Measuring the UA-Readiness of Web Hosting Tools

13 September 2022

//*.*/

TABLE OF CONTENTS

About Nexperteam

For over 25 years, Nexperteam has been providing secure solutions in IT infrastructure, cloud technology, and DNS to customers worldwide and companies of various sizes from different business sectors. Nexperteam offers hosting and configuration, resilient DNS services with the strictest security standards, a wide range of cloud services, professional IT solutions, and 24/7 service support.

Nexperteam provides consultancy for registries and registrars and provides a full technical solution for trademark protection in the new generic top-level domain (gTLD) space.

Executive Summary

This UA-readiness assessment reveals that none of the web hosting tools tested (cPanel, Plesk and ISPConfig) comply with the defined objectives of the Universal Acceptance Steering Group (UASG). Since a number of underlying services are not ready to support local scripts, the creators of the web hosting tools block any non-ASCII characters throughout most of their interfaces.

Considering the three web hosting tools tested, we believe that Plesk has the highest chance of providing a full local script experience because it offers a file access tool. As such, the whole user experience could be driven through a web interface. With Plesk, email, domain name, user, database, and files can be managed using various web services all available from a local native script supporting web browser.

Such tools could also be installed on the other web hosting tools making them fully web browser managed but this would require additional installation and configuration steps. On the other hand, ISPConfig has the advantage of being an open-source web hosting tool and allows making ad hoc changes. For those willing and capable of doing their own patches and updates, this might be a good way forward to implement a full native script experience.

When it comes to the support of the underlying tools, we were surprised to see several frequently used tools lag behind in native script support. Dovecot as a POP/IMAP providing service should not interfere with anything encoding related although that is easier said than done. When looking through their code stack, it seems they have mainly been focusing on providing a rich feature set and supporting UTF-8 would be quite a task. As an alternative, one could use Courier. They seem to be aware of UTF-8 and the problems it brings but they seem to lag in support for modern features. None of the web hosting tools currently use Courier so we considered it out of scope.

The three web hosting tools tested are aware of local scripts as they provide additional services to translate their interface into any language. A decent set of already supported languages in various scripts does exist or can be accessed on demand. Overall, many of the underlying services have been made UTF-8 aware. This should drive the web hosting tools to support more local scripts in the future and provide a full local script experience.

Testing Status Overview

Web Hosting Tool	OS	Web Server Version	Email Server	Database	UTF-8 Environment	Access	Domain Name Activation and Management	Email Service	Database	Webhosting
cPanel 104.0.7	Rocky Linux 8.6 (Green Obsidian)	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	F	F	F	F	A
	Ubuntu 20.04 LTS	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	F	F	F	F	A
	CentOS Linux 7 (Core)	Apache 2.4.54	Exim 4.95	MySQL 5.7.38	W	F	F	F	F	A
Plesk Obsidian Web Host Edition Version 18.0.45	AlmaLinux 8.6 (Sky Tiger)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.14	MariaDB 10.3-32	W	F	F	F	F	A
					W	F	F	F	F	A
	Debian GNU/Linux 11 (bullseye)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.13	MariaDB 10.5-15 PostgreSQL 13.7	W	F	F	F	F	A
					W	F	F	F	F	A
	Windows 10	Microsoft IIS 10.0	MailEnable Standard 10.34	MariaDB 10.6	W	F	F	F	F	A
ISPConfig 3.2.2	openSUSE Leap 15.4	Apache 2.4.51	Exim 4.94.2	MySQL 8.0.29	F	W	W	F	W	A
	Oracle Linux Server 8.6	Apache 2.4.37	Postfix 3.5.8	MySQL 8.0.26	W	W	W	F	W	A
	Fedora Linux 36 (Server Edition)	Apache 2.4.54	Postfix 3.6.4	MariaDB 10.5-16	W	W	W	F	W	A

Legend

Test passed after additional steps (W)	Test passed when using ASCII credentials (A)
Test failed (F)	

Introduction

The Universal Acceptance Steering Group (UASG) is a community-based organization working towards the goal of Universal Acceptance (UA), 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.

Web servers, email servers, and backend databases provide the building blocks of a website's functionality. The UASG requested a UA-readiness study of the web hosting tools cPanel, Plesk, and ISPConfig, which are commonly used for hosting websites and enabling emails by Internet Service Providers (ISPs).

The aim of this work is to build capacity for small and medium businesses and organizations to provide UA in simple websites and the related email addresses which they operate. UA is constrained by the web servers, email servers, and other tools, which the ISPs provide for their customers, and by the interface and configuration options, which the ISP places in front of those building blocks.

Project Scope

The purpose of this project is to determine to what extent the web hosting tools offer the capacity for customers to build websites and host emails in accordance with Universal Acceptance in nine different configurations on different operating systems (Linux and Windows for Plesk; Linux for cPanel and ISPConfig).

It is important to understand that the assessment of the web hosting tools does not constitute any sort of approval or certification of UA-readiness, regardless of results.

The information gathered by this project is intended only as a snapshot of current capabilities and is not exhaustive in its coverage of UA-related behaviors. These results will become out of date as the web hosting tools develop over time.

