Email Address Internationalization (EAI): Evaluation of Major Email Software and Services

Between

Universal Acceptance Steering Group (UASG) &

Catalyst.Net Limited

30 August 2019

Report UASG021B

TABLE OF CONTENTS

Background

The Universal Acceptance Steering Group (UASG) is a community-based organization working towards the goal of Universal Acceptance (UA). Universal Acceptance is the idea that all domain names and email addresses should be treated correctly and consistently by Internet-enabled applications, devices, and systems. Specifically, this requirement includes new generic top-level domains (gTLDs), internationalized domain names (IDNs), and internationalized email addresses, which must be accepted, validated, stored, processed, and displayed as well as their traditional ASCII-based counterparts.

Because email messages constitute a significant part of online communications, it's important that email software and email service providers meet these requirements to achieve Universal Acceptance. Email Address Internationalization (EAI) is the protocol that allows email addresses that incorporate IDN or Unicode components to function correctly within the email software ecosystem. The purpose of this project is to evaluate the existing EAI capabilities of that ecosystem, in order to measure its “EAI Readiness”.

Project Structure

The first phase of this project, Discovery and Analysis, was completed in September 2018 and published on the UASG’s website as UASG021A. That initial phase analyzed a broad set of components from the email software ecosystem, defined the test criteria to use when evaluating a particular piece of software, and provided an estimate of the effort required to evaluate a select few representative software and service offerings from that set. The resulting Discovery and Analysis Report and accompanying test criteria can be found on the UASG’s website.

The second and current phase of this project comprises a proof of concept evaluation. A targeted set of software and service providers have been evaluated according to the criteria detailed by UASG021A. The purpose of this phase is to:

- Confirm whether the evaluation criteria developed in the UASG021A report are fair and reasonable, and amend or clarify them where necessary.
- Develop a template for reporting future test results.
- Determine the amount of effort required to evaluate additional email software and service providers.

The third and final phase of the project will provide a more complete evaluation of the email software ecosystem, including a greater number of components that have been informed by the experience of this pilot.

About this Document

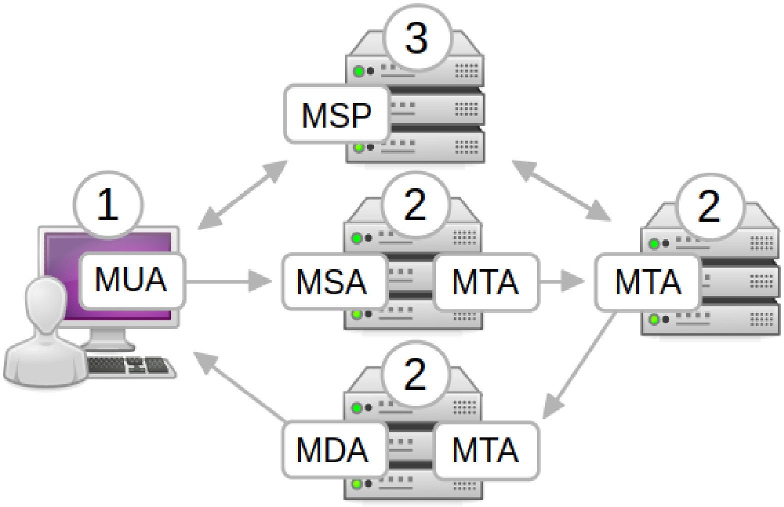
This document follows UASG021A and describes the proof-of-concept phase of the EAI Evaluation project. It provides clarifications and improvements to the test criteria developed in the first phase of the project and reports on the execution of those tests against the set of email software and services described above. Accompanying this document are test execution reports for the evaluated software and services, a spreadsheet containing the revised test cases and input data to use for future analyses, and a template that can be used when reporting future results. Finally, refined estimates of the effort required to evaluate the various types of software and service identified by UASG021A are given. This document assumes familiarity with UASG021A, so readers are encouraged to review that document before continuing.

Evaluation

The evaluation was carried out over the course of May and June 2019. It focused on three pieces of software encompassing five functional categories: MUA, MSA, MTA, MDA, and MSP. These categories are described by UASG 012, RFC 5598, and UASG021A section 3.2. Note that for the purposes of this project, the MDA category includes software that provides access to stored mail through a protocol such as IMAP or POP.

Because the goal of this pilot is to validate and refine the project's evaluation criteria, individual software components have not been named. Rather, throughout this report they are referred to as below:

- **Component 1**: which was evaluated in its capacity as a Mail User Agent ("MUA").
- **Component 2**: which was evaluated in its capacity as a Mail Submission Agent ("MSA"), Mail Transfer Agent ("MTA"), and Mail Delivery Agent ("MDA").
- **Component 3**: which was evaluated in its capacity as a Mail Service Provider ("MSP").



