L2/05-229

Unicode Security Considerations (TR#36)

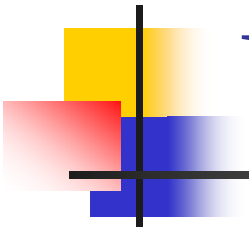
Michel Suignard
Senior Program Manager, Microsoft
Technical Director, Unicode Consortium

Unicode in short

- About 98000 characters allocated, cover all major writing systems, languages of the world
- More to come (new additions every year) as lesser known repertoires are added, tuned
- Coupled with ISO/IEC 10646 repertoire
- Specifies algorithm (such as Bidirectional) and character properties required for implementation
- Stability is a growing concern, new versions may add characters but impact existing implementation as little as possible
  - Recent case: Lower case folding
- Redundant repertoire (canonical equivalence)
  - Ö is either <U+00D6> or <U+004F,U+0308>
  - But Ø is only <U+00D8>, **not** <U+004F, U+0338>
- Canonical equivalences can be filtered using normalization
- More details on www.unicode.org

Unicode security

- UTF-8 exploit
  - Avoided by enforcing shortest form processing only.
- Multiple canonical representations
  - Use of normalization (NFC, NFKC, NFD, NFKD)
  - Already enforced in RFCs (IDNA, IRI)
- Identifier syntax (UAX#31 Identifier and Pattern Syntax)
  - Subset and guidelines for characters suitable for identifier syntax
  - Identifier-Start and Identifier-Only-Continue
  - Stability requirement (using the 'Other_' Identifiers)
  - Meant to be used as a relative reference
- Visual confusability not addressed by normalization
  - Main topic of TR#36 Unicode Security Considerations



Unicode and identifiers

- Text in general not a very good visual identifier mechanism
  - Safest: numbers (numbers work very well as attested by phone system)
  - ASCII still works ok (some issues with 0O, 1l, rnm)
  - Unicode repertoire changes the magnitude of the problem
  - Private use characters are the extreme abomination (no attached semantics)

Text confusability

- Single script confusability
  - Latin using combining sequences
  - Common in Indic scripts (e.g. आ ; आा)
  - Endemic with CJK ideographs (扮 vs 扮)
  - Happen in other scripts such as Canadian Syllabaries (ᐅ ; ᐅ)
  - User expected (inherent language ambiguity)
- Mixed scripts confusability
  - Famous paypal example
  - Very common among Latin, Greek, Cyrillic
  - Also happen among Indic scripts
  - Very user unfriendly
- Whole script confusability
  - A whole sequence can be interpreted as belonging to a different script (such as 'scope' being either Latin or Cyrillic'
- Syntax character confusability
  - Non ASCII symbols look-alike U+2215 ∕for 005C /

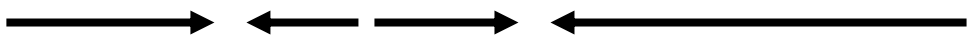
Bidirectional issues

- Bidirectional is a feature of many Middle East languages/scripts (Arabic, Hebrew)
- Logical order and visual order are different
- Require Unicode Bidi Algorithm to determine directionality of weak direction characters (separators)
- Arbitrary mixing of RtL and LtR characters creates visually undecipherable text
- Following IDN and IRI recommendations for host labels
  - Label cannot use both RtL and LtR characters
  - Label using Rtl Characters must start and end with them
  - (still may make them hard to read)
- Render bidi identifiers as if embedded left-to-right

Bidirectional IRI examples

http://معكم?١٢٣/دائم.سلام

http://سلام.abc.معكم?١٢٣/دائم



http://سلام/دائم.سلام/Path-part/١٢٣? معكم

Example of a RFC with Unicode security concerns: IDNA

- IDNA allows a very large repertoire
  - including symbols, not in-modern-use characters
- Repertoire not aligned with identifier guidelines (UAX#31)
- Current ICANN guidelines are language based, not addressing multi-lingual communities
- Case insensitive on input
- Confusable characters issues not addressed
- Stuck at Unicode 3.2 level
  - No support for N'Ko, Tifinagh, no process to update to newer version of Unicode/ISO 10646
  - Slightly deficient normalization

TR#36 recommendation

- Normalize data (NFC, NFD, NFKC, NFKD)
- Use a repertoire as small as possible
  - If you don't need symbols, don't allow them
- Restrict repertoire to UAX#31 content (start and continue-only), or at least use it as a reference point
  - Recognize that some characters cannot be first
- Use Unicode script property to avoid spurious multi-script text
- Stay away from language based policies
- When multi-script is allowed, use TR#36 tables to detect visual confusable
- Never, never allow PUA characters in identifiers