On the flipside, this document describes the results of the UA-readiness of the "Web Hosting Tools" assessment and helps small and medium businesses and organizations choose the best configuration to provide Universal Acceptance in simple websites and in the related email addresses.

Assessment Procedure

This section describes the procedure used for analyzing the web hosting tools for UA-readiness.

Test Objectives

The objective of the assessment was to verify if the web hosting tools in the below-specified configurations are in full compliance with the objectives defined by the UASG. This means the activation and management of the domain name, as well as the webhosting and email service provided, need to be IDN-enabled, accept Unicode strings, and provide an easy and transparent way of working with them throughout the interface.

Scope of Testing

All basic functions relevant to UA for domain names and email addresses were tested for input, validation, storage, processing, and display:
- Domain names can be entered using native script and the correct encoding schemes are followed.
    - Supporting Unicode input, processing, and display in UTF-8 format.
    - Normalization of Unicode string into NFC form.
    - Conversion between U-label and A-label forms of the domain name.
    - Validating labels based on Unicode.
- Email addresses can be entered using native script and are encoded correctly.
    - Validating internationalized email addresses based on Unicode and IDNA2008.
    - Storing IDNs and internationalized email addresses.
- Email can be sent, received, and read in the appropriate script.
    - Sending out an email message using internationalized email addresses.
- Web service can be set up using native script.
- Management visualizes all these settings correctly and in the original script.

The purpose of the assessment was to determine if native scripts are accepted and handled correctly throughout the setup. This includes management of services as well as the services.

Test Scenarios

The following table provides the test scenarios and specific test cases that were used during the assessment.

Test Scenario #	Test Scenario Description	Test Cases High Level Overview
1	Test environment configuration	1.1 Check if the web server supports UTF-8 in default installation. 1.2 Check if the DB supports UTF-8 in default installation. 1.3 Check if email system is UTF-8 compliant.
2	Test access	2.1 Create accounts with different roles using UTF-8 credentials. 2.2 Check that those accounts can log in.
3	Test domain	3.1 Create domain names, check that they are properly visualized, and manage zones.
4	Test email	4.1 Create UTF-8 mailboxes that allow sending and receiving emails. 4.2 Create UTF-8 email aliases for the mailboxes. 4.3 Send and read emails to/from UTF-8 mailboxes and UTF-8 email aliases and check their content is ok.
5	Test database	5.1 Create DB user accounts with UTF-8 credentials and check that you can access the DB with those accounts. 5.2 Check that you can store and extract data in native scripts into and from the DB.
6	Test website hosting	6.1 Publish a website for the web domain names.

Test Cases

The test cases are detailed in the annexes. They are divided into specific testing areas to cover the main features of the web hosting tools.

Test Scenario	Test Category
Test environment configuration	ENV
Test access	ACC
Test domain	DN
Test email	MAIL
Test database	DB
Test website hosting	WEB

Each individual test case consists of the following information:
- Category that helps easily link the test cases to the main features of the web hosting tools (defined test scenarios).
- A unique identifier for the test case.
- A summary of the behavior to be tested.
- A description of the test.
- The action required to execute the test.
- The expected result.

Test Principles

- Testing will be focused on meeting the UA objectives and quality.
- There will be common, consistent procedures for all teams supporting testing activities.
- Testing processes will be well defined yet flexible, with the ability to change as needed.
- Testing activities will build upon previous stages to avoid redundancy or duplication of efforts.

Test Cycle

Nexperteam Test Team will perform:
- User accreditation and account setup in different scripts.
- Evaluation of the user experience for the end user in setting up various domain name services such as webhosting and email hosting.
- Testing of native script support when using the web interface to manage the services as well as testing the services themselves.
- Configuration changes in case these would have substantial impact on the services as they are delivered by the defined setups based on feedback.
- Reporting of any service that is deemed not to be UA-compliant by submitting bug reports to the specific web hosting provider.

Tested Native Scripts

The following native scripts have been tested: Arabic, Cyrillic (Bulgarian), Devanagari (Hindi), Greek, Hangul (Korean), Malayalam, Telugu, Thai, Simplified Chinese, Traditional Chinese, Latin (French).

Environments Setup

This section provides the installation details, default installation steps, and additional requirements to set up the nine configurations used for testing.

Initial Chosen Environment Configurations

In choosing the initial environment configurations, we researched the web servers, email servers, and databases supported by the three selected web hosting tools to be tested: cPanel, Plesk, and ISPConfig. To determine the most widely used configurations, we researched the top web servers*, email servers** and databases*** for the various platforms. All three suggested platforms have a set of out-of-the-box supported configurations. We considered these the low hanging fruit, as most providers would be inclined to use these configurations.

Web Hosting Tool	Web Server version	OS	Email Server	Database
cPanel	nginx	Rocky Linux 8.5	Exim	MySQL
	Apache	Ubuntu Server 22.04 LTS	Postfix	MariaDB
	LiteSpeed	Fedora 35 Server	Exim	PostgreSQL
Plesk	nginx	Alma Linux 8.5	Postfix	MySQL
	Apache	Debian 11.3	Exim	PostgreSQL
	IIS Web Server	Windows 10	MailEnable	MariaDB
ISPConfig	Apache	openSUSE Leap 15.3	Exim	MySQL
	Apache	Oracle Linux 8.5	Postfix	MySQL
	nginx	CentOS 7-2009	Postfix	MariaDB

*Based on w3techs statistics (https://w3techs.com/technologies/overview/web_server) as of 19 January 2022, the top three most used web servers are: nginx, Apache and Cloudflare Server summing up over 85% of the websites whose web servers are known.

**According to Stackoverflow (https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database) the most popular databases in use are MySQL and PostgreSQL. The MySQL replacement MariaDB is on the rise and is currently 7th.

***According to a survey (http://www.securityspace.com/s_survey/data/man.202112/mxsurvey.html) conducted by securityspace.com on January 1, 2022, the top three email servers on the Internet are Exim (with over 58%), Postfix (over 34%), and Sendmail (about 4%).

We initially installed all Linux distributions and then tried to install the web hosting tools in the configurations chosen above. It turns out that the web hosting tools are either not available on some of the chosen Linux distributions, or only available on a different OS version. Additionally, some web hosting tools do not support either the proposed email server or database.

We ended up setting up and testing the environment configurations provided in the table below.

//*.*/

Web Hosting Tool	OS	Web Server version	Email Server	Database
cPanel 104.0.7	Rocky Linux 8.6 (Green Obsidian)	Apache 2.4.54	Exim 4.95 (Dovecot 2.3.18)	MySQL 8.0.29 for Linux on x86_64 (MySQL Community Server - GPL)
	Ubuntu 20.04 LTS	Apache 2.4.54	Exim 4.95 (Dovecot 2.3.18)	MySQL 8.0.29 for Linux on x86_64 (MySQL Community Server - GPL)
	CentOS Linux 7 (Core)	Apache 2.4.54	Exim 4.95 (Dovecot 2.3.18)	MySQL 5.7.38, for Linux (x86_64) using EditLine wrapper
Plesk Obsidian Web Host Edition Version 18.0.45	AlmaLinux 8.6 (Sky Tiger)	Apache 2.4.37 - serves Plesk admins and resellers nginx 1.20.2 - serves end users	Postfix 3.5.14 (Dovecot 2.3.18)	MariaDB 10.3-32, for Linux (x86_64) using readline 5.1
	Debian GNU/Linux 11 (bullseye)	Apache 2.4.54 - serves Plesk admins and resellers nginx 1.20.2 - serves end users	Postfix 3.5.13 (Dovecot 2.3.18)	MariaDB 10.5-15, for debian-linux-gnu (x86_64) using EditLine wrapper - for Plesk management PostgreSQL 13.7 - for end users
	Windows 10	Microsoft IIS 10.0	MailEnable Standard 10.34	MariaDB 10.6
ISPConfig 3.2.2	openSUSE Leap 15.4	Apache 2.4.51	Exim 4.94.2 (Dovecot 2.3.15)	MySQL 8.0.29 for Linux on x86_64 (MySQL Community Server - GPL)
	Oracle Linux Server 8.6	Apache 2.4.37	Postfix 3.5.8	MySQL 8.0.26 for Linux on x86_64 (Source distribution)
	Fedora Linux 36 (Server Edition)	Apache 2.4.54	Postfix 3.6.4	MariaDB 10.5-16, for Linux (x86_64) using EditLine wrapper

Details on why we set up environments in configurations different from the initial chosen ones are provided throughout the installation procedures of the different web hosting tools detailed below.

Linux Distributions Installation

To install the Linux distributions a standard Linux KVM setup on a Centos 7 cluster was used. The VMs were configured to have 2 CPUs and 4GB of memory. Disk space consisted of a 20GB disk hosted on an SSD based Gluster cluster.

All distributions were installed using the default settings with no additional requirements or changes. Where possible, the server-based install was selected with slight variations on a per distribution base. The installations were performed using the provided (graphical) interface. All but one installation provided a standard server-like environment with no graphical interface. Oracle Linux was the notable exception, which made its install the slowest. All these machines were hooked up to the Internet via an IPv4 address.

cPanel Installation

To install cPanel, we followed the cPanel Installation procedure (https://docs.cpanel.net/installation-guide/install/).

The cPanel installation consists of a binary install file that contains an archive that can be unwrapped. This gives some insight into how the installation process is performed. This allows to a certain degree to understand what is going on and if things fail to determine what could be the root cause.

We ran the following command within a Linux `screen` session:
```
cd /home && curl -o latest -L
https://securedownloads.cpanel.net/latest && sh latest
```

cPanel Installation on Rocky Linux 8.6 (Green Obsidian)

As Rocky Linux and Alma Linux are equivalent distributions, we followed the cPanel Installation Guide - System Requirements for AlmaLinux OS (https://docs.cpanel.net/installation-guide/system-requirements-almalinux/).
```
# dnf install epel-release
# dnf install exim
# dnf install nginx
# dnf install mysql-server
```

We did an additional installation step to fix error "cPanel & WHM only supports Rocky Linux on version 106 or greater" following the fix as per https://forums.cpanel.net/threads/cpanel-whm-only-supports-rocky-linux-on-version-106-or-greater.702325/.

Using the above procedure, the install passed without problems.

cPanel Installation on Ubuntu 20.04

To install cPanel on Ubuntu 20.04, we followed the Installation Guide - System Requirements for Ubuntu (https://docs.cpanel.net/installation-guide/system-requirements-ubuntu/).

