

The Challenge of Realizing an Internationalized Internet

Linuxforum 2007
Copenhagen, Denmark

Tina Dam
Director of IDN Program
ICANN

Email: tina.dam@icann.org



Session Overview

- What is Internationalization
- IDN general information and definitions
- Historic overview
- Protocol functionality
- Implementation difficulties and issues
- How far are we today and what is missing
- ICANN's IDN Program Plan
- Q/A

Internationalizing or localizing the Internet

- Internationalization of the internet means that the internet is equally accessible from all languages and scripts
- Domain names represent only a small part of internationalization of the internet
- Controversy about how important the domain names are compared to search capabilities...etc...
 - Accessibility from all languages is important which means that the way IDNs are handled is very important
 - Continuously making characters available as much as possible as these are added to Unicode
 - Disagreement about whether domain names are used by typing into browsers and usability of IDNs
 - But agreement that email addresses based on local characters are necessary for large parts of the world,
 - and URL's listed in offline documents need to be usable by local communities

What is an IDN?

- IDN stands for Internationalized Domain Name
 - Domain name labels containing non-host name characters.
 - Valid hostname characters are: a-z, 0-9, “-”
 - Valid hostname characters sometimes referred to as ASCII or LDH
 - Only host name strings are entered into the DNS
 - IDN in general refers to both displayed form (Unicode) and stored form (ACE or punycode) of the domain name
- Example: rødgrød.tld → xn--rdgrd-vuad.tld
 - ø is LATIN SMALL LETTER o WITH STROKE: U+00F8
 - Used in for example Danish, Norwegian, Faroese

Domain Names in General

- Domain names are not general natural language expressions
- Domain names that are not lexically words in a language are possible and quite common
- Domain names are identifiers that help users uniquely reference information in the Internet using sequence of characters into strings
- Domain names must be unique
- Not all words in all languages will be available as domain name labels

Internationalization/Historic Overview

Domain Names Based on
ASCII / LDH Rule

- IDN second level
- Internationalized top level

ASCII based browser/email
clients/...

- Application upgrades to get
web access in local chars +
IDN enabled emails...

Content have been available
in many languages for
some time

- Expected to continue to
expand

example.test → 실례.test and 실례.테스트

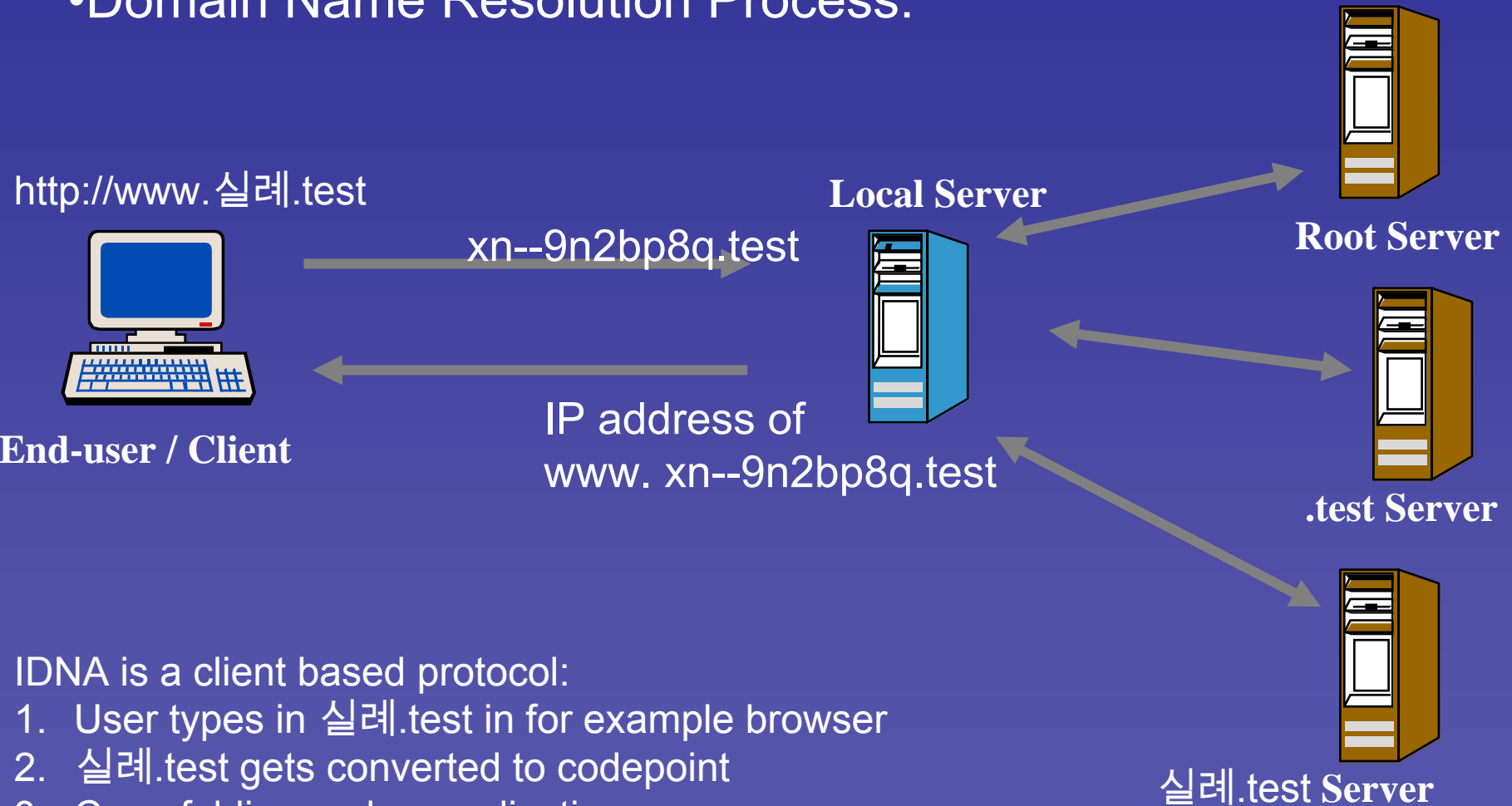
(stored form: example.test → xn--9n2bp8q.test and xn--9n2bp8q.xn--9t4b11yi5a)

