

 README.md

# Microsoft Exchange Online Functions for IBM Resilient

- [Release Notes](#)
- [Overview](#)
- [Requirements](#)
- [Installation](#)
- [Uninstall](#)
- [Troubleshooting](#)
- [Support](#)

## Release Notes

### v1.0.0

- Initial Release

## Overview

Resilient Integration with Exchange Online provides the capability to access and manipulate Microsoft Exchange Online (Office 365 in the cloud) messages from the IBM Resilient Soar Platform. The integration uses Microsoft Graph API to access the data in Office 365. Included in the integrations are the following capabilities:

- Get the user profile of the specified email address in JSON format.
- Get a specified message and return the results in JSON format.
- Get a specified message in .eml format and write as an incident attachment.
- Move a message to a specified "Well-known" Outlook folder.
- Send an message: from the specified email address to the specified recipients with specified message subject and body text.
- Query messages of a single user, a list of users, or the whole tenant and return a list of messages matching the criteria: message sender, messages from a specific Well-known folder, a time frame for when the message was received, text contained in the message subject or the message body, whether the message has attachments. Results are returned in the Exchange Online Query Message Results data table.
- Delete a single specified message from a specified email address.
- Delete a list of messages that are the results of a message query. The messages deleted are written to the Exchange Online Query Messages data table.
- Create a meeting event in the organizer's Outlook calendar and send a calendar event message to meeting participants inviting them to the meeting.

The integration contains the following functions:

The screenshot shows the Resilient platform's navigation bar at the top with links for Dashboards, Inbox, Incidents, Create, Search, and User Profile. Below the navigation is a breadcrumb trail: Customization Settings > Functions. The main content area is titled "Functions" and features a search bar labeled "Search...". A table lists various Exchange Online functions, each with a description and a small icon. At the bottom right of the table is a blue button labeled "New Function".

Name	Description
Exchange Online: Create Meeting	This function creates a meeting event in the organizer's Outlook calendar and sends a calendar event mail message to the meeting participants inviting them to the meeting.
Exchange Online: Delete Message	Delete a message in the specified user's email address mailbox. The email address of the mailbox and the message id are required input parameters. The mail folder is an optional parameter.
Exchange Online: Delete Messages From Query Results	This Exchange Online function deletes a list of messages returned from the Query Message function. The input to the function is a string containing the JSON results from the Query Messages function.
Exchange Online: Get Message	This function returns the contents of an Exchange Online message in JSON format.
Exchange Online: Get User Profile	This function gets Exchange Online user profile for a given email address.
Exchange Online: Move Message to Folder	This function moves an Exchange Online message to the specified folder in the users mailbox.
Exchange Online: Query Messages	This function queries Exchange Online to find messages matching the specified input parameters. A list of messages is returned from the function.
Exchange Online: Send Message	This function creates a message and sends it to the specified recipients.
Exchange Online: Write Message as Attachment	This function gets the mime content of an Exchange Online message and writes it as an incident attachment.

## Requirements

- Resilient platform >= v34.2.47
- An Integration Server running:
  - resilient\_circuits>=31.0.0
  - resilient\_lib>=35.0.0
  - The minimum set of Resilient API permissions for this integration if using an API key account:
    - Edit Org Data
    - Incidents.Edit.Fields
    - Functions.Read
    - Functions.Edit
    - Layouts.Read
    - Other.ReadIncidentsActionInvocations
    - Scripts.Create
    - Scripts.Edit
    - Workflows.Create
    - Workflow.Edit
  - To set up an Integration Server see: [ibm.biz/res-int-server-guide](http://ibm.biz/res-int-server-guide)
- The following Microsoft Graph API "Application permissions" (See Microsoft Azure App Configuration section below to configure):
  - Calendar.ReadWrite
  - Mail.ReadWrite
  - Mail.Send
  - MailboxSetting.Read
  - User.Read.All

## Installation

---

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

```
$ unzip fn_exchange_online-x.x.x.zip
```

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

```
$ cd fn_exchange_online-x.x.x
```

- **Install** the package:

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

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

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

- Import the `fn_exchange_online customizations` into the Resilient platform:

```
$ resilient-circuits customize -y -l fn-exchange-online
```