Visual confusability mitigation

- Smallest repertoire possible (LDH principle)
- Avoid multi-script text unless required by writing system (Japanese, Korean)
- Avoid case insensitivity
  - Otherwise NUVY become mixed-script confusable
- White list for questionable sequences
- Mixed script exploits can be detected by using whole-script confusable tables
  - For each script found in a given string, see if all characters in the string outside of that script have whole-script confusables for that script.
  - ‘Paypal’ is an exploit because it is made of two scripts and the Cyrillic set is whole script confusable.
  - ‘Toy-Я-us’ is not an exploit because neither set is whole script confusable.
  - Won’t protect against ‘Toy-Я-us’ because it is not mixed-script confusable.

TR36 IDN characters

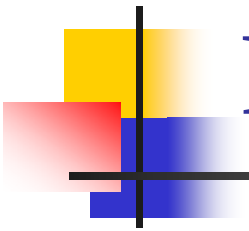
- Script policy
  - Remove punctuations and symbols
  - Remove not in modern use characters
- General purpose symbols
  - Stay as close as possible to the LDH principle
  - Incorporate those already used by TLD
  - 002D - hyphen-minus
  - 00B7 · middle dot
  - 02B9 ʹ modifier letter prime **or** 2018 ‘ left single quotation mark
  - 3003 〃 ditto mark (JP)
  - 3005 々 ideographic iteration mark (JP)
  - 3006 〆 ideographic closing mark (JP)
  - 3007 〇 ideographic number zero (JP)
  - 30FB ・katakana middle dot (JP)
- No archaic scripts
- CJK content, union of:
  - Existing ccTLD registration policy
  - CJK Unified Ideographs main block (4E00-9FA5)
  - ISO 10646 CJK IICORE collection
- http://www.unicode.org/reports/tr36/data/idnchars.txt

Example: Cyrillic script subset

- Full Unicode ranges:
  - 0400-0486, 0488-04CE, 04D0-04F5, 04F8-04F9, 0500-050F
- Exclusion:
  - 0482 ҂ Cyrillic thousand signs (symbol)
  - 0483-0486 Combining characters not in modern use
  - 0488-0489 Combining characters used for symbols
  - 04C0 Ӏ Cyrillic letter Palochka (lack of lower case letter, would be added back as soon as a lower case is encoded)

Example: Latin Script subset

- TR36 IDN ranges exclusion:
  - 0180, 018D, 01AA-01AB, 01B9-01BB, 01BE-01C3, 021D, 0250-0252, 0255, 0258, 025A, 025C-025F, 0261-0262, 0264-0267, 026A-026E, 0270-0271, 0273-0274, 0276-027F, 0281-0282, 0284-287, 0289, 028C-0291, 0293, 0295-02AD
- Archaic
  - ƀ, ɋ, ɭ, ʈ, ɛ, ʒ, ƨ, ʅ, ƿ, ȝ, ɐ , ɑ, ɒ, ɕ, ɘ, ɞ, ɜ, ɝ, ɞ, ɟ, g, ɢ, ɤ, ɥ, ɦ, ɧ, ɪ , ɬ, ɬ, ɭ, ɮ, ɯ, ɱ, ɳ, ɴ, ɷ, ɸ, ɹ, ɻ, ɻ, ɾ, ɾ, ɾ, ɰ, ʁ, ʂ, ʄ, ʅ, ʇ, ʉ, ʌ, ʍ, ʎ, ʏ, ʐ, ʑ, ʓ, ʕ, ʖ, ʗ, ʘ, ʙ, ʚ, ʛ, ʜ, ʝ, ʞ, ʟ, ʠ, ʡ, ʢ,
- Digraphs
  - œ, ʣ, ʤ, ʥ, ʦ, ʧ, ʨ, ʩ, ʪ, ʫ, ʬ, ʭ
- Symbol-like (click)
  - ǀ, ǁ, ǂ, ǃ



References

- UAX#9 (Bidirectional Algorithm)
  http://www.unicode.org/reports/tr9/tr9-15.html
- UAX#15 (Unicode Normalization Forms)
  http://www.unicode.org/reports/tr15/tr15-25.html
- UAX#24 (Script Names)
  http://www.unicode.org/reports/tr24/tr24-7.html
- UAX#31 (Identifier and Pattern Syntax)
  http://www.unicode.org/reports/tr31/tr31-5.html
- UTR#36 (Unicode Security Considerations)
  http://www.unicode.org/reports/tr36/