Internationalization of Domain Names

Marc Blanchet (Marc.Blanchet@viagenie.qc.ca)

Co-chair of the IETF idn working group

Viagénie (http://www.viagenie.qc.ca)

Do You Like Quoted Printable?

- If yes, then I don’t like you… ;-))
- If no, then, bad news, we are doing worst!
- Welcome to the Internationalization of Domain Names!

Plan

- What are we talking about?
- Some background
- Bits of history
- Architecture of the solution
- Current status
- Impacts on users, applications, DNS, registration process, registries
- References

What are we talking about?

- Domain names are currently encoded in ASCII (restricted set).
- Fits for English names.
- Goal: have domain names in any language.
  - www.bébé.ca
  - Any kind of records
- Some « obvious » requirements:
  - Uniqueness of domain names
  - Universal
  - No harm to the current DNS
  - Preserve current use (case insensitive)
  - Use _one_ charset
  - Domain names are typed by humans…

Some « Obvious » Characteristics of DNS

- Only a lookup system
- Has one root
- Caching system
- 63 octets max for each label. Yes, people use it:
  - BRELLA-INTERACTIVE-MULTIMEDIA-INTERNET-VIDEO-PRODUCTION-CHICAGO.COM
- Can transport binary data.

Bits of History

- Before June1999:
  - Many prototypes. Nothing concluding.
  - The Universal Character Set (Unicode/ISO 10646) is starting to be implemented in the market (OS, Applications, …)
- June 1999: BOF at INET99 regarding a prototype implemented that encodes the domain name in an ASCII representation.
- November 1999: IETF BOF. James Seng and Marc Blanchet, co-chairs.
- February 2000: idn working group officially approved by IESG.
- December 2000: consensus on main architecture points.
- Spring 2001: received notification of patent, similar to the architecture.
- August 2001: Reconfirming architecture and working documents. Few issues added.
- December 2001: Additional options for some language support were withdrawn. Going to WG last call in January.

« Industry history »

- DNS clients (and resolvers) usually do not enforce ASCII-only domain names (ex: usual browsers, email clients)
- DNS transport is binary
- Industry:
  - At the same time that IETF idn wg was working on the spec,
  - Various registries started to « sell » i18n domain names, based on the two previous facts. But technically, it does not work.
    - Multiple character sets: same octet pattern have multiple interpretation depending on which character set. No « character set signalling in DNS.
    - Completely proprietary, non interoperable, mostly related to having an alternate-private-registry owned DNS root.
  - An OS manufacturer wanted to support i18n in their naming infrastructure and had moved to DNS.
    - But the initial implementation did not do some necessary pre-processing steps to remove ambiguities in the character set used (called normalization).
- Official messages sent by IETF, ICANN and ISOC to « stop » this.

IETF work

- Initial proposal (based on 1999 prototype)
  - Encode idn in ASCII. No modification to DNS
- Went to a list of alternative proposals
  - In DNS:
    - Modify the DNS protocol to signal that this is an idn request
    - Keep the encoding in binary (UTF-8)
  - In applications
    - Encode idn in ASCII
  - Over DNS:
    - Have an upper layer dealing with languages
    - “similar” but extended keywords
    - Could take care of language subtleties

Consensus

- Encode idn in ASCII (called ACE: ASCII Compatible Encoding)
  - Advantages:
    - No need to change the DNS protocol
    - No need to update the DNS infrastructure
    - Does not break current applications behavior
  - Disadvantages:
    - Current applications that knows nothing about idn will display weird names on the screen.
    - Encoding means expansion. An English label could be up to 63 chars. A non-English label will be less than 63 chars (for some cases, max of ~25 chars)

Proposal

- Use the Universal Character Set:
  - ISO 10646/Unicode: 16/32 bits binary set.
- Nameprep:
  - Prohibit some characters in UCS
  - Case Folding:
    - É => é
  - Normalization:
    - e <backspace>^ = ê
- Convert ISO 10646/Unicode codepoints in ASCII
  - Bébé = ad7xn73j75q76z7j75xp62p7mu
  - Make sure the expansion is less or equal to 63 octets
- Add a prefix to signal an idn to the applications
  - bébé.ca = zq--ad7xn73j75q76z7j75xp62p7mu.ca

