

fn-outbound-email Functions for IBM Resilient

- [fn-outbound-email Functions for IBM Resilient](#)
 - [Release Notes](#)
 - [v1.1.0](#)
 - [v1.0.9](#)
 - [v1.0.8](#)
 - [Requirements](#)
 - [Installation](#)
 - [Installation \(App Host\)](#)
 - * Once installed, navigate to the app's Configuration tab and edit the app.config file updating the [\[resilient\]](#) section as necessary and updating the [\[fn_outbound_email\]](#) section as necessary.
 - [Installation: New users](#)
 - [Installation: Existing Users](#)
 - [Common connection issues with TLS and TroubleShooting](#)
 - [Uninstall](#)
 - [Troubleshooting](#)
 - [Resilient Action Status](#)
 - [Resilient Scripting Log](#)
 - [Resilient Logs](#)
 - [Resilient-Circuits](#)
 - [Support](#)
-

Release Notes

v1.1.0

- Bug fixes and send all or specific attachments

v1.0.9

- Apphost compatibility

v1.0.8

- Initial Release after internal development by Professional Services, no prior release notes
-

Requirements

- IBM Resilient >= [v33.0.5112](#)
 - An Integration Server running [resilient_circuits](#)
 - The minimum set of Resilient API permissions for this integration if using an API key account:
 - Org Data: Read and Edit
 - Incident: Read
 - Functions: Read
 - To set up an Integration Server see: [ibm.biz/res-int-server-guide](#)
-

Installation

Installation (App Host)

With App Host, all the run-time components are pre-built. Perform the following steps to install and configure:

- Download the `app-fn_outbound_email-x.x.x.zip`.
- In Resilient navigate to **Administrator Settings > Apps**
- Click the Install button and select the downloaded `app-fn_outbound_email-x.x.x.zip`. This will install the associated customizations.
- Once installed, navigate to the app's Configuration tab and edit the `app.config` file updating the `[resilient]` section as necessary and updating the `[fn_outbound_email]` section as necessary.

Installation: New users

- Download the `fn_outbound_email.zip`.
- Copy the `.zip` to your Integration Server and SSH into it.
- **Unzip** the package:

```
$ unzip fn_outbound_email-x.x.x.zip
```

- **Change Directory** into the unzipped directory:

```
$ cd fn_outbound_email-x.x.x
```

- **Install** the package:

```
$ pip install fn_outbound_email-x.x.x.tar.gz
```

- Import the **configurations** into your `app.config` file:

```
$ resilient-circuits config -u
```

- Import the `fn_outbound_email` **customizations** into the Resilient platform:

```
$ resilient-circuits customize -y -l fn-outbound-email
```

- Open the config file, scroll to the bottom and edit your `fn_outbound_email` configurations:

```
$ nano ~/.resilient/app.config
```

Config	Required	Example	Description
<code>smtp_server</code>	Yes	<code>xxx.xxx.xxx.xxx</code>	server IP or <code>smtp.example.com</code>
<code>smtp_user</code>	Yes	<code>''</code>	Blank on port 25, login email on 587

Config	Required	Example	Description
smtp_password	Yes	``	Blank on port 25, login password on 587/2525
smtp_port	Yes	25	Defaults to unauthenticated, 587/2525 for TLS
smtp_conn_timeout	Yes	15	Time in seconds
smtp_ssl_mode	No	None	Not supported
template_file	No	fn_outbound_email/data/example_send_email.jinja	Optional - Path to a custom template file for formatting HTML email/ differs in apphost

- **Save** and **Close** the app.config file.
- Run selftest to test the Integration you configured:

```
$ resilient-circuits selftest -l fn-outbound-email
```

- **Run** resilient-circuits or restart the Service on Windows/Linux:

```
$ resilient-circuits run
```

Installation: Existing Users

- Prior to the steps above, if you have a modified jinja template in use in the preprocessing script, save it offline as a .jinja file
- This file can be specified using the **template_file** parameter, an example template is provided in the **data/** directory, which the integration uses out of the box
- Export the current working state in Administrator Settings/ Organization/ Export
- Install the new version in a separate virtual python environment so you can easy revert

Common connection issues with TLS and Troubleshooting

```
fn-outbound-email:
  SMTP AUTH extension not supported by server.
  selftest: failure, Elapsed time: 0.416000 seconds
```

Email servers are often restrictive on which applications/users that are authorized to send emails, for instance if you have 2FA authentication enabled on a gmail account, you must add a specific application password or allow less secure apps (Not recommended)

<https://hotter.io/docs/email-accounts/app-password-gmail/>

<https://hotter.io/docs/email-accounts/secure-app-gmail/>

Occasionally, mailservers may indicate that emails have been sent successfully (including a successful note on the the associated incident) and yet they be blocked by the receiving mailserver due to insecure spam filters. This is a limitation of SMTP

authentication mechanism.

The port of TLS handshakes may also differ between mailservers (587/2525), a short history of port allocation can be found at: <https://pepipost.com/blog/25-465-587-2525-choose-the-right-smtp-port/>

More info on smtp protocol:

<https://pepipost.com/blog/what-is-smtp>

We cannot guarantee that all mailservers will work with this level of authentication/protocol and cannot support specific mailserver issues for that reason.

Uninstall

- SSH into your Integration Server.
- **Uninstall** the package:

```
$ pip uninstall fn-outbound-email
```

- Open the config file, scroll to the [fn_outbound_email] section and remove the section or prefix **#** to comment out the section.
- **Save** and **Close** the app.config file.

Troubleshooting

There are several ways to verify the successful operation of a function.

Resilient Action Status

- When viewing an incident, use the Actions menu to view **Action Status**.
- By default, pending and errors are displayed.
- Modify the filter for actions to also show Completed actions.
- Clicking on an action displays additional information on the progress made or what error occurred.

Resilient Scripting Log

- A separate log file is available to review scripting errors.
- This is useful when issues occur in the pre-processing or post-processing scripts.
- The default location for this log file is: **/var/log/resilient-scripting/resilient-scripting.log**.

Resilient Logs

- By default, Resilient logs are retained at **/usr/share/co3/logs**.
- The **client.log** may contain additional information regarding the execution of functions.

Resilient-Circuits

- The log is controlled in the **.resilient/app.config** file under the section [resilient] and the property **logdir**.
- The default file name is **app.log**.
- Each function will create progress information.
- Failures will show up as errors and may contain python trace statements.

Support

Name	Version	Author	Support URL
------	---------	--------	-------------

Name	Version	Author	Support URL
fn_outbound_email	1.1.0	Sean@IBM Resilient	https://www.ibm.com/security/intelligent-orchestration/resilient