Functionally, these components comprise an end-to-end test of email composition, submission, transfer, delivery, and retrieval, although in practice they were tested individually rather than in an interconnected way. To enable testing, supporting software infrastructure such as mail servers and user accounts were required. Many of these requirements were known in advance and provisioned before testing commenced, while others were only discovered to be necessary during the process. Find a discussion of these requirements in **Error! Reference source not found.**.

The test cases and input data used for the evaluation can be found in the attached **Error! Reference source not found.** spreadsheet[1]. Where a test case's Action column refers to an <Email>, <Domain>, or <Unicode> value, that action was repeated once for each row in the corresponding worksheet. To enable these tests, email accounts and corresponding DNS records for these input data were created on newly-provisioned EAI and non-EAI mail servers. A set of addresses hosted by Coremail were also provided by the UASG to use while testing.

Component 1

Component 1 was evaluated as a Mail User Agent (MUA). This is the software that a user interacts with when reading or sending messages, so the analysis included significant user

[1] Note that this document contains updated test cases and differs from the one distributed with UASG021A.

interface testing. Two versions of the software were evaluated: a browser-based webmail interface provided by the vendor as a hosted service, and a mobile application for the Android operating system.

The web application was tested with the Chromium web browser on Ubuntu Linux, while the Android application was tested with two mobile devices. These were configured as SMTP submission clients for sending messages through the test accounts, and as IMAP and POP clients for retrieving messages from them.

The MUA test category contains both the largest number of individual test cases and the most repetition within them, and as such, Component 1 comprised the bulk of the testing effort. Additionally, because the majority of the tested behaviors require some manual action on the part of the user, for example, typing an email address or creating an address book entry, they cannot be triggered easily by another software tool and involve a fixed amount of manual effort.

Component 2

Component 2 is a mail server including related integrations such as webmail and mail retrieval services. This pilot focused on the MSA, MTA, and MDA-related aspects of the software, which are responsible for receiving messages from a Mail User Agent, relaying them, and ultimately delivering and making them accessible to another MUA.

The newest available version of this software was evaluated. It is distributed for users to install within their own infrastructure, so the necessary packages were configured and installed on a mail server, made accessible at both ASCII-only and IDN addresses, and instrumented such that server traffic for the ESMTP, POP3, and IMAP protocols could be inspected. An older version of the software without EAI support was also installed for comparison and to allow testing scenarios involving a non-EAI mail server.

The mail server-related test suites require the most advance configuration of the five test categories. However, given the prerequisites, they take less time than the MUA tests since many of the tests involve simply checking whether the software does or doesn’t support a given feature without needing to be repeated for multiple inputs.

Component 3

Component 3 is an enterprise software product whose features include a mail server and multi-user account hosting. A trial of the full software is available for evaluation purposes, as well as a free-to-use instance providing mailboxes for individual users, both of which were used for this pilot. These are hosted services, so no special configuration of the software was required beyond creating the test accounts.

This pilot analyzed the product as a Mail Service Provider (MSP), an entity that operates mail services on behalf of other organizations or users. The MSP test category takes the least amount of time to execute of the five simply because it involves the fewest tests. It focuses on a small set of the functionality typically provided by MSPs, specifically those relating to account and mailbox hosting. A Mail Service Provider will often provide a webmail interface and mailbox access services as well, but these were not tested during this evaluation.

Results

Summary

In general, the components tested in this pilot were found to behave well with respect to EAI. This was generally expected since they were selected in part because they were believed to be relatively advanced in their support of EAI, meaning that this pilot evaluation would exercise most of the test criteria defined by UASG021A. The pilot ultimately covered 118 of the 152 defined test cases.

Individual test reports for each component can be found in the appendices, as well as a reporting template on which these were based. Each test report includes a description of the component under test, a summary of the overall test results, and details of all non-passing test cases with a description of either the observed behavior or the reason the test could not be executed. In general, test cases were only skipped when a preceding test did not pass, meaning that further testing in that area of the software would not be possible or useful.

Component 1

Test results for the Component 1 web interface and mobile application are included (page 13). Both applications were found to behave well when presented EAI inputs for common use patterns such as reading and composing messages, but their POP3 and IMAP features are not yet EAI-ready.

Component 2

Test results for the Component 2 mail server are included (page 19). The software was found to be mostly EAI-ready, with the primary exception being that its POP3 server does not implement the “LANG” extension as defined by RFC 6856.

Component 3

Test results for the Component 3 service are included (page 24). This software handled EAI inputs correctly except where prevented by other software restrictions. The most significant such restriction was the rejection of mixed scripts, where for example left-to-right and right-to-left texts were combined within an address, or where Chinese text was combined with ASCII characters. This is an intentional software policy that needed to be worked around during the analysis.

Future Evaluations

A primary goal of this pilot was to identify potential pitfalls that might be encountered during future evaluations. This section provides an experience report and recommendations for future executors.

Prerequisites

A significant part of the effort was dedicated to setting up infrastructure in support of the tests. The term “infrastructure” is used loosely here to describe things like hardware, software, software configuration, user accounts, and network access rules. Indeed, nearly half of the effort involved in this pilot was consumed by tasks of this type *other* than running the tests themselves. We recommend that executors carefully consider the following prerequisites before commencing.

EAI and Non-EAI Mail Servers

All test categories involve some degree of interaction with an EAI-ready mail server. In this pilot, an instance of the Component 2 mail server was provisioned, tested, and found to be EAI-ready with respect to the extensions it advertises (per EAI-MSA-001 and EAI-MTA-001). As such, it could be reused for many of these integration tests. For the simple purposes of sending and receiving EAI messages, executors might use Gmail or Outlook.com, both of which are EAI-ready in this regard.

A non-EAI mail server is also required to test downgrading, correct rejection of messages, and so on. Currently, most open source mail servers are not EAI-enabled by default, so common software packages such as Postfix or Exim may be used for these purposes. However, executors should verify that the software has not been updated, or modified by an upstream provider, in a way that enables EAI features.

Note that, because many test cases require inspection of traffic sent to or from the server, the software must either be configured so as to disable encryption or instrumented in a way that provides the same sort of visibility.

EAI and Non-EAI Addresses

Many test cases require one to send and receive messages from a variety of email addresses, including EAI addresses. We were fortunate to have access both to pre-existing email accounts provided by the UASG for testing purposes, and to easy in-house provisioning of new addresses and DNS records. Future executors should ensure that they have access to functioning email addresses of all the types specified by the Test Cases spreadsheet in the <Email> Use Cases worksheet.

IDN Domains

Apart from email addresses, some of the tests require integration with software addressable at an IDN (for example EAI-MUA-048 and EAI-MUA-071). For this, a set of IDN records must be provisioned, or a hosted service available at an IDN domain must be used.

Mobile Phones and Mobile Service Subscriptions

Online services often require an active mobile phone subscription when registering an account or two-factor authentication for the account. In this pilot, both Component 1 and Component 3 required a mobile phone to register new accounts, authenticate logins, or access administrative interfaces. Moreover, one of these required a unique mobile phone number for each user account. Executors should determine in advance whether the components they will be testing require mobile phone accounts and how many.

Unrelated Software Issues

Occasionally, shortcomings in software used to facilitate the testing would interfere with the evaluation. This is to be expected but it does mean that time must occasionally be spent determining whether a given problem is caused by the software under test or something peripheral. Time for such troubleshooting has been factored into the **Error! Reference source not found.** section of this report.

Partial Test Execution

The attached test criteria are very thorough in that they require tests to be repeated for all possible address types specified in the <Email> Use Cases worksheet. This includes non-EAI addresses, which have long been widely-supported. Although these scenarios were tested during the pilot in order to establish the baseline behavior of the software under test, future executors may wish to skip these scenarios to save time. The same can be said for the ASCII-only scenarios in <Domain> and <Unicode> Use Cases worksheets.

The tests have also been classified according to the two phases of EAI Readiness, where Phase 1 involves sending to and receiving from EAI addresses, while Phase 2 involves hosting EAI addresses. Executors may wish to skip tests for the second of these phases.

Changes to the Test Suite

The test suite defined by UASG021A has been modified in a few ways during this exercise, and an updated version has been included with this report. Apart from a series of corrections and clarifications, the biggest change is a reordering of the test cases so that they follow a more logical order with respect to each test's prerequisites (and the tests renumbered accordingly) to make future evaluations more efficient. An example of this is in the Mail User Agent tests, where tests that do not rely on any external server have been moved before those that do, and those that do have been grouped based on the characteristics of the external server. A series of redundant test cases have also been consolidated.

Estimates of Effort

The first phase of this project provided tentative estimates of the effort involved in running this pilot. Having actually done so, we can now provide updated guidelines regarding the amount of effort an evaluation may involve depending on the type of software it aims to test.

- **Setup tasks:** 12 hours (applies once for all components)
- **Administrative tasks:** 4 hours (applies once for each component)
- **Mail User Agent:** 32 hours
- **Mail Submission Agent:**16 hours
- **Mail Transmission Agent:** 16 hours
- **Mail Delivery Agent:** 8 hours
- **Mail Service Provider:** 8 hours

As with the earlier estimates, these numbers assume ready access to the prerequisites described in the **Error! Reference source not found.** section (page 6) but do *not* include the time required to configure them for the evaluation; that time is captured within the setup and administrative tasks outlined above.

