Proofpoint TRAP Functions for IBM Resilient

- Key Features
- Poller
- Function Proofpoint TRAP: Get Incident Details
- Function Proofpoint TRAP: Get List Members
- Function Proofpoint TRAP: Add Members to List
- Function Proofpoint TRAP: Update List Member
- Function Proofpoint TRAP: Delete List Member
- Data Table Proofpoint TRAP Events
- Data Table Proofpoint TRAP List Members
- Custom Fields
- Rules

Key Features

Resilient integration for Proofpoint TRAP

The ProofPoint Trap function package provides the following features:

- Poll a Proofpoint TRAP server for incidents and create corresponding incidents in Resilient.
- Get Proofpoint TRAP incident details.
- Get a Proofpoint Trap list member or members.
- Add a member to a Proofpoint Trap list for artifacts of type host, IP address, or URL to a list.
- Update a member of a Proofpoint Trap list.
- Delete a member from of a Proofpoint Trap list.

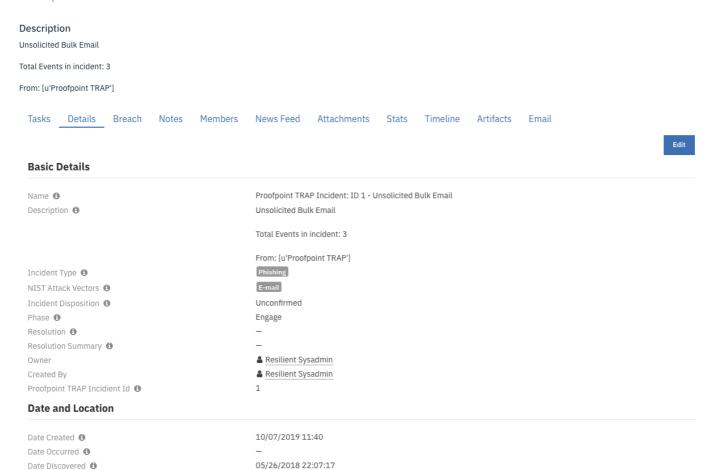
Poller:

Threaded Poller which runs continuously while the integration is running.

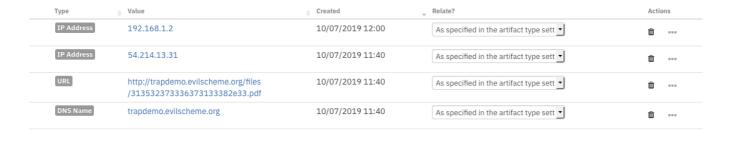
- Polls a Proofpoint TRAP server for incidents and create corresponding incidents in Resilient.
- Adds Proofpoint Trap events to incident data table Proofpoint TRAP Events in Resilient.
- Adds artifacts to incidents in Resilient corresponding to hosts artifacts in Proofpoint TRAP incident events. The actual
 artifacts added are determined by the host_categories configuration option.
- Adds a note with Proofpoint TRAP events details to an incident in Resilient.

Example incident created by the poller:

Proofpoint TRAP Incident: ID 1 - Unsolicited Bulk Email



Examples of incident artifacts created by the poller:



Functions:

Function - Proofpoint TRAP: Get Incident Details

Fetch Incident Details from Proofpoint Trap

- Adds a note to the Resilient incident with ProofPoint Trap incident details.
- An example workflow which uses this Resilient Function includes Example: Proofpoint Trap: Get Incident Details.

The workflow is initiated by the incident rule, Example: Proofpoint Trap: Get Incident Details.

1. Open an incident and select Example: Proofpoint Trap: Get Incident Details from Actions.

```
Example: Proofpoint TRAP: Get Incident Details
Example: Proofpoint TRAP: Get List Members

Action Status
Workflow Status
Close Incident
Delete Incident
```

- 2. Click Action-> Example: Proofpoint Trap: Get Incident Details.
- 3. A note is added to the incident with the indicident details in JSON format.

This invokes the Example: Proofpoint Trap: Get Incident Details workflow, which calls the Proofpoint TRAP: Get Incident Details function.

► Inputs:

Config	Type	Required	Example	Description
trap_incident_id	number	Yes	1	Proofpoint TRAP Incident ID

► Outputs:

```
results = {
    # TODO: Copy and paste an example of the Function Output within this code block.
    # To see view the output of a Function, run resilient-circuits in DEBUG mode and invoke the Function.
    # The Function results will be printed in the logs: "resilient-circuits run -- loglevel=DEBUG"
}
```

► Example Pre-Process Script:

```
inputs.trap_incident_id = incident.properties.proofpoint_trap_incident_id
```

► Example Post-Process Script:

```
note = "{}".format(unicode(results.data))
incident.addNote(helper.createRichText(note))
```

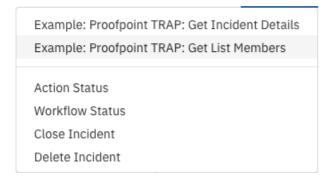
Proofpoint TRAP: Get List Members

Get member or members of a Proofpoint Trap list.

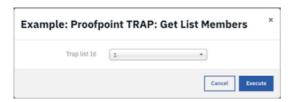
- Adds Proofpoint Trap list members to incident datatable Proofpoint TRAP List Members in Resilient.
- Example workflows which use this Resilient Function include Example: Proofpoint TRAP: Get List Members.

The workflow is initiated by the incident rule, Example: Proofpoint TRAP: Get List Members.

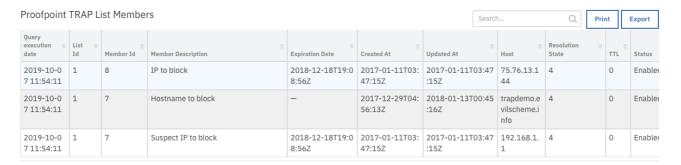
1. Open an incident and select Example: Proofpoint TRAP: Get List Members from Actions.



2. Select a list id from the displayed drop-down list and click Execute.



- 3. This invokes the Example: Proofpoint TRAP: Get List Members workflow, which calls the Proofpoint TRAP: Get List Members function.
- 4. The datatable Proofpoint TRAP List Members will be updated in Resilient with the member details for the selected list.



► Inputs:

Config	Type	Required	Example	Description
trap_list_id	number	number	1	Proofpoint TRAP List ID
trap_member_id	number	text	1	Proofpoint TRAP List member ID.
trap_members_type	number	text	members.json	The Proofpoint TRAP information format to get in result for list membership. Default is members, ison.

► Outputs:

```
u'response_id': None, u'expiration': u'2018-12-18T19:08:56Z',
u'list_id': 2, u'host_id': 20,
                       u'hash_reputation_id': None, u'id': 8, u'reverse_user_id':
None
                      {u'user_id': None, u'description': u'test', u'deleted': False,
u'created_at': u'2017-01-11T03:43:54Z',
                       u'enabled': True, u'updated_at': u'2017-01-11T03:43:54Z',
                       u'host': {u'created_at': u'2016-12-29T04:56:13Z',
u'updated_at': u'2017-01-13T00:45:16Z',
                                u'host': u'string', u'ttl': 0, u'resolution_state':
4, u'id': 6
                       u'response_id': None, u'expiration': None, u'list_id': 2,
u'host_id': 6, u'hash_reputation_id': None,
                       u'id': 6, u'reverse_user_id': None
          'raw': '[{"user_id": null, "description": "IP to block", "deleted": false,
"created_at": "2017-01-11T03:47:15Z", '
                 '"enabled": true, "updated_at": "2017-01-11T03:47:15Z", "host":
{"created_at": "2017-01-11T03:47:15Z", '
                 '"updated_at": "2017-01-11T03:47:15Z", "host": "75.76.13.144",
"ttl": 0, "resolution_state": 4, "id": 20}, '
                 '"response_id": null, "expiration": "2018-12-18T19:08:56Z",
"list_id": 2, "host_id": 20, '
                 '"hash_reputation_id": null, "id": 8, "reverse_user_id": null},
{"user_id": null, "description": "test", '
                 '"deleted": false, "created_at": "2017-01-11T03:43:54Z", "enabled":
true, "updated_at": "2017-01-11T03:43:54Z", '
                  '"host": {"created_at": "2016-12-29T04:56:13Z", "updated_at": "2017-
01-13T00:45:16Z", "host": "string", '
                 '"ttl": 0, "resolution_state": 4, "id": 6}, "response_id": null,
"expiration": null, "list_id": 2, '
                 '"host_id": 6, "hash_reputation_id": null, "id": 6,
"reverse user id": null}]',
          'reason': None,
          'version': '1.0'
}
```

► Example Pre-Process Script:

```
inputs.trap_list_id = rule.properties.trap_list_id
inputs.trap_member_id = rule.properties.trap_member_id
```

```
## ProofPoint Trap - fn_proofpoint_trap_get_list_member ##
# Globals
# List of fields in datatable fn_proofpoint_trap_get_list_members script
DATA_TBL_FIELDS = ["member_list_id", "member_id", "member_description", "expiration",
"created_at", "updated_at"]
DATA_TBL_FIELDS_HOST = ["created_at", "host", "resolution_state", "ttl"]
FN_NAME = "fn_proofpoint_trap_get_list_member"
WF_NAME = "Example: Proofpoint TRAP: Get List Member"
MEMBER = results.content
```

```
INPUTS = results.inputs
QUERY_EXECUTION_DATE = results["metrics"]["timestamp"]
# Processing
def main():
    note_text = ''
    if MEMBER is not None:
        note_text = "ProofPoint Trap Integration Integration: Workflow <b>{0}</b>:
There was a result returned for Resilient function " \
                   "<b>{1}</b>".format(WF_NAME, FN_NAME)
        newrow = incident.addRow("trap_list_members")
        newrow.query_execution_date = QUERY_EXECUTION_DATE
        for f in DATA_TBL_FIELDS:
            f_base = '_'.join(f.split('_')[1:])
            if not f_base:
                f_base = f
            if f == "query_execution_time":
                continue
            if MEMBER[f_base] is not None:
                  newrow[f] = MEMBER[f base]
        host = MEMBER["host"]
        if host is not None:
            for d in DATA_TBL_FIELDS_HOST:
                newrow[d] = host[d]
    else:
        noteText += "ProofPoint Trap Integration: Workflow <b>{0}</b>: There were
<b>no</b> results returned for " \
                    "list id <b>{1}</b> for Resilient function <b>{3}
</b>".format(WF_NAME, INPUTS["trap_list_id"], FN_NAME)
    incident.addNote(helper.createRichText(note text))
if __name__ == "__main__":
    main()
```

Function - Proofpoint TRAP: Add Members to List

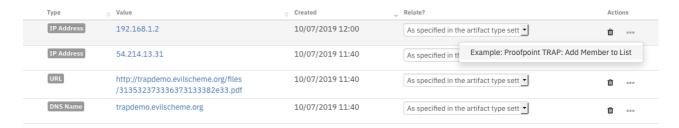
Add host, IP address, or URL to a Proofpoint TRAP list. List can be host list, url list, user list, or file list.

- Adds a new member to a Proofpoint TRAP list.
- Adds the new Proofpoint TRAP list member details to the incident datatable Proofpoint TRAP List Members in Resilient.
- Example workflows which use this Resilient Function includes Example: Proofpoint TRAP: Add Member to List.
- The workflow is applicable for Resilient artifact type: 'URL', 'User Account', 'DNS Name', 'IP Address', System Name', 'Malware SHA-256 Hash', 'Malware SHA-1 Hash' and 'Malware MD5 Hash'.

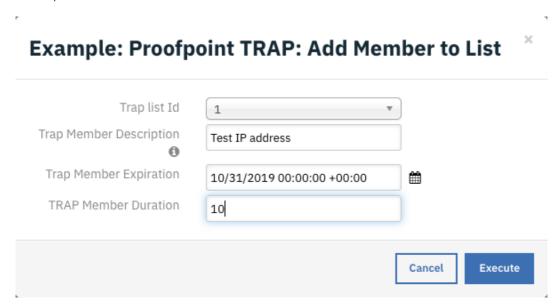
The workflow is initiated by the incident rule, Example: Proofpoint TRAP: Add Member to List.

1. Open an incident and select the 'Artifacts' tab.

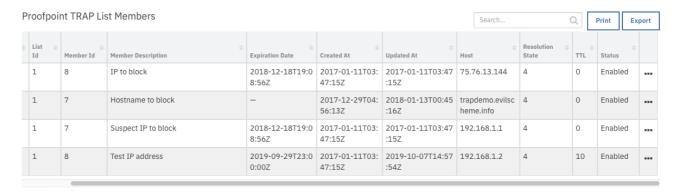
2. For a Resilient artifact of an applicable type e.g. 'IP Address' click Action-> Example: Proofpoint TRAP: Add Member to List.



3. The user is presented with a selection of user defined inputs. In the example, Proofpoint TRAP list id 1 is selected from the drop-down.



- 4. The user inputs values for the remainder of the fields and clicks Execute.
- 5. This invokes the Example: Proofpoint TRAP: Add Member to List workflow, which calls the Proofpoint TRAP: Add Members to List function.
- 6. The datatable **Proofpoint TRAP List Members** will be updated in Resilient with the member details for the selected list.



► Inputs:

Name	Туре	Required	Example	Tooltip
trap_description	text	No	_	Short description of Proofpoint TRAP list member.

Name	Туре	Required	Example	Tooltip
trap_duration	number	No	-	# of milliseconds after which to expire Proofpoint TRAP list membership.
trap_expiration	datetimepicker	No	-	Timestamp to expire Proofpoint TRAP list member.
trap_list_id	number	No	_	Proofpoint TRAP List ID.
trap_member	text	No	_	Proofpoint TRAP List member to add. Can be of type host, IP address, or URL.

► Outputs:

```
results = {
    # TODO: Copy and paste an example of the Function Output within this code block.
    # To see view the output of a Function, run resilient-circuits in DEBUG mode and invoke the Function.
    # The Function results will be printed in the logs: "resilient-circuits run -- loglevel=DEBUG"
}
```

► Example Pre-Process Script:

```
inputs.trap_member = artifact.value
inputs.trap_description = rule.properties.trap_description
inputs.trap_list_id = rule.properties.trap_list_id
inputs.trap_expiration = rule.properties.trap_expiration
inputs.trap_duration = rule.properties.trap_duration
```

```
## ProofPoint Trap - fn proofpoint trap add members to list ##
# Example result:
Result: {'inputs': {u'trap_list_id': 1, u'trap_member': u'75.76.13.144',
u'trap_description': u'A test',
                    u'trap_expiration': 1567526694000, u'trap_duration': 10},
         'metrics': {'package': 'fn-proofpoint-trap', 'timestamp': '2019-09-03
17:05:06', 'package_version': '1.0.0',
                     'host': 'myhost.ibm.com', 'version': '1.0', 'execution time ms':
3512},
         'success': True,
         'content': {u'user_id': None, u'description': u'IP to block', u'deleted':
False,
                     u'created_at': u'2017-01-11T03:47:15Z', u'enabled': True,
u'updated_at': u'2017-01-11T03:47:15Z',
                     u'host': {u'created_at': u'2017-01-11T03:47:15Z', u'updated_at':
u'2017-01-11T03:47:15Z',
                               u'host': u'75.76.13.144', u'ttl': 0,
u'resolution state': 4, u'id': 20
                     u'response_id': None, u'expiration': u'2018-12-18T19:08:56Z',
u'list_id': 1, u'host_id': 20,
```

```
u'hash_reputation_id': None, u'id': 8, u'reverse_user_id':
None},
         'raw': '{"user id": null, "description": "IP to block", "deleted": false,
"created at": "2017-01-11T03:47:15Z", '
                ""enabled": true, "updated_at": "2017-01-11T03:47:15Z", "host":
{"created_at": "2017-01-11T03:47:15Z", '
                '"updated_at": "2017-01-11T03:47:15Z", "host": "75.76.13.144", "ttl":
0, "resolution_state": 4, "id": 20}, '
                '"response_id": null, "expiration": "2018-12-18T19:08:56Z",
"list_id": 1, "host_id": 20, '
                '"hash_reputation_id": null, "id": 8, "reverse_user_id": null}',
         'reason': None,
         'version': '1.0'
}
0.00
# Globals
# List of fields in datatable fn_proofpoint_trap_add_members_to_list script
DATA_TBL_FIELDS = ["query_execution_time", "member_id", "list_id",
"member_description", "expiration", "created_at", "status"]
DATA_TBL_FIELDS_HOST = ["created_at", "updated_at", "host", "resolution_state",
"ttl"l
FN NAME = "fn proofpoint trap add members to list"
WF_NAME = "Example: Proofpoint TRAP: Get List Members"
MEMBER = results.content
INPUTS = results.inputs
QUERY_EXECUTION_DATE = results["metrics"]["timestamp"]
# Processing
def main():
    note_text = ''
    if MEMBER is not None:
        note_text = "ProofPoint Trap Integration Integration: Workflow <b>{0}</b>:
There was a valid result returned for Resilient function " \
                   "<b>{1}</b>".format(WF NAME, FN NAME)
        newrow = incident.addRow("trap list members")
        newrow.query_execution_date = QUERY_EXECUTION_DATE
        newrow.member_id = MEMBER["id"]
        newrow.member_description = MEMBER["description"]
        for f in DATA_TBL_FIELDS:
            if f in ["query_execution_time", "member_id", "member_description",
"status"]:
                continue
            if MEMBER[f] is not None:
                  newrow[f] = MEMBER[f]
        if MEMBER["enabled"]:
            newrow.status = "Enabled"
        host = MEMBER["host"]
        if host is not None:
            for d in DATA TBL FIELDS HOST:
                newrow[d] = host[d]
    else:
        noteText += "ProofPoint Trap Integration: Workflow <b>{0}</b>: There were
<b>no</b> results returned for " \
                    "list id <b>{1}</b> for Resilient function <b>{3}
</b>".format(WF NAME, INPUTS["trap list id"], FN NAME)
```

```
incident.addNote(helper.createRichText(note_text))

if __name__ == "__main__":
    main()
```

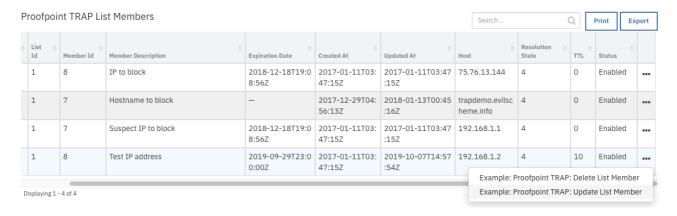
Function - Proofpoint TRAP: Update List Member

Update the member of a list by specifying the list and the member to update.

- Updates an existing member of a Proofpoint TRAP list.
- Refreshes Proofpoint TRAP list member details for the incident datatable Proofpoint TRAP List Members in Resilient
- Example workflows which use this Resilient Function includes Example: Proofpoint TRAP: Update List Member.
- The workflow is applicable for incident datatable Proofpoint TRAP List Members populated rows in Resilient.

The workflow is initiated by the datatable rule, Example: Proofpoint TRAP: Update List Member.

- 1. Open an incident and select the row of datatable **Proofpoint TRAP List Members** corresponding to the list member to update.
- 2. From the selected row actions select-> