URLScan.io Function

https://urlscan.io is a service to scan and analyse websites. When a URL is submitted to urlscan.io, an automated process will browse to the URL like a regular user and record the activity that this page navigation creates. This includes the domains and IPs contacted, the resources (JavaScript, CSS, etc) requested from those domains, as well as additional information about the page itself. urlscan.io will take a screenshot of the page, record the DOM content, JavaScript global variables, cookies created by the page, and a myriad of other observations.

This integration is a Resilient function that can be called from workflows, to submit a URL for analysis by urlscan.io. It returns the report metadata, report URL, and base64-encoded screenshot that is attached to the incident.

Installation

To install in "development mode"

```
pip install -e ./fn_urlscanio/
```

After installation, the package will be loaded by `resilient-circuits run`.

To uninstall,

```
pip uninstall fn_urlscanio
```

To package for distribution,

```
python ./fn_urlscanio/setup.py sdist
```

The resulting .tar.gz file can be installed using

```
pip install <filename>.tar.gz
```

After installation, before running, you must import the customizations into your Resilient platform,

```
resilient-circuits customize
```

app.config settings

The following block is automatically added to your app.config file when running `resilient-circuits config -u`. You will need to add your API key and have the flexibility to adjust the URL parameters if required.

```
[urlscanio]
# API key for urlscan.io
urlscanio_api_key=
# Base URL for the urlscanio API
urlscanio_report_url=https://urlscan.io/api/v1
# Base URL to access screenshots in urlscanio
urlscanio_screenshot_url=https://urlscan.io/screenshots
# Optional timeout (seconds)
# timeout=300
```

Pre-Processing Script

```
# This is an artifact workflow;
# The URL to scan is the artifact value
inputs.urlscanio_url = artifact.value

# Set the incident id
inputs.incident_id = incident.id
```

Post-Processing Script

No action is performed after the workflow is complete, so we simply outline the result format in the results for ease of use.

```
# The result contains,
# {
#   "png_url": the URL of the screenshot image
#   "png_base64content": the base64-encoded screenshot (PNG)
#   "report_url": the URL of the JSON report_url
#   "report": the JSON report, which will contain lots of detail of the
page analysis (see urlscan.io for details).
# }
#
# In this case, the file is already attached to the incident.  Nothing to
do here.
```

Other notes

To regenerate the customization blob, `resilient-circuits codegen -p fn_urlscanio -m urlscanio --workflow example_urlscanio --rule "Example: urlscan.io"`

Changelog

1.0.0

- Initial Release

1.1.0

- Removed worklfow dependency on fn_utilities
- Added incident_id parameter to workflow inputs