BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2018
http://excel.fit.vutbr.cz

Survey of EAI and IDN implementation

Irena Talašová*

Abstract
The main topic of this work is to find and summarize the current state of implementation and support of Unicode characters in the most known network protocols and applications. This is a topic that, in my opinion, deserves more attention from IT professionals and the general public, as it is likely to be the direction that application development will take. The following chapters summarize a brief introduction to this issue and will then introduce the current state of support for UTF-8 characters in selected network protocols and applications. Since, in general, the state of implementation and actual functionality of applications do not match the RFC specification of the corresponding network protocols, it is necessary to test and explore available applications and obtain relevant information. The information presented here are obtained both from expert articles and on-line accessible websites of organizations, and above all the results obtained during the testing. The information found can be used by all who want or plan to implement local language support in their developed tools. PCAP files containing accented network identifiers are provided as supplementary material that can be used for testing purposes.

Keywords: EAI — IDN — Unicode — Email address — Internet services — UA — Network protocols

Supplementary Material: Demonstrative PCAPs

*xtalas04@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Increasing user's demand for multilingual Internet has led to the implementation of UTF-8 character support for some network protocols and applications. We are increasingly confronted with terms like EAI (Email Address Internationalization) in emails or IDN (Internationalized Domain Names) in the landscape of domain names. These principles also apply to other IT areas and protocols such as SIP or FTP.

This creates a potential vulnerability for lawful interception and other software monitoring network communications, which can not process and recognize a network flow containing national characters. The aim of this work is to carry out research on the current state of implementation of IDN and EAI. It is not possible to examine all network applications, so the most

prominent representatives for whom the support will be tested are selected for each protocol being examined. The knowledge gained is usable for all network application developers who are in the process of deciding whether to implement EAI and IDN support in the software.

The second chapter provides an introduction to the topic and explains the important concepts found in other parts of the paper.The third chapter focuses on IDN TLDs. The fourth chapter deals with practical knowledge with focus on SMTP, POP3 / IMAP, SIP and FTP protocols.

2. Preliminaries and Definitions

Universal Acceptance (UA) is the state where all domain names and email addresses are accepted, validated, stored, processed and displayed correctly and consistently by all applications, devices and systems [1], so internet applications and systems must treat all TLDs in a consistent manner, including new gTLDs and internationalized TLDs. This is a key measure of the success of IDNs (UA). Universal Acceptance is a foundational requirement for a truly multilingual Internet, in which users around the world can navigate entirely in local languages [2].

It is also an opportunity to support consumer choice and innovation in the domain name industry. Many systems do not recognize or appropriately process new domain names, primarily because the top-level domain may be new, more than three characters in length or in non-ASCII format (Internationalized Domain Names, or IDNs). The same is true for email addresses that incorporate these new domain names or use Unicode in the mailbox names [3].

The Universal Acceptance Steering Group [2], supported by Internet Corporation for Assigned Names and Numbers (ICANN), is a community-led initiative working on creating awareness and identifying and resolving problems associated with Universal Acceptance. The purpose of these efforts is to help ensure a consistent and positive experience for Internet users globally. The group's primary objective is to help software developers and website owners understand how to update their systems to keep pace with an evolving domain name system (DNS).

Previous studies have shown that there are significant barriers to Universal Acceptance of IDNs [4]. Progress toward UA for IDNs is especially slow in applications and security-related software. While there have been significant announcements of support for IDNs in email and other applications, the pace of uptake remains very slow [4].

IDN is an abbreviation for International Domain Names. This is a method and standard that allows people worldwide to use domain names in their local language. Domain name characters are encoded as UTF-8 string [5].

EAI is the extension that allows email addresses with IDNs in the domain part and/or Unicode (non-ASCII) characters in the mailbox name to function within the traditional email environment. Email software and services need to make specific changes to support EAI. Client software should display the domain name and mailbox name in Unicode. Server software should confirm EAI-readiness (e.g. advertise SMTPUTF8 support) when making connection to another MTA [3]. An example of Internationalized Email Address is "háčky@čárky.cz".

Punycode is a way to represent IDNs with the limited character set (A-Z, 0-9) supported by the domain name system [6]. For example, "háčkyčárky.cz" is encoded as "xn–hkyrky-ptac70bc.cz".