Despite our best efforts to install on Ubuntu 22.04, it did not work but support for it is expected to be provided soon. We could have fixed the perl IO::Net::SSL issue but were forced to abandon it when a set of packages, including some libidn dependencies, were flagged. It would be unfair to check UTF8 compliance with these prerequisites missing. With no specific delivery date set for support for 22.04, we decided to fall back on Ubuntu 20.04 and install cPanel on a classic Postfix/Mysql/Apache2 stack as per the current "supported combinations."

To install MySQL, additional signatures need to be imported as:
```
# curl -s http://repo.mysql.com/RPM-GPG-KEY-mysql-2022 | apt-key add -
```

Using the above describe procedure the install passed without problems.

cPanel Installation on CentOS 7

It worth mentioning that cPanel does not support NetworkManager. Before installing cPanel we followed the procedure for Disabling Network Manager (https://go.cpanel.net/disablenm).

Since Fedora is not supported and tuning the cPanel installation for Fedora was out of scope, the third cPanel install was done on CentOS 7.

To install cPanel on CentOS 7, we followed the Installation Guide - System Requirements for CentOS (https://docs.cpanel.net/installation-guide/system-requirements-centos/).

By default, cPanel uses MySQL to store information. However, it does offer the possibility to install PostgreSQL as an alternative database. This does require additional manual installation:
```
# yum install postgres-server
# sudo su – postgres
$ postgresql-setup initdb && exit
# systemctl start postgresql
```

Additional Configuration Steps for UTF-8 Support

We tested the environments (test scenario #1 - test environment configuration) for UTF-8 support and found out that in all cPanel environment configurations:
- The web server has no default "charset" set enabled. This is in accordance with standards. However, it also means that HTML files are best served with a header indicating the character set used.
- SMTPUTF8 support on email needs to be enabled. To enable SMTPUTF8, follow this procedure:
  - Exim https://support.cpanel.net/hc/en-us/articles/360053824393-How-to-use-SMTPUTF8-with-Exim. Alternatively set "Hosts to which to advertise the SMTPUTF8 SMTP option" in basic options to "enable" and enter "*" in the field.
  - Postfix https://www.postfix.org/SMTPUTF8_README.html#enabling. Alternatively execute the following command: postconf "smtputf8_enable = yes" and restart the postfix service.

Plesk Installation

Plesk installation comes as a stand-alone binary. It is almost impossible to get it installed on unsupported configurations. Furthermore, it uses a default set of services that it requires to be present. It will do so at the expense of uninstalling certain packages and installing whatever it deems necessary. This substantially limits the number of possible configurations.

Since the focus of the tests is on the users of the system and not on the packages providing the services, the enforced configurations were tested.

To install Plesk, we followed the official installation procedures, Install Plesk on Linux (https://docs.plesk.com/en-US/obsidian/deployment-guide/plesk-installation-and-upgrade-on-single-server/1click-plesk-installation/installing-plesk-for-linux-in-one-click.76444/) and Install Plesk on Windows (https://docs.plesk.com/en-US/obsidian/deployment-guide/plesk-installation-and-upgrade-on-single-server/1click-plesk-installation/installing-plesk-for-windows-in-one-click.76448/).

Additional Configuration Steps for UTF-8 Support

We have tested the environments (test scenario #1 - test environment configuration) for UTF-8 support and found out that in all Plesk environment configurations:

- The web server has no default charset set enabled. Most typical web tools set the header correctly, for plain HTML files you must specify the charset UTF-8.
- On AlmaLinux, the DB does not have UTF-8 enabled by default. To enable it:
  1. Add the following line to '/etc/my.cnf':
     ```
     character-set-server=utf8
     ```
  2. Restart the mysql server:
     ```
     # systemctl start mysqld
     ```
- SMTPUTF8 support on Postfix needs to be enabled. To enable SMTPUTF8 on email server:
  ```
  # postconf "smtputf8_enable = yes"
  # postfix reload
  ```

For more details, see Enabling Postfix SMTPUTF8 (https://www.postfix.org/SMTPUTF8_README.html#enabling) support.

ISPConfig Installation

Although most installation guidelines are slightly outdated, they provide excellent guidance to install ISPConfig 3.2 on the servers that were prepared.

ISPConfig Installation on OpenSUSE

We followed the installation instructions available for The Perfect Server - OpenSUSE 13.1 x86_64 (Apache2, Dovecot, ISPConfig 3) (https://www.howtoforge.com/the-perfect-server-opensuse-13.1-with-apache2-mysql-php-postfix-and-ispconfig-3).

Below are some installation exceptions from the default setup instructions:

```
# zypper install exim libmysqlclient-devel dovecot23 dovecot23-
backend-mysql pwgen cron python
Select to uninstall postfix and replace with Exim
# curl -LO https://dev.mysql.com/get/mysql80-community-release-sl15-
4.noarch.rpm
install mysql server
# zypper --no-gpg-checks install -y ./mysql80-community-release-sl15-
4.noarch.rpm
# zypper ref
# zypper install mysql-community-server
# systemctl start mysql
```

```
# systemctl enable mysql
# mysql -V
mysql  Ver 8.0.29 for Linux on x86_64 (MySQL Community Server - GPL)
# zypper install spamassassin
# sa-update
# zypper install apache2-mod_php7 php7
# zypper install roundcubemail
```

In addition, we also needed to:
- Make sure the system was detected as OpenSUSE by changing the install.php script to use 'os-release' as the release file.
- Identify the system as being compatible with OpenSUSE '11.2' – release 'unknown'.
- Make some changes to the MySQL database access rights of the ispconfig user to get everything working properly.