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

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

Config	Required	Example	Description
<code>microsoft_graph_token_url</code>	Yes	<code>https://login.microsoftonline.com/{tenant}/oauth2/v2.0/token</code>	<i>Microsoft Graph URL endpoint for acquiring access token</i>
<code>microsoft_graph_url</code>	Yes	<code>https://graph.microsoft.com/v1.0</code>	*Microsoft Graph base URL *
<code>tenant_id</code>	Yes	xxx	<i>Microsoft Azure Tenant ID</i>
<code>client_id</code>	Yes	xxx	<i>Microsoft Azure Client ID (Application ID)</i>
<code>client_secret</code>	Yes	xxx	<i>Microsoft Azure Client Secret</i>
<code>max_messages</code>	Yes	100	<i>Maximum number of messages that a query will return</i>
<code>max_users</code>	Yes	2000	<i>Maximum number of users searched in a query</i>

- **Save** and **Close** the `app.config` file.

- [Optional]: Run selftest to test the Integration you configured:

```
$ resilient-circuits selftest -l fn-exchange-online
```

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

```
$ resilient-circuits run
```

## Custom Layouts

Create an Exchange Online custom incident tab and drag the Exchange Online Message Query Results data table on to the layout and click Save as shown in the screenshot below:

The screenshot shows the Resilient platform's customization interface. On the left, there's a sidebar with a tree view of incident tabs: New Incident Wizard, Incident Tabs (Manage Tabs, Summary Section, Tasks, Details, Breach, Notes, Members, News Feed, Attachments, Stats, Timeline, Artifacts), Exchange Online (selected), and Add Tab. The main area is titled "Incident: Exchange Online" and contains a "Fields" section with a search bar and a list of fields like Address, Alberta Health Risk Assessment, etc., and a "Data Tables" section with a list of tables like Exchange Online Message Query Results (which has a red arrow pointing to it) and Views. The "Exchange Online Message Query Results" table is currently selected.

The results of any Exchange Online message query are displayed in this data table on the Exchange Online custom incident tab.

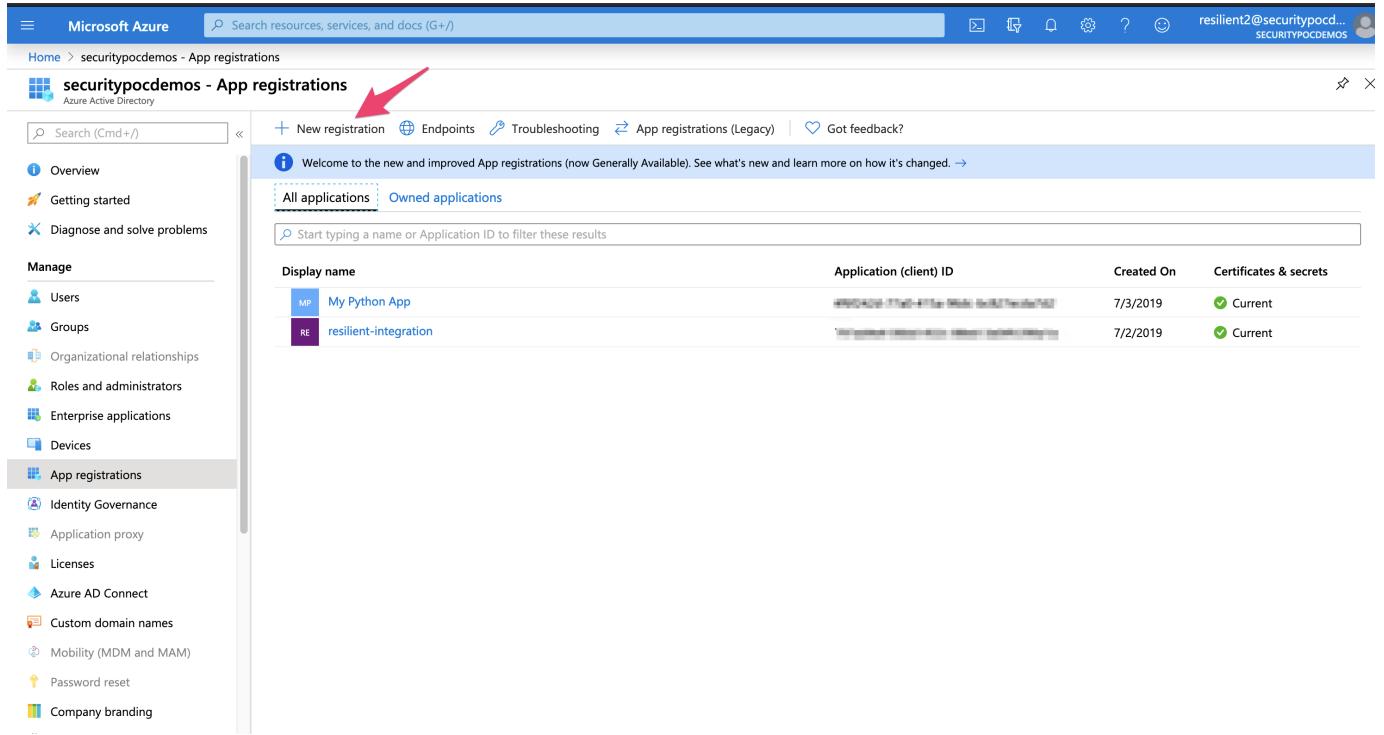
The screenshot shows the Exchange Online custom incident tab. At the top, there's a navigation bar with tabs: Tasks, Details, Breach, Notes, Members, News Feed, Attachments, Stats, Timeline, Artifacts, Email, and Exchange Online (selected). Below the navigation bar is a "Edit" button. The main area is titled "Exchange Online Message Query Results" and contains a data table with columns: Query Date, Received Date, Queried Email Address, Sender Email, Message Subject, Has Attachments, Web Link, Status, and Message ID. A search bar, Print, and Export buttons are at the top of the table. A message "There is no data for this table" is displayed. At the bottom, it says "Showing 0 to 0 of 0 entries".

## Microsoft Azure App Configuration

To run the Resilient Exchange Online integration, you must first register the application on Microsoft Azure portal. The tenant ID, client ID and the client secret that are defined in the fn\_exchange\_online section of the app.config are assigned by Azure when the application is registered.

## App Registration

To register the Resilient integration application click "App registrations" in Manage section of your Azure Active Directory domain account. Then click the "New Registration" button as depicted in the image below.



The screenshot shows the Microsoft Azure portal interface. The left sidebar is titled 'Manage' and includes sections for Users, Groups, Organizational relationships, Roles and administrators, Enterprise applications, Devices, App registrations (which is selected), Identity Governance, Application proxy, Licenses, Azure AD Connect, Custom domain names, Mobility (MDM and MAM), Password reset, and Company branding. The main content area is titled 'securitypocdemos - App registrations' under 'Azure Active Directory'. It shows a list of registered applications: 'My Python App' (client ID: 72f988bf-69fc-4270-9a4d-39f9eef32acc, created on 7/3/2019) and 'resilient-integration' (client ID: 12345678-1234-1234-1234-1234567890ab, created on 7/2/2019). At the top, there are tabs for 'All applications' (selected) and 'Owned applications'. Below the tabs is a search bar with placeholder text 'Start typing a name or Application ID to filter these results'. At the very top of the page, there is a navigation bar with a search bar, a user profile icon, and other global navigation links.

Enter a name for the integration. In this example, the name is "resilient-integration". Then press the "Register" button.

**Name**  
The user-facing display name for this application (this can be changed later).

**Supported account types**

Who can use this application or access this API?