Setup tasks include provisioning infrastructure, creating DNS records, creating user accounts, and configuring software *other* than the software under test. These tasks can be performed once, and the resulting systems can be reused during later evaluations. In contrast, the Administrative tasks include time to configure the software under test, troubleshoot integration issues, collate results, and prepare a test report. These tasks must be repeated for each subsequent evaluation.

The estimates for each functional category can then be added to determine the effort involved in evaluating a particular piece of software based on the features it provides and which of those features are to be tested. For example, a piece of software providing an MSA, MTA, and MDA might be tested in full (40 hours excluding setup and administrative tasks) or in a single one of those functional capacities (16, 16, or 8 hours).

Document History

Version	Author	Date	Description
0.1	Evan Hanson	28 June 2019	Initial draft
0.2	Evan Hanson	8 July 2019	Include compiled test results
0.3	Evan Hanson	25 July 2019	Aggregate and anonymize test results
0.4	Evan Hanson	26 August 2019	Changes after feedback

Appendices

Test Cases

The revised test cases are defined within UASG021B Test Cases (pending publication).

Reporting Template

The template used for reporting test results is attached as UASG021B Reporting Template (pending publication).

Test Reports

Individual test reports for the software and services analyzed during this phase of the project are included in this document.

Test Execution Report: Component 1

About this Report

This report describes the EAI-related behaviors of Component 1 in its capacity as a Mail User Agent (MUA). It is part of a proof-of-concept evaluation of the EAI-readiness of the email software ecosystem, the details of which can be found in UASG021A (Email Address Internationalization – Discovery and Analysis). The software has been configured and evaluated according to the test criteria and procedures described in that report for the purposes of validating the criteria and providing an estimate of effort for future evaluations.

About Component 1

Component 1 is a free-to-use hosted email platform. In addition to mailbox hosting services, it provides multiple client applications including a browser-based webmail interface and a mobile application for the Android operating system, both of which were tested during this evaluation. Besides typical MUA features such as message composition and address book management, these applications also include support for sending and receiving mail through arbitrary email addresses via ESMTP, POP3 and, in the case of the mobile application, IMAP.

- **Type:** Mail User Agent (MUA)
- **Platform:**
    - Chromium browser (version 75) on Ubuntu Linux (version 18.04)
    - Samsung Galaxy J2 running Android OS (version 5.1)
    - Samsung Galaxy S8 running Android OS (version 7.0)
- **Date:** Evaluation occurred between 28 May and 2 July 2019

Evaluation Results

Summary

This evaluation analyzed the software's ability to compose and read messages, send messages to and from a variety of email addresses, and interact with remote servers using SMTP, POP, and IMAP protocols. These remote servers were configured in advance. A set of email accounts were created on the platform under test using the web-based interface, then associated with two instances of the mobile application.

The two versions of the application differ only very slightly in the features they support. In particular, the web application does not provide IMAP client functionality, so test cases EAI-MUA-040 through EAI-MUA-060 apply only to the mobile application. Otherwise, in the test results below, a passing test indicates that both the web application and mobile application worked correctly, while other results mean that one or both did not work correctly as noted.

The software's EAI-related behavior can be divided into two parts. The first part involves sending and receiving messages and managing contacts within a user's account hosted on the platform under test. For these tasks, the software performed well; the majority of EAI inputs were handled correctly and issues only appeared when domain label separators other than U+002E FULL STOP were used. The second part involves the software's integration with *other* email accounts via IMAP, POP, and SMTP connections. For these, the software performed less well; IDNs could not be used in general unless entered in A-label form, and its IMAP and POP client functionality did not support the features necessary for EAI message retrieval. Because of this, a large set of test cases could not be exercised.

Test Results

- **Total Tests:** 72
- **Passing Tests:** 39
- **Skipped Tests:** 25
- **Readiness Status:** N/A

Test Cases

Test ID	Summary	Status	Notes
EAI-MUA-001	EAI values can be saved to address book	**PASS**	
EAI-MUA-002	EAI values in address book display address local part as Unicode	**PASS**	
EAI-MUA-003	EAI values in address book display address domain part as Unicode	**PASS**	
EAI-MUA-004	EAI values can be recalled from address book	**PASS**	
EAI-MUA-005	EAI values recalled from address book display address local part as Unicode	**PASS**	
EAI-MUA-006	EAI values recalled from address book display address domain part as Unicode	**PASS**	
EAI-MUA-007	Address book search supports Unicode	**PASS**	
EAI-MUA-008	Originator fields accept EAI values	**PASS**	
EAI-MUA-009	Originator fields display EAI address local part as Unicode	**PASS**	
EAI-MUA-010	Originator fields display EAI address domain part as Unicode	**PASS**	
EAI-MUA-011	Destination address fields accept EAI values		Addresses containing label separators other than FULL STOP are not supported, with an error message displayed.
EAI-MUA-012	Destination address fields display EAI address local part as Unicode	**PASS**	
EAI-MUA-013	Destination address fields display EAI address domain part as Unicode	**PASS**	