3. EAI & IDN in domain names

3.1 IDNs in the .cz TLD

CZ.NIC is the registry manager of the .cz country code top-level domains. According to their statement [7], they are technically prepared for the introduction of diacritics in Czech domains. The fact that the support has not yet been implemented is due to the alleged lack of interest among Czech users and organizations. On the other hand, second-level domains do not have problems with diacritics. Every two years the organization conducts research of interest among Czech users. User's interest in IDNs is represented in Figure 1. Numbers are given in percent.

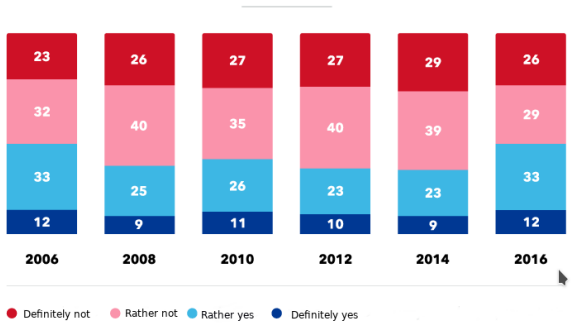

Figure 1. User's opinion on the introduction of diacritics in .cz domain [7].

The introduction of IDN is a relatively complex process which, in addition to its benefits, includes several relatively important negative effects. According to CZ.NIC surveys, Czech users and organizations rank among the advantages of introducing IDN, clarity, simplicity, better readability and grammar and also it

can be used to protect their brand and it's quite common abroad. On the other hand, the disadvantages are mainly problematic communication with abroad, it can be confusing and there is possibility of errors. We must consider some IDN security issues [7].

3.2 IDNs in the other TLDs

IDNs are only possible with some TLDs as .com, .eu, .de, .pl, .at, .biz and others. One can learn more about introducing national alphabet characters into .eu domains at EURid website, a .eu domain administrator [8].

As can be seen in Figure 2 [9], since 2009, the number of registered IDN domains has been growing steadily. A particularly sharp increase is at second level IDNs between 2015-2016. Nearly all of the growth during that period is accounted for by .cn.

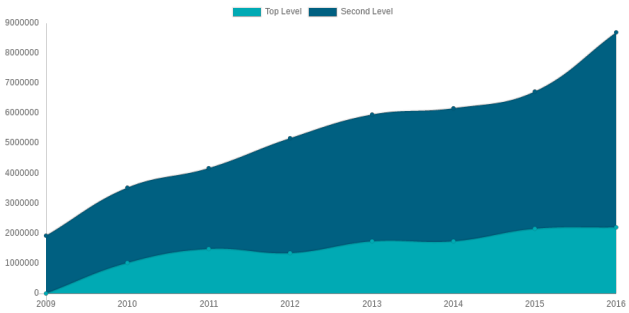

Figure 2. Growth of IDNs by level in the world [9].

World Map Growth of IDNs can be seen in Figure 3. This map shows the growth and distribution of IDNs from 2013 onwards. The distribution is mapped to the country of the host rather than the country of the registrant.

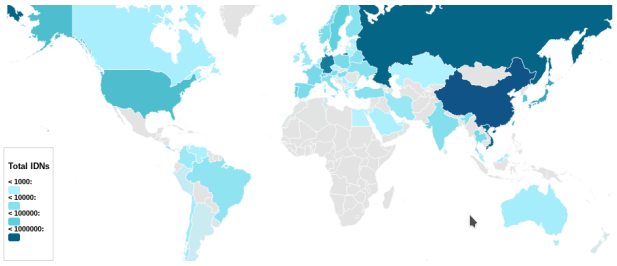

Figure 3. Total number of registred IDNs in the world [9].

Other interesting charts can be found on [9].

3.3 IDN security issues

There are some security issues, the most common ones are [7]:

1. **Typo-squatting** - the attacker registers the domain very similar to some of the most visited domains. For example, attacker place pornography or a fake bank page on his website, waiting for users to get here by mistake and get some sensitive data. To avoid the risk of typo-squatting, you can register multiple domains. One can register all combinations of common domains and accented domains, but there are many. The international system for resolving disputes between trade mark owners and domain name registrants is described in [10].
2. **DN homograph attack** is based on the interchangeability of some similar looking characters. This means that different interpretations of these similar characters confuse the user. An example is the "o" and "o" that looks the same, although it can be written both in Latin and Cyrillic. We get quite different from what we wanted, though we clicked on a link that is seemingly right. Homograph can also be related to the Latin (e.g. "I" and "l", "O" and "0").

