

NAME

send_email - Send an email using an RFC 5322 formatted file.

DESCRIPTION

Synopsis:

send_email.py message_file [...]

Description:

This program sends emails using the contents of the message_file given as arguments to this program.

Each file must be formatted using the RFC 5322 format and contain at least the "To" and "Subject" header lines. Optionally it can have the header "Attachment" if there is a file to be attached to the email. All other header lines are ignored. The "From" address will be the device name and the user account name from the dotenv configuration.

Every message is tested for properly formatted "To:" address and that there is a "Subject:" line. If these checks fail, the message is written to the "other_msg" folder for inspection.

The validated message is then send to the SMTP server using the dotenv credentials.

When a message has been emailed successfully, it is moved to the 'unqueue' folder where it is awaiting deletion. Otherwise the file is moved to the queue folder. This allows a message that was not send to be tried again at the next running of this program.

Arguments:

message_file -- The RFC 5322 formatted message(s).

Return:

0 -- No errors detected.
1 -- Error detected. Errors are written to the logging file.
3 -- No internet access or SMTP connection failed.

Run info:

This program must be run from the send_email.sh script.

See also:

dotenv -- Dotenv file with the secrets and other configurable data.
send_email.sh -- Script that executes this program.
unqueue/ -- Folder for storing emailed message.
queue/ -- Folder that holds messages to be queued for sending.
signage.conf -- Configuration data providing environment variables.
Only if program executed by a shell script.

Version:

1.0

Copyright (C) 2023 Gert Bakker, Coleshill, UK

This file is part of the <Digital Signage> project.

Digital Signage is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, version 3.

Digital Signage is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

FUNCTIONS

email_checks(p_to_adrs, p_subject, p_msgname)
Email TO and SUBJECT sanity checks.

Check that the TO string is not empty or None. Then extract the email address from the TO string (if needed) and then check that the email address has the correct format.
Check that the SUBJECT string is not None or empty.
Any errors, return with a False value.

Arguments:

p_to_adrs -- The TO email address to be checked
p_subject -- The email subject text to be checked
p_msgname -- Name of the message file to include in error messages

Return:

True -- The TO and SUBJECT are OK
False -- The TO and/or SUBJECT data are invalid

initialise()

Initialise the program.

Declare variables used in other parts of the program as Global.
Set the hostname to the name of the device for use in the From: line of the email message.
From the dotenv file, get the user and server data.
Set the logging level depending on the value of the dotenv DEBUG variable.

Arguments:

None

Return:

None

main()

Main control of sending email in an RFC 5322 formatted file.

Initialise the program and its global variables.
Check that there is at least one argument given. This must be an RFC 5322 formatted message filename.
Call the function to open the SMTP link and to logon to the email account. This involves an internet access check. If failed, exit.
Call the function to send the email. Outcome is True if the email was send successfully and False if there was an error detected.
When a file has been emailed, it is moved to the "unqueue" folder so that it can be deleted by the script in which this program is run.
Message file that was not send correctly is moved back to the queue folder to be tried again later.

Arguments:

None

Return:

0 -- Emailed message OK
1 -- Error detected. See the logging messages file.
3 -- SMTP link could not be established.

open_smtp_server()

Open and connect to the SMTP server.

First check that we have internet access. Then open a link to the SMTP server and log-on to the email account.

Arguments:

None

Return:

True - Internet access is OK and logged onto the SMTP server
False - There is no internet access or SMTP logon failed

send_the_message(p_filename, p_from_adrs)

Send the message contained in the file passed in p-filename.

Send the message that is in the RFC 5322 formatted file.

Extract from the header the "To" and "Subject" information.

If the "Attachment" header is present, then the file named on this line is to be attached to the email message.

Other header lines in the message are ignored.

Arguments:

p_filename -- Name of the file with the message to email.

p_from_adrs -- The sender address for the email to use.

Return:

True -- Send message OK

False -- Error detected.