Test ID	Summary	Status	Notes
EAI-MUA-014	Unstructured header fields accept Unicode	**PASS**	
EAI-MUA-015	EAI mailto link targets are supported	**PASS**	
EAI-MUA-016	Message text linkifies EAI values	**N/A**	Software does not linkify email addresses in message bodies.
EAI-MUA-017	SMTP server address can be specified by A-label	**PASS**	
EAI-MUA-018	SMTP server address can be specified by U-label		Web application's "Send mail as" feature worked correctly. Android SMTP configuration does not accept U-labels.
EAI-MUA-019	SMTP server address displayed as Unicode		Web application's "Send mail as" feature worked correctly. Android SMTP configuration does not accept U-labels.
EAI-MUA-020	Connection to SMTP server at IDN address is supported	**N/A**	Software does not support U-labels.
EAI-MUA-021	Username can be provided as Unicode	**PASS**	
EAI-MUA-022	Username displayed as Unicode	**PASS**	
EAI-MUA-023	EAI originator header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MUA-024	EAI destination address header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MUA-025	Unicode unstructured header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MUA-026	Unicode header values are not encoded with MIME encoded-words	**PASS**	
EAI-MUA-027	EAI originator header values are transmitted to non-SMTPUTF8 server as ASCII	**PASS**	
EAI-MUA-028	EAI destination address header values are transmitted to non-SMTPUTF8 server as ASCII	**PASS**	
EAI-MUA-029	Unicode unstructured header values are transmitted to non-SMTPUTF8 server as ASCII	**PASS**	
EAI-MUA-030	Message-ID of EAI message submitted to non-SMTPUTF8 server is ASCII-only	**PASS**	

Test ID	Summary	Status	Notes
EAI-MUA-031	SMTPUTF8 parameter is provided with MAIL command for EAI messages	**PASS**	
EAI-MUA-032	SMTPUTF8 parameter is not provided with MAIL command for non-EAI messages	**PASS**	
EAI-MUA-033	Multipart MIME message parts of type message/global are recognized	**PASS**	
EAI-MUA-034	Single-part MIME messages of type message/global are recognized	**PASS**	
EAI-MUA-035	Multipart MIME message parts of type message/global are sent to non-SMTPUTF8 servers with a valid content-transfer-encoding	**N/A**	The software does not allow customization of the content type or disposition.
EAI-MUA-036	Single-part MIME messages of type message/global are sent to non-SMTPUTF8 servers with a valid content-transfer-encoding	**N/A**	The software does not allow customization of the content type or disposition.
EAI-MUA-037	Local part of EAI values in atoms are displayed as Unicode	**PASS**	
EAI-MUA-038	Domain part of EAI values in atoms are displayed as Unicode	**PASS**	
EAI-MUA-039	Message lines longer than 998 characters are limited to 998 octets	**PASS**	
EAI-MUA-040	Server address can be specified by A-label	**PASS**	
EAI-MUA-041	Server address can be specified by U-label		Software does not accept U-labels.
EAI-MUA-042	Server address displayed as Unicode	**N/A**	Software does not support U-labels.
EAI-MUA-043	Connection to IMAP server at IDN address is supported	**N/A**	Software does not support U-labels.
EAI-MUA-044	Username can be provided as Unicode	**PASS**	
EAI-MUA-045	Username displayed as Unicode	**PASS**	
EAI-MUA-046	AUTHENTICATE command is used for authentication		The LOGIN command is used, rather than AUTHENTICATE.
EAI-MUA-047	UTF8=ACCEPT is enabled		The UTF8=ACCEPT option is never enabled.

