Guidelines on linkification for URLs with non-ASCII characters

This document is intended for developers of a software implementing linkification mechanisms (conversion of strings into hyperlinks).

The document provides best practices related to identification in a text and automated creation of hyperlinks, containing domain names and email addresses in non-ASCII scripts, in various software.

1. Recommendations on creating hyperlinks

To identify in text a string in non-ASCII script that represents a website address (domain name) or an email address and to convert such a string into a hyperlink, it is recommended to follow all steps listed below*:

1.1. Check the string for a protocol prefix (e.g. http://, https://, ftp://, mailto:).

1.2. Check the string for compliance with domain name format using the following criteria:

a. it has no less than two strings (labels) separated by dots;

b. it does not exceed the maximum length of a domain name in the ASCII format (not more than 255 characters with 63 characters per label~~ in the ASCII format~~);

c. it contains a ~~top-level~~ domain name that exists, which should be confirmed by DNS request ~~or searching~~. If for some reason DNS requests can’t be used for checking, then at least top level domain should be searched in the IANA repository ~~(http://data.iana.org/TLD/tlds-alpha-by-domain.txt);~~(http://data.iana.org/TLD/tlds-alpha-by-domain.txt)[1];

d. it ~~contains only acceptable~~shouldn’t contain disallowed characters (~~check IDN tables, if available)~~for more information see 3.1 and 3.2).

1.3. Check the string for compliance with email address format using the following criteria:

a. the correct domain part of the address is used, see 1.2;

b. the address contains @;

c. the maximum length of the local part of the address (before @) does not exceed 64 characters;

d. the local part of the address contains only acceptable characters in accordance with the best practices (special symbols such as dot (.), underscore (_), hyphen (-) or in certain cases the plus sign (+)).

*This list of steps is not exhaustive and there may be used other methods to identify strings, that are domain names or email addresses, for their further conversion into hyperlinks. The list contains the most popular and frequently used methods. The more methods are applied, the better for the efficiency of identifying such strings.

[1] Be informed that in some cases your application activity on client side can be recognized as suspicious by cybersecurity systems. For such cases you can consider making special notifications for application users.

1.4. Check the string for script mixing:

a. if script mixing is detected in any label of a domain name, then this label should be converted using Punycode encoding syntax (https://www.rfc-editor.org/rfc/rfc3492), and the label itself should not be converted into a hyperlink.

b. to improve user security defense, it is acceptable to additionally transform the string (e.g. replace “http” with “hxxp” and/or “.” with “[.]”). In this case, adding relevant comments to the program library that analyzes the text is recommended.

c. if script mixing was detected in website catalogues names, file names, variable URLs, etc., the string should be converted into a hyperlink but percent-encoded (https://www.rfc-editor.org/rfc/rfc3986#section-2.1).

d. if script mixing is allowed for use in internationalized domain names/email addresses in a certain language, then paragraph 1.4 should not be applied to such domain names and/or e-mail addresses. In these cases, the script/language specification should be followed for allowed combinations of scripts (for more information see 3.1 and 3.2).

2. General linkification guidelines

2.1 The proposed linkification mechanism should be applied for the strings when creating and/or changing them.

2.2 The user should be able to turn off the proposed linkification mechanism or cancel it for a specific string (for example, when typing text in MS Word, Backspace will remove a hyperlink).

2.3 The linkification mechanism should process hyperlinks in the same way for the entire interacting software regardless of whether the text is typed manually, copied from the clipboard or delivered via program interfaces.

3. Additional materials:

3.1 IDNA Derived Properties https://www.iana.org/assignments/idna-tables-12.0.0/idna-tables-12.0.0.xml#idna-tables-properties

3.2 IANA Repository of IDN Tables https://www.iana.org/domains/idn-tables

3.3 ICANN IDN guidelines https://www.icann.org/en/system/files/files/idn-guidelines-22sep22-en.pdf

3.4 Unicode Technical Standard 39 http://www.unicode.org/reports/tr39/

3.5 UASG 004 Test Cases for UA Readiness Evaluation https://uasg.tech/download/uasg-004-use-cases-for-ua-readiness-evaluation-en/