

Add New Incident Type

- This project lets you run a script that adds new functionality to the Resilient UI.
- This integration
 - A component for a “resilient circuits” framework
 - A fragment of a configuration file

Project Overview

The project runs as a standalone Python application. When this application starts, it connects to the Resilient platform and starts listening for actions. If the `add_new_incident` action is invoked then the application locates the script and runs it. When the script is run a new functionality is added to the Resilient UI. After running the script, the user will be able to create a new incident type and have it be saved to the incident from the New Incident page, rather than Customization Settings.

Installation

Install the project onto a Windows or Unix machine. This machine requires

- Python version 2.7.6 or later
- Network access to the Resilient appliance via ports 443 and 65001
- Resilient Systems application version 24 or later, with Action Module.

Python Libraries

You must install several Python libraries that are required by the application.

First, install the “co3” module, which is the Resilient Systems REST API client library for Python; and the “resilient_circuits” module, which is the Resilient action module application framework for Python. You must also install the `Add_Inc` library.

Follow the installation instructions here:

<https://github.com/Co3Systems/co3-api/tree/master/python>

Basic Configuration

First, edit the “app.config” file and edit the values needed to connect to your Resilient server. You only need to edit: hostname, email, password, & org values.

UI Configuration

To run the `auto_add_new_inc_types` script to Resilient you must configure the UI as follows:

In Resilient, open Customization Settings → Actions → Message Destinations.

Click on Message Destinations and copy the fields below except Users. Enter your API user account to the Users list for this message destination:

Edit Message Destination

	type	Queue
Name *		<input type="text" value="Add new object"/>
Programmatic Name *		<input type="text" value="add_new_object"/>
Expect Acknowledgement		<input type="text" value="Yes"/>
Users		<div><input type="text" value="Alexander Gamota (agamota@gmail.com)"/></div>



Cancel Save

Save

Next open → Action Fields. Enter the information as below. Tooltip and Placeholder don't have to be the same as below.

Editing Field

What type of field is this? ⓘ Text


What is the label for this field? * ⓘ

API Access Name * ⓘ

Placeholder ⓘ

Requirement ⓘ Optional

Tooltip ⓘ

CancelSave

Editing Field

What type of field is this? Text

What is the label for this field? *

Add new

Requirement

Optional

API Access Name *

add_new

Tooltip

Holds new incident type value

Placeholder ⓘ

Add New Incident Type



Cancel

Save

Next open → Automatic Actions. Enter the information as listed below.

Edit Automatic Action

Object Type

Incident

Display Name *

Add New

Destinations *

Add new object ✕

Conditions

Add Condition

Incident Type ▼

contains ▼

Other ✕ Remove

✕

Cancel

Save

Edit Automatic Action

Object Type

Incident

Display Name *

Add New

Destinations *

Add new object ✕

Conditions

Add Condition

Incident Type ▾

contains ▾

Other ✕

Remove



Cancel

Save

Running the Application

Follow the configuration instructions from Basic Configuration to set appropriate values into the “app.config” file.

Add the auto_add_new_inc_type.py file to the components folder.

Access the shell runner directory.

Then append the app.config.fragment to the app.config file by using:

```
cat app.config.fragment >> app.config
```

Then, start the application, using:

```
python run.py
```

You might be asked to enter your email and server:

```
python run.py --email example@site.com --host hostname
```

If ran successfully you should expect the following code:

```
INFO app.py Configuration file is app.config
INFO connectionpool.py Starting new HTTPS connection (1): resilient
INFO app.py App Started
INFO actions_component.py STOMP connected
INFO actions_component.py Component <Shell/actions.shell 51251:MainThread
(queued=0) [S]> registered to actions.shell
INFO actions_component.py Subscribe to 'shell'
INFO component_loader.py Loaded component 'auto_add_new_inc_type.py'
INFO app.py Components loaded
```

On Windows, the application can be configured to run as a service; instructions are not included here, but are available on request.

Configuration

The default behavior above only requires that you

- Configure the action, using the administration UI. (Listed under UI configuration)
- Deploy the auto_add_new_inc_type.py file to the components file of the repository.

The “[framework_section]” section of the config file has:

```
# -----
# The 'framework_section' action component
# -----
[framework_section]
queue=add_new_object
```

This creates the default behavior described above:

- Listens to actions on the message destination (queue) named “add_new_object”
- For that action, the script “auto_add_new_inc_type” is executed, passing the artifact value as a command- line parameter

Examples:

*When other is selected the Enter New Incident Type Window opens prompting entry.

Layouts
Actions
Phases & Tasks
Incident Types
Breach

Incident Types

Filters
☐ Display Hidden

Malware

Edit

Not an Issue

Edit

Other

Edit

Phishing

Edit

Stolen documents / files / records

Edit

Stolen PC / laptop / tablet

Edit

Stolen PDA / smartphone

Edit

Stolen storage device / media

Edit

System (OS) account compromises

Edit

Delete

System Intrusion

Edit

TBD / Unknown

Edit

Vendor / 3rd party error

Edit

+ Add Type

Incidents can be organized under various types. You can also create [automatic tasks](#), which will get created when incidents are assigned to these types.

Grey types are system default types, and can not be removed, only hidden.

🔒 indicates that a type is hidden, and will not be a selectable type when creating a new incident. Typically these hidden types are used for tree organizational purposes.

Example

Actions ▾

Summary

ID 5450
Phase Engage
Severity Low
Date Created 07/27/2016
Date Occurr... —
Date Discov... 07/27/2016
Data Compr... Unknown

Incident Type

Malware

Other

System (OS) account compromises

People

Created By [Alexander Gamota](#)
Owner [Alexander Gamota](#)
Members *There are no members.*

Related Incidents

No related incidents.

Description

No description.

Tasks	Details	Breach	Notes	Members	News Feed	Attachments	Stats	Timeline	Artifacts	
-------	---------	--------	-------	---------	-----------	-------------	-------	----------	-----------	--

Details

Edit

Basic Details

Name ⓘ Example
Twenty plus Supercalafragalisticexpialodocious
Description ⓘ —
Incident Type ⓘ

Malware

Other

System (OS) account compromises

NIST Attack Vectors ⓘ —
Incident Disposition ⓘ Confirmed
Phase ⓘ Engage
Resolution ⓘ —
Resolution Summary ⓘ —
Owner [Alexander Gamota](#)
Created By [Alexander Gamota](#)

*The 'System (OS) account compromises' type is saved under the incidents Incident Type field.