Test ID	Summary	Status	Notes
EAI-MUA-048	SEARCH command does not specify charset	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-049	APPEND command uses UTF8 extension	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-050	Mailbox with Unicode name can be created	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-051	Unicode mailbox names can be accessed	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-052	Unicode mailbox names display as Unicode	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-053	Messages in mailbox with Unicode name can be retrieved	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-054	Message with EAI address in originator header is accessible	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-055	Message with EAI address in destination address header is accessible	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-056	Message with Unicode value in unstructured text header is accessible	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-057	Messages can be stored in mailbox with Unicode name	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-058	Message with EAI address in originator header can be stored	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-059	Message with EAI address in destination address header can be stored	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-060	Message with Unicode value in unstructured text header can be stored	**N/A**	IMAP client does not enable UTF8 support.
EAI-MUA-061	Server address can be specified by A-label	**PASS**	
EAI-MUA-062	Server address can be specified by U-label		
EAI-MUA-063	Server address displayed as Unicode	**N/A**	Software does not allow server address to be specified by U-label.
EAI-MUA-064	Connection to POP server at IDN address is supported	**N/A**	Software does not support U-labels.
EAI-MUA-065	Username can be provided as Unicode	**PASS**	
EAI-MUA-066	Username displayed as Unicode	**PASS**	

Test ID	Summary	Status	Notes
EAI-MUA-067	Connection to server with Unicode username is supported	**PASS**	
EAI-MUA-068	UTF8 mode is enabled		
EAI-MUA-069	STLS command is not used in UTF8 mode	**N/A**	POP client does not enable UTF8 support.
EAI-MUA-070	Message with EAI address in originator header is accessible	**N/A**	POP client does not enable UTF8 support.
EAI-MUA-071	Message with EAI address in destination address header is accessible	**N/A**	POP client does not enable UTF8 support.
EAI-MUA-072	Message with Unicode value in unstructured text header is accessible	**N/A**	POP client does not enable UTF8 support.

Test Execution Report: Component 2

About this Report

This report describes the EAI-related behaviors of Component 2 in its capacity as a Mail Submission Agent (MSA), Mail Transfer Agent (MTA), and Mail Delivery Agent (MDA). It is part of a proof-of-concept evaluation of the EAI-readiness of the email software ecosystem, the details of which can be found in UASG021A (Email Address Internationalization – Discovery and Analysis). The software has been configured and evaluated according to the test criteria and procedures described in that report for the purposes of validating the criteria and providing an estimate of effort for future evaluations.

About Component 2

Component 2 is an open source software project providing a mail transfer agent, POP3 and IMAP servers, and associated services such as integrated webmail and mailing lists. The project's source code is available for download at its website, and the software is also available in the form of precompiled packages for a variety of Unix-based operating systems.

- **Type:** Mail Submission Agent (MSA), Mail Transfer Agent (MTA), Mail Delivery Agent (MDA)
- **Platform:** Ubuntu Linux (version 18.04)
- **Date:** Evaluation occurred between 20 May and 24 June 2019

Evaluation Results

Summary

For this evaluation, two Linux servers were provisioned. On the first, all necessary software packages were downloaded, built, and installed. The Linux host was assigned DNS entries for each Domain Use Case and the software was configured to accept and send mail for each of these. User accounts for each Unicode Use Case were created, with mail for these delivered to a "Maildir" for each of those users for inspection. The software was configured to use the PLAIN authentication method for ESMTP, POP3, and IMAP connections. The second Linux server was configured similarly, but with a version of Component 2 without EAI support for use when testing behaviors involving a server without EAI capabilities.

By default, Component 2 will attempt to upgrade plain text connections to use TLS encryption. This feature was disabled so that network connections would not be encrypted. A piece of software called Wireshark was installed and configured to log traffic for these protocols to a file for inspection, allowing protocol-level messages exchanged between Component 2 and other systems to be reviewed. The software was then exercised using a combination of client services such as Gmail and Coremail and local utilities such as Sendmail and Telnet.

The software was found to handle EAI inputs well. Five of the 68 test cases found potential issues with EAI support, but only two of these represent missing features (the "LANG" extension for POP3 access, as defined by RFC 6856). Six of the test cases could not be exercised because the MSA does not attempt transform EAI messages into a form suitable for transfer to a non-EAI server as permitted by RFC 6531 (but rather rejects them), and the final one was skipped because the MDA did not add trace information to delivered messages. Due to the non-passing test cases, no level of EAI-readiness has been assigned in this report. However, because those test cases are advisory, and based on the overall results, we feel that this software is a positive example of EAI support.

Test Results

- **Total Tests:** 68
- **Passing Tests:** 56
- **Skipped Tests:** 7
- **Readiness Status:** N/A

Test Cases

Test ID	Summary	Status	Notes
EAI-MSA-001	SMTPUTF8 capability is advertised	**PASS**	
EAI-MSA-002	8BITMIME capability is advertised	**PASS**	
EAI-MSA-003	EHLO command argument is transmitted as ASCII	**PASS**	
EAI-MSA-004	SMTPUTF8 parameter is provided with MAIL command for EAI messages	**PASS**	
EAI-MSA-005	EAI reverse path values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MSA-006	EAI forward path values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MSA-007	EAI originator header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MSA-008	EAI destination address header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MSA-009	Unicode unstructured header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MSA-010	SMTPUTF8 parameter is not provided for non-EAI messages		SMTPUTF8 parameter is provided in all cases when destination server advertises SMTPUTF8 support.
EAI-MSA-011	EAI messages sent to non-SMTPUTF8 server are rejected or transformed	**PASS**	
EAI-MSA-012	EAI reverse path values are transmitted to non-SMTPUTF8 server as ASCII	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.
EAI-MSA-013	EAI forward path values are transmitted to non-SMTPUTF8 server as ASCII	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.
EAI-MSA-014	EAI originator header values are transmitted to non-SMTPUTF8 server as ASCII	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.