Process

Keyboard entry
Convert to ISO10646
Nameprep
Convert to ACE
DNS Request

Bébé.ca (encoded in ISO-8859-1: 0x62c962e9)

Bébé.ca (encoded in ISO-10646: 0x006200c9006200e9)

bébé.ca (normalized/case fold: 0x006200e9006200e9)

zq--ad7xn73j75q76z7j75xp62p7mu.ca

This process is done by the application. No change to resolver or DNS. However, operating systems could help applications by offering system libraries.

Language Considerations

- These days, domain names are used for many things. They mean something for the user (identity, trademark, …).
- Domain names must be unique.
- Domain names must be resolved the same way everywhere, in every country, in whatever language.
- But:
  - Human languages are not designed to be so constrained.
  - Languages may use the same word:
    - with different meanings.
    - Context based
  - Languages are evolving, not only in words but also in character representations.

– DNS is very very limited (read: can’t do anything) to manage language subtleties. Like it or not. Need an upper layer for that processing.

Current Status

- IETF process:
  - Jan 2002: going WG last call on the 3 documents.
  - If success, will be forwarded to IESG for RFC standard track publication
  - Goes IETF last call.
  - If success, then RFC-editor.
  - RFC-editor will handle the prefix assignment at the time of publication.
  - Impact on other protocols
  - Other i18n of identifiers: email left part is an example.
- Industry:
  - Registration process to be modified
  - Application/OS modifications

Applications not supporting IDN

- Will continue to work.
- No processing of IDN:
  - ACE display (leakage): zq--ad7xn73j75q76z7j75xp62p7mu.ca

- User must type the ACE version of an idn
  - Must type: bq--ad7xn73j75q76z7j75xp62p7mu.ca
  - But:
    - Copy/paste should work, but some cases are « interesting »

DNS Server Management

- The zone data should be “prepared” and will have the ACE:
  - www.zq--ad7xn73j75q76z7j75xp62p7mu.ca IN A 206.123.31.2
  - Is this really www.bébé.ca?
- Tools for zone file administration
- However, the current server software works as is:
  - Since no change in the protocol
  - No change in the zone (records, …)

Domain Name Registration

- Registrars:
  - Many changes to their applications
  - Multilingual/multi-character set Interface
  - Charset conversion-> Nameprep -> ACE
  - Maximum label length to be verified
  - User support
- Registry:
  - No « big » change
  - Verify the idn in the registration request
  - Tools to help zone file administration

User Considerations

- Long deployment period
  - Long leakage period.
- Security issues?
  - User does not have installed the font to display the idn. Visual verification?
  - A non knowledgeable person in a foreign language might have problems seeing the difference between two chars. For example, for me, if I do not look it carefully, this is identical:
    - 憶 = 臆 ?
  - Domain names in certificates
- Data entry
- « Business card » issue: if I put my idn on my business card, do all recepients of this card will be able to type it? If not, then do I also put the ACE version? Yearkkk…

Some Upper Layer Considerations

- Are dispute resolution processes updated/ok for idn?
- I18n TLDs?
  - 京東.jp ou 京東.日本 ?

Conclusion

- IETF is standardizing idn.
- Based on ASCII compatible encoding. No change in DNS protocol and infrastructure.
- Changes in applications (and registration process)

References

draft-ietf-idn-idna-xx.txt
- Overall architecture

draft-ietf-idn-nameprep-xx.txt
- Name preparation step

draft-ietf-idn-punycode-xx.txt
- ASCII compatible encoding

Unicode UTR15
- Normalization rules

ISO 10646, Unicode 3.1
- Character Set

Mailing list: idn@ops.ietf.org

Web site: http://www.i-d-n.net

Questions?

Marc.Blanchet@viagenie.qc.ca