We are confident that someone who is used to the OpenSUSE stack will be able to address all these changes with little to no additional resources.

ISPConfig Installation on Oracle Linux

We followed the installation instructions available for The Perfect Server - OEL 5.4 [ISPConfig 3] (https://www.howtoforge.com/perfect-server-oel-5.4-ispconfig-3).

We used epel, remi repo to install phpmyadmin and other packages.

Oracle has its own Powertools repo, enable with "`dnf config-manager --set-enabled ol8_codeready_builder`"

```
[root@pfuasg7 install]# php install.php

-----------------------------------------------------------------------
---


 _____ _____  _____               __ _         ____
|_   _/ ___|  __ \ / __ \            / _(_)       /__  \
  | | \ `--.| |_/ / | /  \/ ___  _ __ | |_ _  __ _   _/ /
  | |  `--. \  __/  | |    / _ \| '_ \|  _| |/ _` |  |_ |
 _| |_/\__/ / |     | \__/\ (_) | | | | | | | (_| | ___\ \
 \___/\____/\_|      \____/\___/|_| |_|_| |_|\__, | \____/
                                              __/ |
                                             |___/

-----------------------------------------------------------------------
---
>> Initial configuration
Operating System: Oracle Linux Server 8.6
Following will be a few questions for primary configuration so be careful.
Default values are in [brackets] and can be accepted with <ENTER>.
Tap in "quit" (without the quotes) to stop the installer.
```

```
Select language (en,de) [en]:
Installation mode (standard,expert) [standard]:
Full qualified hostname (FQDN) of the server, eg server1.domain.tld
[pfuasg7.nexperteam.local]:
MySQL server hostname [localhost]:
MySQL server port [3306]:
MySQL root username [root]:
MySQL root password []: xxx
MySQL database to create [dbispconfig]:
MySQL charset [utf8]:
Configuring Postgrey
Configuring Postfix
Generating a RSA private key
.............................................................++++
.................................++++
writing new private key to 'smtpd.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:
State or Province Name (full name) []:
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:
Email Address []:
Configuring Mailman
Configuring Dovecot
Creating new DHParams file, this takes several minutes. Do not interrupt the
script.
Configuring Spamassassin
Configuring Amavisd
[INFO] service Rspamd not detected
Configuring Getmail
Configuring Jailkit
Configuring Pureftpd
Configuring BIND
Configuring Apache
Configuring vlogger
[INFO] service OpenVZ not detected
Configuring Bastille Firewall
[INFO] service Metronome XMPP Server not detected
Configuring Fail2ban
Installing ISPConfig
ISPConfig Port [8080]:
Admin password [a58855a0]: xxx
Re-enter admin password []: xxx
Do you want a secure (SSL) connection to the ISPConfig web interface (y,n)
[y]:
```

```
Checking / creating certificate for pfuasg7.nexperteam.local
Using certificate path /etc/letsencrypt/live/pfuasg7.nexperteam.local
Server's public ip(s) (89.207.184.57, 89.207.184.57) not found in A/AAAA
records for pfuasg7.nexperteam.local:
Ignore DNS check and continue to request certificate? (y,n) [n]:

Could not issue letsencrypt certificate, falling back to self-signed.
Generating a RSA private key
...................................................................
...................................................................
...................................................................
.............................++++
..++++
writing new private key to '/usr/local/ispconfig/interface/ssl/ispserver.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:
State or Province Name (full name) []:
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:
Email Address []:
Symlink ISPConfig SSL certs to Postfix? (y,n) [y]:
Symlink ISPConfig SSL certs to Pure-FTPd? Creating dhparam file may take some
time. (y,n) [y]:
/usr/bin/which: no acme.sh in
(/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin)
/usr/bin/which: no acme.sh in (/usr/local/ispconfig/server/scripts)
Configuring Apps vhost
Configuring DBServer
Installing ISPConfig crontab
no crontab for getmail
Detect IP addresses
Restarting services ...
Installation completed.
```

ISPConfig Installation on Fedora 36

We followed the installation instructions available for The Perfect Server - Fedora 15 x86_64 [ISPConfig 3] (https://www.howtoforge.com/perfect-server-fedora-15-x86_64-ispconfig-3) and in addition performed the following steps:

```
# wget https://openlitespeed.org/packages/openlitespeed-1.7.16.tgz
# dnf -y install libxcrypt-compat
# dnf -y install https://rpms.remirepo.net/fedora/remi-release-36.rpm
# dnf -y install php74-php-litespeed.x86_64
```

Additional required changes:

- You need to use php-7.4 to install/update isconfig.
- Make sure you update the PCRE2 library to fix a nasty bug in one of the early versions.
- Needed to install Amavis::Util

```
# dnf install cpan
# cpan install Amavis::Util
# chmod 755 /etc/amavisd/amavisd.conf
```

Additional Configuration Steps for UTF-8 Support

We tested the environments (test scenario #1 - Test environment configuration) for UTF-8 support and found out that in all ISPConfig environment configurations:

- The web server has no default charset set enabled. Most typical web tools set the header correctly, however, for plain HTML files you must specify the charset UTF-8.
- SMTPUTF8 support on Postfix needs to be enabled. To enable SMTPUTF8 on email server:

```
# postconf "smtputf8_enable = yes"
# postfix reload
```

For more details, see Enabling Postfix SMTPUTF8 (https://www.postfix.org/SMTPUTF8_README.html#enabling) support.

Test Results

The assessment reveals that **none** of the web hosting tools" in the above-specified configurations (or any other configuration) **are in full compliance** with the defined UA objectives of the UASG.

It is worth mentioning that none of the web servers and email servers have UTF-8 enabled by default. They require additional steps to enable it; steps provided accordingly in section Environments Setup. However, although UTF-8 support was enabled for email servers, the email tests did not pass.

After we set up the environments, we first tested the environment UTF-8 compliance, then performed additional steps where needed, and only afterwards, started the actual testing of the web hosting tools. One notable exception is Exim on openSUSE (ISPConfig). We could not enable STMPUTF8 for Exim as openSUSE's Exim package is not build with SMTPUTF8 support. The openSUSE team must have compiled the wrong Exim file. There was nothing we could do to fix this.

The table below provides the overall results per web hosting tool feature, including the UTF-8 compliance of the environment in default installation. To understand how the overall status has been provided, we refer to the test scenarios and see how they are broken down into high-level test cases. If for example, a test case needs additional steps (work) to pass, the overall feature status is "Test passed after additional steps (W)". The exception is the situation when a test case within a test scenario fails, and in this case, the overall feature status is "Failed / F".

Overall Test Scenario Status

Web Hosting Tool	OS	Web Server Version	Email Server	Database	UTF-8 Environment	Access	Domain Name Activation and Management	Email Service	Database	Webhosting
cPanel 104.0.7	Rocky Linux 8.6 (Green Obsidian)	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	F	F	F	F	A
	Ubuntu 20.04 LTS	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	F	F	F	F	A
	CentOS Linux 7 (Core)	Apache 2.4.54	Exim 4.95	MySQL 5.7.38	W	F	F	F	F	A
Plesk Obsidian Web Host Edition Version 18.0.45	AlmaLinux 8.6 (Sky Tiger)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.14	MariaDB 10.3-32	W	F	F	F	F	A
					W	F	F	F	F	A
	Debian GNU/Linux 11 (bullseye)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.13	MariaDB 10.5-15 PostgreSQL 13.7	W	F	F	F	F	A
					W	F	F	F	F	A
	Windows 10	Microsoft IIS 10.0	MailEnable Standard 10.34	MariaDB 10.6	W	F	F	F	F	A
ISPConfig 3.2.2	openSUSE Leap 15.4	Apache 2.4.51	Exim 4.94.2	MySQL 8.0.29	F	W	W	F	W	A
	Oracle Linux Server 8.6	Apache 2.4.37	Postfix 3.5.8	MySQL 8.0.26	W	W	W	F	W	A
	Fedora Linux 36 (Server Edition)	Apache 2.4.54	Postfix 3.6.4	MariaDB 10.5-16	W	W	W	F	W	A

Legend

Test passed after additional steps (W)	Test passed when using user with ASCII credentials (A)
Test failed (F)	

Detailed Test Status Per High-level Test Case

Web Hosting Tool	OS	Web Server Version	Email Server	Database	1. Test environment configuration			2. Access		3. Domain	4. Email			5. Database		6. Website Hosting
					1.1.	1.2.	1.3.	2.1	2.2.	3.1.	4.1.	4.2.	4.3.	5.1.	5.2.	6.1.
cPanel 104.0.7	Rocky Linux 8.6 (Green Obsidian)	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	P	W	F	N	F	F	N	N	F	A	A
	Ubuntu 20.04 LTS	Apache 2.4.54	Exim 4.95	MySQL 8.0.29	W	P	W	F	N	F	F	N	N	F	A	A
	CentOS Linux 7 (Core)	Apache 2.4.54	Exim 4.95	MySQL 5.7.38	W	P	W	F	N	F	F	N	N	F	A	A
Plesk Obsidian Web Host Edition Version 18.0.45	AlmaLinux 8.6 (Sky Tiger)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.14	MariaDB 10.3-32	W	W	W	F	N	PP	F	N	N	F	A	A
	Debian GNU/Linux 11 (bullseye)	Apache 2.4.37 nginx 1.20.2	Postfix 3.5.13	MySQL 15.1 PostgreSQL 13.7	W	P	W	F	N	PP	F	N	N	F	A	A
	Windows 10	Microsoft IIS 10.0	MailEnable Standard 10.34	MariaDB 10.5-15 PostgreSQL 13.7	W	P	W	F	N	P	F	N	N	F	A	A
ISPConfig 3.2.2	openSUSE Leap 15.4	Apache 2.4.51	Exim 4.94.2	MySQL 8.0.29	W	P	F	W	PP	W	PP	W	F	W	P	A
	Oracle Linux Server 8.6	Apache 2.4.37	Postfix 3.5.8	MySQL 8.0.26	W	P	W	W	PP	W	PP	W	F	W	P	A
	Fedora Linux 36 (Server Edition)	Apache 2.4.54	Postfix 3.6.4	MariaDB 10.5-16	W	P	W	W	PP	W	PP	W	F	W	P	A

//*.*/

Legend

		The numbered header row accommodates the test scenarios numbering. For details, see Test Scenarios.
Test passed (P)	Test failed (F)	
Test passed after additional steps (W)	Cannot test (N)	
Test passed when using user with ASCII credentials (A)	Partially Passed (PP)	

The "Partially Passed" status was given to indicate that not all steps within a high-level test case have passed. For more detailed test statuses, you can always refer to the excel files that contain test cases and execution steps.

Installation Results

When installing the different Linux distributions, it is apparent that installing any of these has become quite straightforward. Most of the underlying software does support UTF-8 character sets in some form either out of the box or by some simple changes in the configuration. For instance, all installed FTP services are configured to use a database. The link with this database is UTF-8 compliant if the database is set to be UTF-8 compliant.

The same goes for tools such as virus scanner, SPAM protection, database, DNS, and file storage. Even the Linux user management system allows for UTF-8 compliant users albeit with the note that you would have to supply the "--badname" flag to create the user (see manpage about useradd).

Some of the tools installed already embrace the UTF-8 character set and various global languages. Most notably, phpMyAdmin, a tool to manipulate the database and its content via a web interface, and RoundCubeMail, a tool to read and manage your personal email box via a web interface.

Access (Accounts Creation) Results

Visiting the web hosting tools shows that all three take the prudent approach of blocking UTF-8 characters when it comes to user identification and email addresses. A decision that considering the above arguments is certainly understandable.

We took the liberty to circumvent a number of these on ISPConfig. Since this tool is open source, it can easily be adapted and changed at will. If one does make this type of change, it shows the strengths and the weaknesses of the underlying system and services. Some are quick and easy fixes as with the DNS provided system.

Here the accept, validate, process, and store cycle have been ironed out already. Both UTF-8 as well as IDNA A-labels can be provided. The only thing missing is the display step where the names are visualized in the ASCII-based format rather than the desired UTF-8 format. By simply adding the filter that is already provided, the display part can be fixed to do the right thing.

At this point most of the services have embraced UTF-8 to a certain extent. With little effort, some of the experiences can be made a lot less painful. All web hosting tools do provide a set of languages they support and allow a specific language to be added on demand. Some of the tools allow for an almost 100 percent web-based experience. If all underlying services would support UTF-8, it would be possible for small businesses to create and manage their web presence and email through an all-native interface. However due to the lack of support of some tools at this point, none of the web hosting packages are close to providing such an experience.

Email Results

There were certainly some disappointing pain points where the lack of compliance shows. One of the major pain points is email. Postfix, the most widely used tool to send and receive email, can be configured to support SMTPUTF8. However, only the local part of the email address can be shared in UTF-8 format. The part specifying the domain name must be converted to IDNA A-label to be accepted.

Going further down the email path, the service that provides POP and IMAP delivery services does not support UTF-8. So even if the tool sets described above would be made

UTF-8 compliant and the web-hosting tool would allow for UTF-8 email addresses, the delivery still would fail.

Getting the email system to work turns out to be difficult. Even though setting up the web interface to accept UTF-8 email addresses turns out to be just a small challenge, getting the full-service stack to work is a daunting task. Note that even the build-in PHP email check on which ISPConfig relies does not allow UTF-8 characters (neither in the local part, nor in the domain name part).

Domains

cPanel does not allow creation of IDNs and zone records in native scripts. You must use an external IDN converter and create domain names and add zone records in Punycode. This affects the user experience heavily as it would make it nearly impossible to manage your DNS records from within the provided web hosting tool.

Plesk on Windows allows the creation of IDNs and zone records in native scripts and their proper visualization in the interface; Plesk on Linux allows it, except for Malayalam. For Malayalam, Plesk on Linux fails in some situations (it does not work out for some domain names). The Plesk team will investigate this.

ISPConfig requires installation of an additional patch for the proper visualization of IDNs in the web interface.

Databases

Both cPanel and Plesk do not allow the creation of databases and DB users in native scripts.

When allowing native script databases to be generated from the ISPConfig web interface, a double or mismatching encoding problem arises. Although the metadata such as database name and database user seem to be validated and stored consistently, we can clearly see it is not visualized correctly. We are not sure if this is due to a misconfiguration in the PHP MySQL layer or if this is related to a MySQL library mismatch problem. Further examination is needed but deemed out of scope for this study.

Web Hosting

Another thing to consider is the client tools that must be used to access the web hosting tools. Notably, if you try to upload files you will need an FTP/SFTP client. Although there are many of these tools available, are they providing this service in native scripts and are they allowing UTF-8 login credentials? These are questions that are out of scope for this particular report.

Bug Reports and Suggested Features

cPanel Suggested Features

We initially submitted technical requests to cPanel Support and they indicated that it is currently impossible to create UTF-8 accounts, DB user accounts, email addresses, and IDNs.

cPanel Support also suggested us to submit feature requests and so we did. The status of the feature requests is provided below.

It's also notable that feature requests go through a voting phase, and we do not have insight on how much time it takes for the cPanel development team to address them.

Below is the list of feature requests we submitted.

Topic 1 link: https://features.cpanel.net/topic/22493-utf-8-support-for-creation-of-mysql-databases-and-db-users

Topic 2 link: https://features.cpanel.net/topic/22492-creation-of-utf-8-email-accounts

Creation of UTF-8 Email accounts   `Not Planned`

Oana Bota shared this idea 47 days ago

As a cPanel admin and client, I would like to create UTF-8 email accounts to serve clients in regions where native scripts are used (e.g. Arabic, Cyrillic – Bulgarian, Devanagari – Hindi, Greek, Hangul – Korean, Malayalam, Telugu, Thai, Simplified Chinese, Traditional Chinese, Latin- French>with accent letters) and streamline the user experience.

Right now there are restrictions on the username portion of the email address; it must use ASCII characters. Specifically, the username can only contain letters, numbers, periods, hyphens, and underscores.

1   I like this idea   **0**   Stop Following

Topic 3 link: https://features.cpanel.net/topic/22491-creation-of-cpanel-accounts-with-utf-8-credentials-native-scripts

Creation of cPanel accounts with UTF-8 credentials (native scripts)   `Not Planned`

Oana Bota shared this idea 47 days ago

As a cPanel user (admin, client), I would like to create cPanel accounts using UTF-8 credentials (both username and password), to serve clients in regions where native scripts are used (e.g. Arabic, Cyrillic – Bulgarian, Devanagari – Hindi, Greek, Hangul – Korean, Malayalam, Telugu, Thai, Simplified Chinese, Traditional Chinese, Latin- French>with accent letters) and streamline the user experience.

Right now there are restrictions on cPanel usernames. At this time, usernames are only allowed to use ASCII characters, or more specifically, a-z and 0-9.

1   I like this idea   **0**   Stop Following

Topic 4 link: https://features.cpanel.net/topic/22490-idn-and-utf-8-zone-records-creation-and-display

IDN and UTF-8 zone records creation and display   `Needs Review`

Oana Bota shared this idea 47 days ago

As a cPanel user (admin and client), I would like IDN support, to serve clients in regions where native scripts are used (e.g. Arabic, Cyrillic – Bulgarian, Devanagari – Hindi, Greek, Hangul – Korean, Malayalam, Telugu, Thai, Simplified Chinese, Traditional Chinese, Latin- French>with accent letters) and streamline the user experience.

In this regards, I'd would love to have the possibility to add IDNs and UTF-8 zone records and visualize them as such and not in punnycode as it makes it difficult to spot the right DN/zone record in the UI (requiring more effort to use an IDN converter).

1   I like this idea   **0**   Stop Following

Plesk Suggested Features

We started by asking questions on the Plesk Obsidian Forum. Below is the list of question:

https://talk.plesk.com/threads/utf-8-compliance-of-plesk-interface.365628/

https://talk.plesk.com/threads/domain-creation-in-malayalam-script-fails.365630/

Plesk forum moderators suggested that we submit reports, which we tried to do.

The following two reports were forwarded to their developers:

Report Summary	Plesk Forum Report Link	Plesk Bug No.	Additional information from Plesk
Plesk account creation using UTF-8 credentials fails	https://talk.plesk.com/threads/plesk-account-creation-using-utf-8-credentials-fails.365646/#post-912028	PPPM-13658	The bug they mention has been submitted only for allowing UTF-8 characters in Username.
Domain creation in Malayalam script fails	https://talk.plesk.com/threads/domain-creation-in-malayalam-script-fails.365647/	PPPM-13652	Some Malayalam DNs work, e.g., മല.യാളം

The following two reports were deleted for being a "duplicate". We asked Plesk to provide us the links to the accepted reports, which were submitted by someone else; however, we got no answer.

Report Summary	Plesk Forum Report Link which was available before Plesk Team deleted the reports
Creation of UTF-8 email accounts fails	.../threads/creation-of-utf-8-email-accounts-fails.365649/
Creation of DB user accounts with UTF-8 credentials fails	.../threads/creation-of-db-user-accounts-with-utf-8-credentials-fails.365651/

ISPConfig

As ISPConfig is an open source tool, and our team has the technical knowledge to make the proper code changes, we did so and created patches to allow UTF-8 accounts, email, and IDN creation.

Nevertheless, the code changes we made partially eliminated some of the limitations on the creation of UTF-8 accounts, email, and IDNs. As they do not provide a 100 percent working solution, we think it is not worth submitting these patches to the ISPConfig on GitHub.

ISPConfig code changes can be further implemented to make this web hosting tool UTF-8 compliant (e.g., to allow creation of UTF-8 FTP and SFTP users), however it requires a lot of effort, and it is out of this project scope. We might attempt this in the near future without time constraints.

Annexes

Below is the list of Annexes that are relevant for the testing report.

Name	Description
Test Data	Data that was used during the testing
Test Reports – 9 excel files	The test cases tested for the 9 environment configurations.
ISPConfig Patch	The code changes we did for ISPConfig.