Test ID	Summary	Status	Notes
EAI-MSA-015	EAI destination address header values are transmitted to non-SMTPUTF8 server as ASCII	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.
EAI-MSA-016	Unicode unstructured header values are transmitted to non-SMTPUTF8 server as ASCII	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.
EAI-MSA-017	Message-ID of EAI message transmitted to non-SMTPUTF8 server is ASCII-only	**N/A**	The software rejects EAI messages submitted to a non-SMTPUTF capable server.
EAI-MTA-001	SMTPUTF8 capability is advertised	**PASS**	
EAI-MTA-002	8BITMIME capability is advertised	**PASS**	
EAI-MTA-003	EHLO command argument is transmitted as ASCII	**PASS**	
EAI-MTA-004	SMTPUTF8 parameter is provided for EAI messages	**PASS**	
EAI-MTA-005	Trace information includes domain in U-label form		Trace information includes the WITH protocol type “UTF8ESMTP” (with an S and/or A suffix as appropriate) rather than “UTF8SMTP” as specified by RFC 6531.
EAI-MTA-006	Trace information indicates SMTPUTF8 protocol	**PASS**	
EAI-MTA-007	EAI reverse path values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MTA-008	EAI forward path values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MTA-009	EAI originator header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MTA-010	EAI destination address header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MTA-011	Unicode unstructured header values are transmitted to SMTPUTF8 server	**PASS**	
EAI-MTA-012	SMTPUTF8 parameter is not provided for non-EAI messages		SMTPUTF8 parameter is provided in all cases when destination server advertises SMTPUTF8 support.

Test ID	Summary	Status	Notes
EAI-MTA-013	EAI messages sent to non-SMTPUTF8 server are rejected	**PASS**	
EAI-MDA-001	Trace information includes domain in U-label form	**PASS**	
EAI-MDA-002	Trace information indicates SMTPUTF8 protocol	**N/A**	Trace information is not applied by MDA.
EAI-MDA-003	Message with EAI address in originator header is delivered	**PASS**	
EAI-MDA-004	Message with EAI address in destination address header is delivered	**PASS**	
EAI-MDA-005	Message with Unicode value in unstructured header is delivered	**PASS**	
EAI-MDA-006	Message is delivered to EAI address	**PASS**	
EAI-MDA-007	Unicode username is accepted via IMAP	**PASS**	
EAI-MDA-008	IMAP UTF8=ACCEPT or UTF8=ONLY capability is advertised	**PASS**	
EAI-MDA-009	IMAP AUTHENTICATE command is supported	**PASS**	
EAI-MDA-010	IMAP ENABLE UTF8=ACCEPT command is accepted	**PASS**	
EAI-MDA-011	IMAP SEARCH command with CHARSET specification is rejected	**PASS**	
EAI-MDA-012	IMAP APPEND UTF8 command is accepted	**PASS**	
EAI-MDA-013	IMAP APPEND UTF8 preserves Unicode header values	**PASS**	
EAI-MDA-014	IMAP CREATE command with Unicode mailbox name is accepted	**PASS**	
EAI-MDA-015	IMAP SELECT command with Unicode mailbox name is accepted	**PASS**	
EAI-MDA-016	IMAP EXAMINE command with Unicode mailbox name is accepted	**PASS**	
EAI-MDA-017	IMAP SUBSCRIBE command with Unicode mailbox name is accepted	**PASS**	
EAI-MDA-018	IMAP LIST response includes Unicode mailbox name	**PASS**	
EAI-MDA-019	IMAP LSUB response includes Unicode mailbox names	**PASS**	
EAI-MDA-020	Mailbox with Unicode name can be created via IMAP	**PASS**	