3.4 Browsers

Browsers continue to be a bright spot for the use, display and resolution of IDNs [4]. Progress in browsers has been made steadily in the last years. All major web browsers support IDN.

4. EAI & IDN status survey

This chapter describes the differences between the RFC protocol specifications and the tested behavior of the applications. For the testing purposes, tools listed below were installed in a Ubuntu-based virtual machine.

4.1 Email protocols

The original **SMTP specification**, as with many other protocols, did not support UTF-8 characters. The change brought up SMTP extensions that allow UTF-8 characters in email headers and addresses. UTF-8 characters are supported in both, the user part of the address and the domain part. The domain part is converted to a string without diacritics using the Punycode, when formulating the DNS query.

At this time, none of the tested providers of **freemail services**, such as Seznam, Atlas, Centrum, Volny, Epošta, Yahoo, Gmail and Hotmail, has been able to create an accented email box. The use of accented characters occurring in the Czech alphabet such as "š","č" or"ř" leads to an error: "That email address contains invalid characters", when creating an e-mail box. Only Gmail support both, IDN and EAI. It's interesting, that mobile versions of Centrum.cz, Atlas.cz and Volny.cz support IDN only. A message can be successfully sent

to ”testmail@háčkyčárky.cz” and then the ”IDN OK” returns. I think it is rather an implementation error.

As far as **SMTP servers** are concerned, the situation is better. During testing, Postfix (3.1.0) and Exim (4.88) were able to work (send or receive) with EAI emails (EAI in header, such as ”TO:háčky@čárky.cz”). Only older servers Qmail (1.06) and Sendmail (8.15.2) did not support either IDN or EA.

Just like the original RFC for SMTP, **POP3/IMAP** RFC originally only counted with ASCII characters, EAI support was added later.

The first of the POP3 / IMAP server tests focused on finding and manipulating accented emails in email headers. Servers Dovecot (2.2.29), MS Exchange Server (2016) and Courier (0.78.2) passed this test seamlessly. Only Cyrus (2.4) could not find emails.

The second test was to create an accented account. I tried to create an account ”Talašová@talasova.cz”. Courier, Cyrus, and Dovecot were able to create this account. Only MS Exchange declined account creation due to unauthorized characters.

The third test was to log on to the server using the created account from the previous test. On Dovecot, MS Exchange Server and Courier servers, I was able to sign up for this account. Cyrus server only failed to sign in.

Among the best POP3 / IMAP clients in EAI support are Outlook 2016 and Mutt, who fully support EAI. The remaining tested clients support only IDN.

I first tested whether clients can send an email to an address without diacritics from the address with diacritics (the diacritics are located in the FROM field only). Thunderbird (45.8.0), Outlook 2016 and Mutt (1.5.24) allowed to send such an email. In contrast, in the Roundcube Webmail (1.2.4) client, such message can not be sent.

In the second test, I focused on testing IDN support. I was trying to send a message to autoresponder e-mail address: ”testmail@háčkyčárky.cz”. Thunderbird, Roundcube Webmail, Pošta, Outlook, and Mutt successfully submitted e-mail.

The third and final test was focused on testing support for EAI, that is, support for address with diacritics in the user section. I tried to send the message to ”talašová@talasova.cz”. Sending in Thunderbird and Roundcube Webmail was unsuccessful. Clients Mutt and Outlook 2016 allowed this message to be sent. The support status of UTF-8 strings is different for different clients. All tested clients supported the diacritics in the domain name. However, UTF-8 characters in the user part are mostly problem, especially in the ”TO” field.

The tests **results** show the general effort of application developers to implement EAI support and be prepared if the demand for the diacritics increases. In general, EAI support in e-mails and web browsers is quite good.

4.2 File Transfer Protocol

FTP is one of the oldest and widely used Internet protocols. The original character set of this protocol was 7 bit ASCII [11]. Nowadays, however, there is a need to use national characters. The original FTP specification (RFC 2640) has been extended to support different character sets. Not ASCII mode but binary mode is used for transferring text files containing national characters. UTF-8 characters within the file are not a problem for FTP. I tested Dolphin (4.14.3), FileZilla (3.26.2), FireFTP (2.0), gFTP (2.0.19), Konqueror (4.14.3), Total Commander (9.0a), WS_FTP (12.5), e-FTP (3.2.3.112) and Cerberus (8).

