IDN - what's up?

Patrik Fältström
paf@cisco.com

In the beginning

3454 Preparation of Internationalized Strings ("stringprep"). P.
     Hoffman, M. Blanchet. December 2002. (Format: TXT=138684 bytes)
     (Status: PROPOSED STANDARD)

3490 Internationalizing Domain Names in Applications (IDNA). P.
     Faltstrom, P. Hoffman, A. Costello. March 2003. (Format: TXT=51943
     bytes) (Status: PROPOSED STANDARD)

3491 Nameprep: A Stringprep Profile for Internationalized Domain Names
     (IDN). P. Hoffman, M. Blanchet. March 2003. (Format: TXT=10316 bytes)
     (Status: PROPOSED STANDARD)

3492 Punycode: A Bootstring encoding of Unicode for Internationalized
     Domain Names in Applications (IDNA). A. Costello. March 2003.
     (Format: TXT=67439 bytes) (Status: PROPOSED STANDARD)

What is this?

- 3454 Specifies overall algorithm - stringprep
- 3490 Specifies IDN algorithm - IDNA
- 3491 Specifies Nameprep
- 3492 Specifies Punycode

stringprep

- With profiles, any Unicode based string can be converted to another Unicode string so that they can be compared

  - ✓ Include illegal codepoints

  - ✓ Include mapping table

  - ✓ Give ability to create profiles

- Used for IDN, LDAP and other protocols

idna

- Algorithm for how to convert a domain name with Unicode codepoints to ascii

  ✓ How to use the stringprep profile and unicode

  ✓ Includes specification on how to handle unallocated codepoints

  ✓ "core" to IDN standard

nameprep

- Specific stringprep profile for unicode based domain names

- Convert a domain name with unicode codepoints to one of

  - ✓ Illegal domain name

  - ✓ Domain name with Unicode codepoints

punycode

- Converts a label with unicode codepoints to a domain name in ascii

- Example:

  ✓ fältström

  ✓ xn--fltstrm-5wa1o

What happened?

4690 Review and Recommendations for Internationalized Domain Names

  (IDNs). J. Klensin, P. Faltstrom, C. Karp, IAB. September 2006.

  (Format: TXT=100929 bytes) (Status: INFORMATIONAL)

In short...

- Explains the problems in the earlier standards
  - ✓ Bidirectional scripts
  - ✓ Non-spacing codepoints
- Explains the problems with scripts not yet created when IDNA was written
- Explains problem with versioning of Unicode
  - ✓ Old standard based on Unicode 3.2

Example

- If a label include a character that has right to left directionality, both first and last character of the string has to have right to left directionality

- Creates problem if for example the string ends with a codepoint with no directionality

יִיְווָא

U+05D9 HEBREW LETTER YOD (R)

U+05D9 HEBREW LETTER YOD (R)

U+05B4 HEBREW POINT HIRIQ (NSM)

U+05D5 HEBREW LETTER VAV (R)

U+05D5 HEBREW LETTER VAV (R)

U+05D0 HEBREW LETTER ALEF (R)

U+05B8 HEBREW POINT QAMATS (NSM)

- Note that last codepoint has no directionality (Non Spacing Mark)

יְיְוָא

U+05D9 HEBREW LETTER YOD (R)

U+05D9 HEBREW LETTER YOD (R)

U+05B4 HEBREW POINT HIRIQ (NSM)

U+05D5 HEBREW LETTER VAV (R)

U+05D5 HEBREW LETTER VAV (R)

U+05D0 HEBREW LETTER ALEF (R)

U+05B8 HEBREW POINT QAMATS (NSM)

- Note that last codepoint has no directionality (Non Spacing Mark)

New IDN standard

- Will consist of a few documents

- Will not change punycode

- Backward compatible

  ✓ Some incompatibilities exists

New documents

- draft-ietf-idnabis-rationale

- draft-ietf-idnabis-protocol

- draft-ietf-idna-bidi

- draft-ietf-idnabis-tables

- draft-ietf-idnabis-mappings

draft-ietf-idnabis-rationale

- In fact named "Rationale and issues..."

- Addresses the concerns in the IAB document RFC 4690

- Explain how the issues are resolved

draft-ietf-idnabis-protocol

- Replaces the IDNA specification
- Core specification of new IDN standard

draft-ietf-idna-bidi

- Gives specifics for bidirectional scripts

draft-ietf-idnabis-tables

- Defines algorithm to use to calculate whether a codepoint in Unicode is in one of the categories
  - ✓ PVALID (Protocol Valid)
  - ✓ CONTEXTO / CONTEXTJ
  - ✓ DISALLOWED
  - ✓ UNASSIGNED

But IDNA2003 had mappings

- Mappings are not part of IDNA200x

- Labels MUST be stable under NFC

- Codepoints in label MUST pass bidi requirements

- Codepoints MUST be ok according to algorithm specified in tables document (which might include contextual rules)

- We MIGHT see a separate document on mapping, recommended behaviour for different applications etc

Why is this needed?

- IDNA standard must be independent of Unicode version

- IDNA standard must handle bidirectional scripts

- ...plus other things mentioned in RFC 4690

When will it be ready?

- "Within 6 months"
  - ✓ Actually, it is approved by IESG
- Mailing list: idna-update@alvestrand.no

Patrik Fältström
paf@cisco.com