Accounts in this organizational directory only (securitypocdemos only - Single tenant)

Accounts in any organizational directory (Any Azure AD directory - Multitenant)

Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

[Help me choose...](#)

**Redirect URI (optional)**  
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web

**By proceeding, you agree to the Microsoft Platform Policies** [View policies](#)

**Register**

Click on the newly created application. A page appears that is similar to the screenshot below. Get the tenant and client IDs for the application, which are parameters in the app.config file:

**resilient-integration**

**Overview**

Display name : **resilient-integration**      Supported account types : **My organization only**  
 Application (client) ID : **[REDACTED]**      Redirect URIs : [Add a Redirect URI](#)  
 Directory (tenant) ID : **[REDACTED]**      Application ID URI : [api://\[REDACTED\]](#)  
 Object ID : **[REDACTED]**      Managed application in ... : [resilient-integration](#)

**Call APIs**

Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

[View API permissions](#)

**Documentation**

[Microsoft identity platform](#)  
[Authentication scenarios](#)  
[Authentication libraries](#)  
[Code samples](#)  
[Microsoft Graph](#)  
[Glossary](#)  
[Help and Support](#)

Next, click on the left menu item, "Certificates & secrets" and create a secret, which is another application credential in the app.config.

The screenshot shows the Microsoft Azure portal with the URL [https://portal.azure.com/#blade/Microsoft\\_Azure\\_CertificatesSecrets/OverviewBlade/ResourceType/Certificates](https://portal.azure.com/#blade/Microsoft_Azure_CertificatesSecrets/OverviewBlade/ResourceType/Certificates). The left sidebar is open, showing various management options. The 'Certificates & secrets' option is selected and highlighted with a red arrow. The main content area shows a section for 'Certificates' with a button to 'Upload certificate'. Below it, a section for 'Client secrets' is shown with a table. A red arrow points to the '+ New client secret' button. Another red arrow points to the 'Value' column of the first row, which contains the value 'XbQ\*\*\*\*\*'. The top right corner shows the user's email 'resilient1@securitypocd... SECURITYPOCDEMONS'.

Thumbprint	Start Date	Expires

**Client secrets**

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

Description	Expires	Value
Password uploaded on Tue Jan 14 2020	12/31/2299	XbQ*****
Res-Integration	12/31/2299	j+3*****

## API Permissions

For the Resilient integration app to access data in Microsoft Graph, an administrator must grant it the correct permissions via a consent process. Click on "API permissions" on the left menu and then "+ Add a Permission".

The screenshot shows the Microsoft Azure portal with the URL [https://portal.azure.com/#blade/Microsoft\\_Azure\\_AppRegistration/AppDetailsBlade/ResourceType/APIPermissions](https://portal.azure.com/#blade/Microsoft_Azure_AppRegistration/AppDetailsBlade/ResourceType/APIPermissions). The left sidebar is open, showing various management options. The 'API permissions' option is selected and highlighted with a red arrow. The main content area shows a table of 'Configured permissions' for 'Microsoft Graph'. A red arrow points to the '+ Add a permission' button. Another red arrow points to the 'Status' column, which shows 'Granted for securitypoc...' for all listed permissions. The top right corner shows the user's email 'resilient1@securitypocd... SECURITYPOCDEMONS'.

API / Permissions name	Type	Description	Admin Consent Requir...	Status
Calendars.ReadWrite	Application	Read and write calendars in all mailboxes	Yes	Granted for securitypoc...
Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	Granted for securitypoc...
Mail.Send	Application	Send mail as any user	Yes	Granted for securitypoc...
MailboxSettings.Read	Application	Read all user mailbox settings	Yes	Granted for securitypoc...
User.Read.All	Application	Read all users' full profiles	Yes	Granted for securitypoc...

Click on Microsoft Graph:

The screenshot shows the Microsoft Azure portal interface. On the left, a sidebar menu includes options like Overview, Quickstart, Manage, API permissions (which is selected), Expose an API, Owners, Roles and administrators, Manifest, Support + Troubleshooting, and New support request. The main content area is titled "resilient-integration - API permissions". It shows a table of configured permissions for the Microsoft Graph API, which is expanded to show five specific permissions: Calendars.ReadWrite, Mail.ReadWrite, Mail.Send, MailboxSettings.Read, and User.Read.All. A red arrow points from the top right towards the Microsoft Graph section in the "Commonly used Microsoft APIs" grid.