We will now focus on the use of diacritics in file names transmitted via FTP clients and servers. There is no problem with diacritics but with encoding. Transferring the file with accented name was not a problem with any of the clients, if the file was created and subsequently moved with the help of the same client. Encoding errors sometimes occurred, when different clients were used. For example, using one client to save the file to the server and use the other one to view or download this file. Clients have support for different encoding of file names. The safest is to always use UTF-8 encoding.

4.3 Session Initiation Protocol

SIP is a text-based protocol that uses UTF-8 encoding. The field format in the header is defined by header name. It can be a sequence of UTF-8 octets or a combination of white characters, tokens, delimiters, and quoted strings. Message bodies may use different encodings, this encoding should be reported in the value of the Content-Type header field. SIP messages may contain binary bodies; UTF-8 is assumed if the sender does not specify encoding. Caller names are encoded as UTF-8, so national characters can be used. UTF-8 characters can only be used for descriptive character field values and those fields that are not expected to be interpreted by the parser [12].

By RFC, the URI domain must contain only the selected ASCII characters. This is from the SIP URI syntax in the source [13]. However, for example, a Zoiper client sends a DNS query to the domain in the form given to it, whether punycode, hexadecimal, or UTF-8. Ekiga leaves the UTF-8 characters unchanged,

but the hexadecimal characters are changed to UTF-8, and punycode leaves also unchanged.

According to RFC, the user part of the SIP URI must contain only the selected ASCII characters or the hex hex sequence [13]. This is a system where "% xx" is written in place of the character, where x is the hexadecimal digits characterizing the bytes of the original character in UTF-8. In practice, each SIP client will handle differently an accented address. The Zoiper client accepts a user part in the form of talašová, tal% c5% a1ov% c3% a1 and tal% c5% a1. Accented characters are transmitted either in hexadecimal or as UTF-8 characters. While the Ekiga client change all these addresses to the hexadecimal form. In all cases, the Asterisk SIP server evaluated the address correctly.

4.4 Other protocols

These improvements in e-mails, TLDs and browsers can not hide the fact that, in other parts of the Internet, Universal Acceptance is at best marginal and in some cases non-existent [4]. In the coming years we will see how and in what direction will EAI and UA support be developed within the Internet.

5. Conclusion

This work has highlighted modern tendencies towards the use of national languages when using network services, primarily web and email protocols. Not only was the goal to familiarize readers with concepts such as IDN, EAI or UA, but also to provide a comprehensive view of the results of testing and reviewing the current state of implementation of IDN and EAI support in the Internet through various applications. The obtained information were used to develop module supporting national characters for lawful interception system. The main benefit is capture of data flows that include accents and security against attempts to hide communication using accented identifiers. And also general awareness raising about the possibilities, benefits and disadvantages of using UTF-8 characters in the Internet world.

Acknowledgements

I would like to thank my supervisor Ing. Libor Polčák, Ph.D. for his support.

References

[1] Don Hollander. Universal acceptance - an update, May 2016. https://www.icann.org/en/system/files/files/gdd-summit-amsterdam-ua-17may16-en.pdf.

[2] UASGtech. Universal acceptance, March 2018. https://uasg.tech/home/.

[3] UASGtech. Quick guide to eai, March 2018. https://uasg.tech/wp-content/uploads/2017/02/UASG014_20170206.pdf.

[4] EURid. Universal acceptance, March 2018. https://idnworldreport.eu/year-2017/universal-acceptance/.

[5] IETF. Internationalized domain names for applications: Definitions and document framework, March 2018. https://tools.ietf.org/html/rfc5890.

[6] Dynadot. What is punycode?, March 2018. https://www.dynadot.com/community/help/question/what-is-punycode.

[7] CZ.NIC. Cz.nic - idn - internationalized domain names, March 2018. https://hackycarky.cz.

[8] EURid. Eurid, March 2018. https://eurid.eu/cs/o-nas/eu-timeline/.

[9] EURid. Idn world report, March 2018. https://idnworldreport.eu.

[10] David Lindsay. *International Domain Name Law: ICANN and the UDRP*. Hart Publishing, 2007. ISBN: 9781847313966.

[11] IETF. Internationalization of the file transfer protocol, July 1999. https://tools.ietf.org/html/rfc2640.

[12] Radhika Ranjan ROY. *Handbook on session initiation protocol: networked multimedia communications for IP telephony*. CRC Press, 2016. ISBN 9781498747707.

[13] IETF. Sip: Session initiation protocol, March 2018. https://tools.ietf.org/html/rfc3261#section-25.1.