprocessing further clarified.
- o Substantive change to [Section 4.2.1](#), turning a SHOULD to a MUST (see note from Mark Davis, 19 November, 2008 18:14 -0800).

B.8. Version -08

- o Added some references and altered text to improve clarity.
- o Changed the description of CONTEXTJ/CONTEXT0 to conform to that in Tables. In other words, these are now treated as distinction categories (again), rather than as specially-flagged subsets of PROTOCOL VALID.
- o The discussion of label comparisons has been rewritten to make it more precise and to clarify that one does not need to verify that a string is a [valid] A-label or U-label in order to test it for equality with another string. The WG should verify that the current text is what is desired.
- o Other changes to reflect post-IETF discussions or editorial improvements.

B.9. Version -09

- o Removed Security Considerations material to Defs document.
- o Removed the Name Server Considerations material to Rationale. That material is not normative and not needed to implement the protocol itself.
- o Adjusted terminology to match new version of Defs.
- o Removed all discussion of local mapping and option for it from registration protocol. Such mapping is now completely prohibited on Registration.
- o Removed some old placeholders and inquiries because no comments have been received.
- o Small editorial corrections.

B.10. Version -10

- o Rewrote the registration input material slightly to further clarify the "no mapping on registration" principle.
- o Added placeholder notes about several tasks, notably reorganizing [Section 4](#) and [Section 5](#) so that subsection numbers are parallel.
- o Cleaned up an incorrect use of the terms "A-label" and "U-label" in the lookup phase that was spotted by Mark Davis. Inserted a note there about alternate ways to deal with the resulting terminology problem.
- o Added a temporarily appendix (above) to document alternate strategies for possible replacements for the former [Section 5.3](#) (see [Appendix B.14](#)).

B.11. Version -11

- o Removed dangling reference to "C-label" (editing error in prior draft).
- o Recast the last steps of the Lookup description to eliminate "apparent" (previously "putative") terminology.
- o Rewrote major portions of the temporary appendix that describes transitional mappings to improve clarity and add context.
- o Did some fine-tuning of terminology, notably in [Section 3.2.1](#).

B.12. Version -12

- o Extensive editorial improvements, mostly due to suggestions from Lisa Dusseault.
- o Conformance statements have been made consistent, especially in [Section 4.2.1](#) and subsequent text, which said "SHOULD" in one place and then said "MAY" as the result of incomplete removal of registration-time mapping. Also clarified the definition of "registration processes" in [Section 4.1](#) -- the previous text had confused several people.
- o A few new "question to the WG" notes have been added about appropriateness or placement of text. If there are no comments on the mailing list, the editor will apply his own judgment.
- o Several of the usual small typos and other editorial errors have been corrected.

- o [Section 5](#) has still not been reorganized to match [Section 4](#) in structure and subsection numbering -- will be done as soon as the mapping decisions and references are final.

[B.13.](#) Version -13

- o Modified the "putative label" text to better explain the term and explicitly point back to Defs.
- o Slight rewrite of former [section 5.3](#) to clarify the NFC requirement and to start the transition toward having some of the explanation in the Mapping document. That whole section has been removed in -14, see [Appendix B.14](#) for more information.

[B.14.](#) Version -14

- o Fixed substantive typographical error caught by Wil Tan.
- o Added a check for consecutive hyphens in positions 3 and 4 to Lookup.
- o Reflected several changes suggested by Andrew Sullivan.
- o Rearranged and rewrote material to reflect the mapping document and its status.
- o The former [Section 5.3](#) and [Appendix A](#), which discussed mapping alternatives, have been dropped entirely. Such discussion now belongs in the Mapping document, the portion of Rationale that supports it, or not at all. [Section 5.2](#) has been rewritten slightly to point to Mapping for those issues.
- o Note: With the revised mapping material inserted, I've just about given up on the idea of having the subsections of Sections [4](#) and [5](#) exactly parallel each other. Anyone who still feels strongly about this should be prepared to make very specific suggestions.
--JcK

[B.15.](#) Version -15

- o Corrected name of protocol in the abstract ("Internationalization" to "Internationalized") and a few other instances of that error.
- o Corrected the hyphen test ([Section 4.2.3.1](#)).
- o Added text to deal with the "upper case in A-labels" problem.

- o Adjusted Acknowledgments to remove Mark Davis's name, per his request and advice from IETF Trust Counsel.
- o Incorporated other changes from WG Last Call.
- o Small typographical and editorial corrections.

B.16. Version -16

- o Adjusted references to current versions.
- o Adjusted discussion of changes to Punycode to make more precise.
- o Inserted text to clarify version matching between IDNA and Unicode.
- o Made several small changes based on Martin Duerst's review.
- o Substituted in Section numbers in references to other IDNA2008 documents.

B.17. Version -17

This is the version of the document produced to reflect comments on IETF Last Call. For the convenience of those who made comments and of the IESG in evaluating them, this section therefore identifies non-editorial changes made in response to Last Call comments in somewhat more detail than may be usual.

- o Eliminated the use of "Fake A-label" in this document because it was causing confusion. Instead, the material in [Section 4.2.1](#) that used that terminology has been recast to be specific about the restriction. (Margaret Wasserman, Ops Directorate review.)
- o Additional paragraph added to Security Considerations to call out the Registry Restrictions/ permitted name policy issue. (Margaret Wasserman, Ops Directorate review.)
- o The statement "IDNA applies only to DNS labels" changed to "IDNA applies only to a specific subset of DNS labels" because it doesn't apply to all of them. (Dan Winship review, 20091013)
- o Clarified some "label" versus "domain name" terminology. (Dan Winship review, 20091013)
- o Corrected an error in reference to [RFC 1034](#) to point to [RFC 1035](#) instead. (Dan Winship review, 20091013)

- o Corrected title and a reference in 4.2.4. (Dan Winship review, 20091013)
- o Restructured the last paragraph of [Section 4.4](#) to finish reflecting the change removing the 63 octet limit on U-labels. (Dan Winship review, 20091013)
- o Another patch to the case-sensitivity of A-labels. (James Mitchell, 20091014)
- o Added text to 3.2 to explicitly indicate that IDNA-aware applications may choose to accept A-labels, U-labels, or both. (Peter Saint-Andre, 20091019)

[B.18.](#) Version -18

Changes made in response to IESG post-Last-Call review.

- o Several typographical and reference fixes.
- o Added new paragraph to the end of [Section 4.2.1](#) to clarify the relationship among the "no U-label" option and the rest of [Section 4](#).

Author's Address

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

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