Aim: An internationalized Internet



IDNA – Protocol Functionality

•Domain Name Resolution Process:



- IDNA is a client based protocol:
1. User types in 실례.test in for example browser
 2. 실례.test gets converted to codepoint
 3. Case-folding and normalization
 4. Stringprep filter
 5. Punycode conversion → xn--9n2bp8q.test

More Protocol Information

- IDNA is the acronym for the IDN protocol, developed within the IETF, published June 2003
- IDNA stands for
 - Internationalized Domain Names in Application.
- Technical details are available in the IETF RFCs:
 - RFCs 3490, 3491, and 3492
- IDNA is currently under revision
 - RFC4690 and associated internet drafts suggesting revisions and solutions to some problems
 - More about this later...

Displayed Form vs. Stored Form

- Historically the domain name you register is also the domain names stored and usable in the DNS
 - This is changed with introduction of IDNs
- Usually the stored form does not make any meaning
 - Example: `فرسانهر.tld` → `xn--mgbtbg2evaoi.tld`
- However, there are exceptions:
 - `xn--gibberish` - decodes into the Arabic characters `ب٨٧٩أ`
 - `xn--trademark` - with different versions of trademarks
 - This is coincidentally and hence not intentionally
- `xn--` prefix specifically designates a system called Punycode
- `xn--` prefix indicates to application software that the label needs to be decoded back into Unicode for proper display to the user



More Punycode and Some User Perspective

- Intention that Punycode (xn--....) never be exposed to users, but there are exceptions
 - situations where IDNs could not be displayed as Unicode characters
 - in such cases the utility of IDN depends on user recognition and understanding of Punycode
- ...otherwise, as a user all you need is the name you want to register
 - TLD Registries will supply a list over available characters, usually in Unicode
 - Registries will handle all encodings needed during registration process
- May be useful to consider usability of the name, keyboards, business cards, and other practical limitations
- Encodings by for example:
 - <http://josefsson.org/idn.php>
 - Others are made available by TLD registries

Same Script Different Language Issues

- Language specific character issues
 - Jorgen =Jørgen = Jörgen in Danish, Swedish, Norwegian
 - But users don't always think that o equal ø and ö
 - ø is LATIN SMALL LETTER o WITH STROKE (U+00F8)
 - ö is 'LATIN SMALL LETTER o WITH DIAERESIS' (U+00D6)
- Not possible to make generic rule at the protocol level
 - But, need for specific rules at TLD registry level
- Some registries have submitted character tables to the IANA repository to show variants
 - Example: the .se table displays that:
 - The letter Ü is referred to in Swedish as a # "German Y" and is # considered to be a variant of the letter Y.
 - The letter Å is not considered to be a variant of the letter A...Earlier practice substituted AA, which is no longer recommended but will still be encountered
- <http://www.iana.org>
 - (link to IANA Repository at bottom left of main page)