Test ID	Summary	Status	Notes
EAI-MDA-021	Mailbox with Unicode name can be subscribed via IMAP	**PASS**	
EAI-MDA-022	Messages in mailbox with Unicode name are accessible via IMAP	**PASS**	
EAI-MDA-023	Message with EAI address in originator header can be stored via IMAP	**PASS**	
EAI-MDA-024	Message with EAI address in destination address header can be stored via IMAP	**PASS**	
EAI-MDA-025	Message with Unicode value in unstructured header can be stored via IMAP	**PASS**	
EAI-MDA-026	Message with EAI address in originator header is accessible via IMAP	**PASS**	
EAI-MDA-027	Message with EAI address in destination address header is accessible via IMAP	**PASS**	
EAI-MDA-028	Message with Unicode value in unstructured header is accessible via IMAP	**PASS**	
EAI-MDA-029	POP UTF8 USER capability is advertised	**PASS**	
EAI-MDA-030	POP UTF8 command is accepted	**PASS**	
EAI-MDA-031	POP UTF8 USER command is accepted	**PASS**	
EAI-MDA-032	POP LANG capability is advertised		The LANG extension is not supported.
EAI-MDA-033	POP LANG command is accepted		The LANG extension is not supported.
EAI-MDA-034	POP STLS command is rejected in UTF8 mode	**PASS**	
EAI-MDA-035	POP LIST command reports size of Unicode message as octet count	**PASS**	
EAI-MDA-036	Unicode username is accepted via POP	**PASS**	
EAI-MDA-037	Message with Unicode value in originator header is accessible via POP	**PASS**	
EAI-MDA-038	Message with Unicode value in destination address header is accessible via POP	**PASS**	

Test Execution Report: Component 3

About this Report

This report describes the EAI-related behaviors of Component 3 in its capacity as a Mail Service Provider (MSP). It is part of a proof-of-concept evaluation of the EAI-readiness of the email software ecosystem, the details of which can be found in UASG021A (Email Address Internationalization – Discovery and Analysis). The software has been configured and evaluated according to the test criteria and procedures described in that report for the purposes of validating the criteria and providing an estimate of effort for future evaluations.

About Component 3

Component 3 is an enterprise email software suite including most features common to enterprise email offerings. The software is available for purchase either as a downloadable package for installation within a client-managed environment or as a hosted service operated by the vendor. The company offers a trial of the hosted offering, as well as a branded, free-to-use instance of the software on which a user can specify a username to register an email account. This username is then used as the local part of their email address, while the domain part is selected by the user from a predefined set of vendor-managed domains spanning multiple languages.

- **Type:** Mail Service Provider (MSP)
- **Platform:** Chromium browser (version 75) on Ubuntu Linux (version 18.04)
- **Date:** Evaluation occurred between 17 June and 26 June 2019

Evaluation Results

Summary

This evaluation included both the free trial of the software suite and the free hosted instance. The former was tested from the point of view of an enterprise email administrator responsible for creating and managing user email accounts, while the latter was tested from the point of view of an individual user registering a personal email address at one of the vendor-managed domains.

A trial was registered under an ASCII domain. Within this trial account, the ability to create accounts with EAI local address parts and mailbox names was tested. Accounts under the hosted service were also created under the various domains in order to test the software's handling of address domain parts and the ability to send and receive messages. Local address parts and mailbox names were also retested on the hosted service for completeness.

The primary difficulty encountered during the evaluation was due to the application's rejection of mixed scripts, which is an intentional policy enforced by the software. According to this policy, for example, mixing Latin and non-Latin-based text within an address is forbidden. This is not strictly related to EAI, but it did interfere with the evaluation since some Use Cases needed to be adjusted. The vendor provided help when these issues were encountered, and in one case manually created an address for use during testing.

Test Results

- **Total Tests:** 12
- **Passing Tests:** 10
- **Skipped Tests:** 2
- **Readiness Status:** Phase 2

Test Cases

Test ID	Summary	Status	Notes
EAI-MSP-001	Local part of address containing Unicode is accepted	**PASS**	
EAI-MSP-002	Domain part of address containing Unicode is accepted	**N/A**	Domain part is provided by software and thus not configurable. It is consistently however displayed in U-label form as expected.
EAI-MSP-003	Local part of address containing Unicode is displayed as entered	**PASS**	
EAI-MSP-004	Domain part of address containing Unicode is displayed as entered	**N/A**	Domain part is provided by software and thus not configurable. It is however consistently displayed in U-label form as expected.
EAI-MSP-005	Address with ASCII equivalent of local part containing Unicode can be created	**PASS**	
EAI-MSP-006	Address with ASCII equivalent of domain part containing Unicode can be created	**PASS**	
EAI-MSP-007	Mailbox name containing Unicode is accepted	**PASS**	
EAI-MSP-008	Mailbox name containing Unicode is displayed as entered	**PASS**	
EAI-MSP-009	Address with local part containing Unicode receives messages	**PASS**	
EAI-MSP-010	Address with domain part containing Unicode receives messages	**PASS**	
EAI-MSP-011	Address with local part containing Unicode sends messages	**PASS**	
EAI-MSP-012	Address with domain part containing Unicode sends messages	**PASS**	