API / Permissions name	Type
Calendars.ReadWrite	Application
Mail.ReadWrite	Application
Mail.Send	Application
MailboxSettings.Read	Application
User.Read.All	Application

### Request API permissions

Select an API

- Microsoft APIs
- APIs my organization uses
- My APIs

**Commonly used Microsoft APIs**

<b>Microsoft Graph</b> Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.	<b>Azure Rights Management Services</b> Allow validated users to read and write protected content	<b>Azure Service Management</b> Programmatic access to much of the functionality available through the Azure portal	<b>Data Export Service for Microsoft Dynamics 365</b> Export data from Microsoft Dynamics CRM organization to an external destination
<b>Dynamics 365 Business Central</b> Programmatic access to data and functionality in Dynamics 365 Business Central	<b>Dynamics CRM</b> Access the capabilities of CRM business software and ERP systems	<b>Flow Service</b> Embed flow templates and manage flows	<b>Intune</b> Programmatic access to Intune data
<b>Office 365 Management APIs</b> Retrieve information about user, admin, system, and policy actions and events from Office 365 and Azure AD activity	<b>OneNote</b> Create and manage notes, lists, pictures, files, and more in OneNote notebooks	<b>Power BI Service</b>	<b>SharePoint</b>
<b>Skype for Business</b>			

Select Application permissions (not Delegated permissions):

This screenshot shows the same Microsoft Azure portal interface as the previous one, but the "Application permissions" tab is selected in the sidebar. The main content area is identical to the first screenshot, showing the Microsoft Graph API permissions table. A red arrow points from the bottom right towards the "Application permissions" section in the "What type of permissions does your application require?" grid.

API / Permissions name	Type
Calendars.ReadWrite	Application
Mail.ReadWrite	Application
Mail.Send	Application
MailboxSettings.Read	Application
User.Read.All	Application

### Request API permissions

[All APIs](#)

**Microsoft Graph**  
<https://graph.microsoft.com/> [Docs](#)

What type of permissions does your application require?

<input checked="" type="radio"/> Delegated permissions Your application needs to access the API as the signed-in user.	<input type="radio"/> Application permissions Your application runs as a background service or daemon without a signed-in user.
---	--

Check each of the following Microsoft Graph API "Application permissions":

- Calendar.ReadWrite
- Mail.ReadWrite
- Mail.Send
- MailboxSetting.Read
- User.Read.All

The screenshot shows the 'Request API permissions' page in the Microsoft Azure portal. On the left, the navigation menu is visible with 'API permissions' selected. The main area displays 'Configured permissions' for the application 'resilient-integration'. Three checkboxes are highlighted with red arrows: 'MailboxSettings.Read' (checked), 'Mail.ReadWrite' (checked), and 'Mail.Send' (checked). These three permissions are highlighted with a gray rectangular box.