Same Language Multiple Scripts Issues

- Some languages can be expressed by multiple scripts
 - Eastern European and Central Asian languages can be expressed in Cyrillic or Latin characters
 - African and Southeast Asian languages can be expressed in Arabic or Latin characters
 - Other languages are written in a combination of scripts- Kanji, Kana, Romanji for Japanese & Hangul and Hanji for Korean
- Hence, same word, same language can be expressed in different ways
 - Some words can only be expressed use a single script
 - Some words are expressed by mixing of scripts
- Result is that script definition is very important and sensitive in terms of IDNs

Visual Confusion Issues

- Well-known example: paypal.com
 - Second character is U+0430, Cyrillic small a
 - Looks like Roman/ASCII “a”
 - This is now prevented by “one label, one script” rule per the IDN Guidelines with exceptions for mixed script languages
- Other example:
 - Russian ccTLD is .ru
 - Cyrillic “r” and “u” is: p and y
 - Which looks like p y (in latin) is ccTLD for Paraguay
 - **Note: Russia did not ask for .py, this is just an example**
 - Process needed to determine labels matching ccTLDs

General Overview of User Confusion Issues

- IDNs Expanding Risk of Known Problems
- Many characters can be confused with others
 - Problem exists in ASCII as well
 - Digit “1” and lower-case “l”
 - Digit “0” and upper-case “O”
 - IDNs increasing the character collection
 - From 64 in ASCII (LDH)
 - To tens of thousands in Unicode
- This kind of confusion
 - create opportunities for user mistakes,
 - fraud, and
 - resulted in different application implementations

Mid-way Summary

We have looked at some of the main issues related to IDNS – what about solutions...

Some user confusion is being solved by

- protocol adjustments
- IDN guidelines revisions
- implementation of adequate registry policies
- standard registration and resolution implementation

Remaining user confusion need to be solved by

- education of community

In other words.....

- Examples of what we have:

brücke.de	→ xn--brcke-lva.de
실례.kr	→ xn--9n2bp8q.kr
فرسالنهر.ae	→ xn--mgbtbg2evaioi.ae
vægtskål.dk	→ xn--vgtskl-e0af.dk

- Example of where we are heading:

실례.테스트 (xn--9n2bp8q.xn--9t4b11yi5a)

- How do we get there: ICANN's IDN Program, recently established within ICANN to achieve the possibility to insert internationalized top level labels in the root zone
 - DNS root software and resolver stability testing
 - Application provider implementation
 - IDNA protocol revision (IETF and IAB via RFC4690)
 - IDN Guidelines → Best-Current-Practice
 - IDN Policy Principles

DNS Root Server Software and
Resolver Stability Testing
&
Application Provider
Implementation

IDN Laboratory Testing Details

- Autonomica will develop and ICANN will publish the test procedure
 - plan detail will be sufficient so that others may replicate the test
 - ICANN will publish the results received of any other test performed in accordance with the publish test plan
- The laboratory test plans includes the following:
 - insertion of NS records into a copy of the root zone
 - tests performed in closed laboratory environment with a series of systems implemented to replicate as closely as possible the server software of the various root servers. This includes:
 - versions of BIND server software, and
 - use of the most popular DNS resolver software packages
- No further end-user or application testing is included as the laboratory environment is closed and not accessible from outside

Phase 1: Laboratory Testing Status Quo

- Test design finalized
 - submitted to RSSAC for comments
 - Test design posted for public comments (2 weeks)
- Autonomica conducted the test as designed
 - Feasibility test to verify that the set-up and design was functional
 - Full test as designed
- One instance of testing to the end-user level completed.
 - End-user software showed difference that was not related to implementation of the IDNA protocol, and has been corrected
- Laboratory Test result
 - Test report from Autonomica is in final stages
 - Test report will be posted and announced on <http://www.icann.org>



<http://museum.flod18hastflod18hastflod18hastflod18hastflod18hastflod18/>



ICANN

Phase 2: Pre-deployment test and Application Software Communication

- The positive result from the laboratory tests will allow move to a pre-deployment IDN TLD test
- Test plans are under development
 - Technical discussions about if any additional aspects should be included in the pre-deployment test
 - Where will the test be conducted
 - Private namespace, live in the root, mix, other?
 - Participation Criteria
 - Evaluation Process
 - Success Criteria
- Test plans will need further discussion with technical community
- Extensive communication plan under development for application providers

IDNA Protocol Revision, By IETF and IAB

Proposed Revisions to IDNA Protocol

- Revising the IDNA protocol will build an “inclusion” based model for determining what scripts may be used for IDNs and potentially increase the number of scripts available for IDN deployment.
- The revision will make the protocol non-dependant on Unicode versions
 - Version 5.0 contains 64 scripts, the existing protocol is based on Unicode 3.2 containing 45 scripts
- The revision to the protocol will:
 - Potentially increase available blocks of characters (107 to 151)
 - Include an easier revision process to accommodate additional scripts in the future
 - include technical review of protocol functionality
- The revision effort is managed through the IAB/IETF

Revisions suggestions of IDNA Protocol

- Three internet-drafts were published providing suggestions for solutions to the issues raised in RFC4690:
 - An overview with proposed issues and changes for IDNA
 - <http://www.ietf.org/internet-drafts/draft-klensin-idnabis-issues-00.txt>
 - A suggestion for solving an IDNA problem in right-to-left scripts by revising the stringprep profile
 - <http://www.ietf.org/internet-drafts/draft-alvestrand-idna-bidi-00.txt>
 - An overview of suggested inclusion based IDNA Unicode Codepoints based on Unicode 5.0
 - <http://www.ietf.org/internet-drafts/draft-faltstrom-idnabis-tables-00.txt>
- new I-D published last week:

<http://www.ietf.org/internet-drafts/draft-klensin-idnabis-issues-01.txt>



IDN Policy Principles

IDN Policy WG's

- GNSO IDN WG: To identify and specify policy issues to be considered by the GNSO via a policy development process (PDP) that have not already been considered within PDP-Dec05 (New gTLDs)
- ccNSO/GAC IDN WG: to produce an issues paper relating to the selection of IDN ccTLDs associated with the ISO 3166-1 two-letter codes
- GAC IDN WG
- ccNSO IDN WG
- March 2007: Anticipated IDN joint policy sessions in Lisbon discussing the overlapping issues from each WG report

Examples of GNSO IDN Policy Issues under consideration

- Should transliterations of existing gTLD strings be addressed?
- Should next round for new gTLDs wait for the inclusion of IDN gTLDs?
- Would aliasing be a preferred, open, or discouraging option?
- Should an existing domain name holder have a priority right for a corresponding domain in another script?
- Should a particular top level script be compulsory on lower levels?
- Existing gTLD strings
 - Allocation and representation of strings
 - Backwards compatibility of existing IDNs when IDNA change?
- Geo-Political Details; countries' and ccTLD roles in IDN gTLDs?
- Privacy & Whois Details; are existing Whois policies adequate?
- Legal Details: Impact on UDRP

Examples of ccNSO-GAC joint WG IDN Policy Issues under consideration

- How is it determined that the string represents the territory?
- Does there have to be a connection with existing "ASCII TLD".
- Who is responsible for picking the string?
- Should there be a mandated process for picking the string?
- Should there be a certain status of the use of the character set in the corresponding territory? For example does the character set have to be an official language?
- Who can apply for a string (sponsoring organization, government, others)?
- Should there be a requirement that the manager of the new IDN ccTLD be connected to the entity that runs the existing TLD?
- Are there any requirements on the number of characters in the string?
- How many IDN ccTLDs can a territory have?
- Should there be specific technical requirements related to running the IDN ccTLD?
- Should there be a policy/process for handling disputes between parties such as incumbent ccTLD manager, government, other applicant?
- Should there be a policy/process for dealing with multiple applications or objections to applications?

Increasing Outreach and Communication Plans

Outreach and Communication

- Increased awareness
 - through regional liaison program
- Extensive communication plan
 - to reach outside domain name industry (applications)
- Calendar of IDN related events & ICANN blog
 - One overview available online
- Increased translation of IDN material
 - To ease local discussions and participation
- Continued IDN workshops
 - ICANN meetings and regional events

Summary of IDN Principles

- Global uniqueness and interoperability of the DNS
 - unique and unambiguous domain names
 - Same functionality regardless of geographic placement of access
 - URLs and emails connect as expected regardless of geographic placement of access
- Promote “Future-Proof” solutions
 - Define Unicode characters to be allowed
 - Provides ability for adding new languages, new characters far in the future
- Avoid or diminish as much as possible user confusion
 - Technical limitations
 - Implementation requirements
 - Registry restricted list and policies
 - User education
- Promote multi-stakeholder involvement

Thank You!

Contact details:

Email: tina.dam@icann.org

Phone: +1-310-862-2026