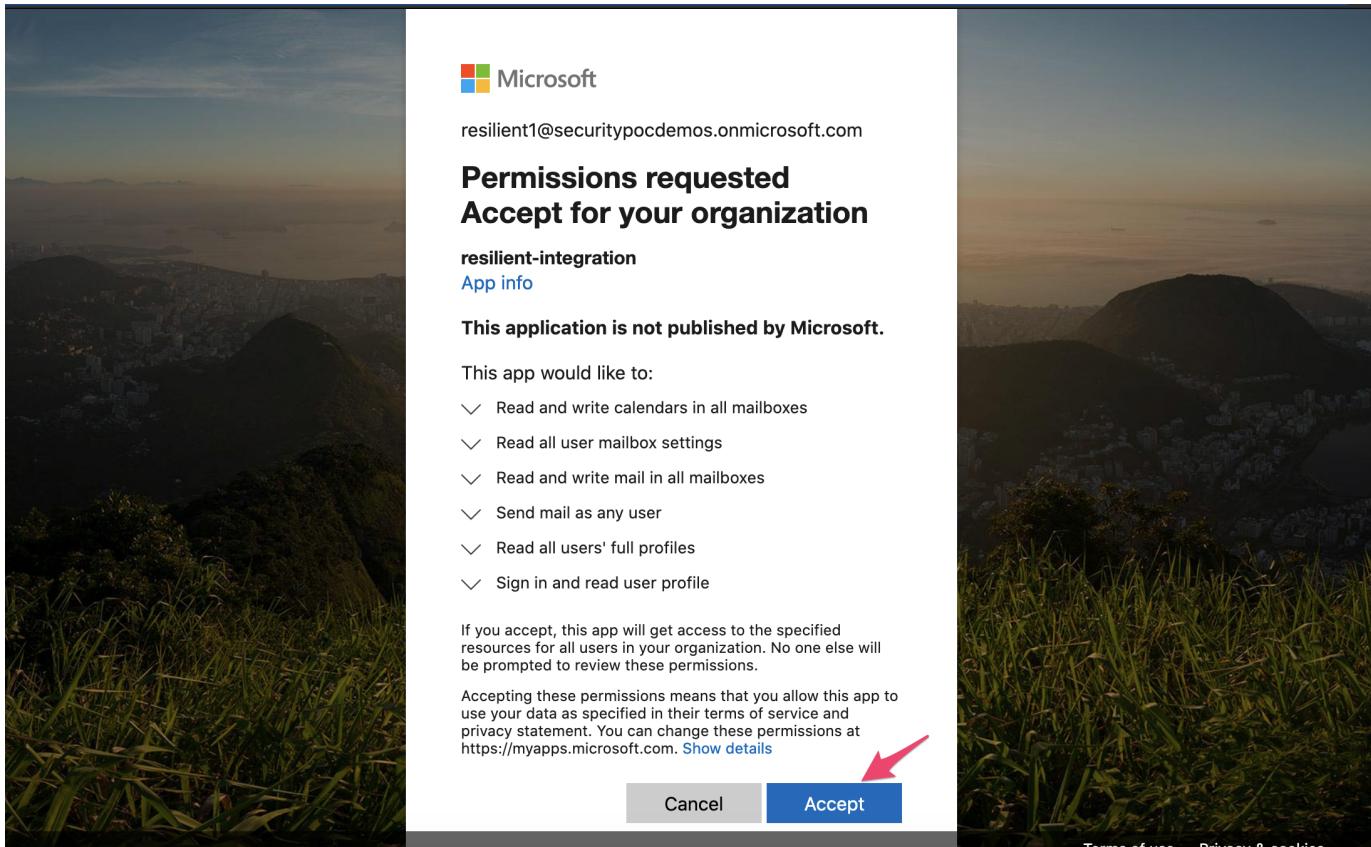
API / Permissions name	Type	Description	Admin Consent Req...	Status
Calendars.ReadWrite	Application	Read and write calendars in all mailboxes	Yes	<span>Granted for securitypoc...</span>
Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	<span>Granted for securitypoc...</span>
Mail.Send	Application	Send mail as any user	Yes	<span>Granted for securitypoc...</span>
MailboxSettings.Read	Application	Read all user mailbox settings	Yes	<span>Granted for securitypoc...</span>
User.Read.All	Application	Read all users' full profiles	Yes	<span>Granted for securitypoc...</span>

Once the API Application permissions are added, click the "Grant admin consent" button for your domain:

The screenshot shows the 'resilient-integration - API permissions' page in the Microsoft Azure portal. The 'Grant admin consent for securitypocdemos' button is highlighted with a red arrow. The table below lists the configured permissions with their status as 'Granted for securitypocdemos'.

API / Permissions name	Type	Description	Admin Consent Req...	Status
Calendars.ReadWrite	Application	Read and write calendars in all mailboxes	Yes	<span>Granted for securitypoc...</span>
Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	<span>Granted for securitypoc...</span>
Mail.Send	Application	Send mail as any user	Yes	<span>Granted for securitypoc...</span>
MailboxSettings.Read	Application	Read all user mailbox settings	Yes	<span>Granted for securitypoc...</span>
User.Read.All	Application	Read all users' full profiles	Yes	<span>Granted for securitypoc...</span>

You may need to log in to an admin account to accept the permissions requested on behalf of your organization:



## Uninstall

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

```
$ pip uninstall fn-exchange-online
```

- Open the config file, scroll to the [fn\_exchange\_online] 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
fn_exchange_online	1.0.0	IBM Resilient	<a href="https://ibm.com/mysupport">https://ibm.com/mysupport</a>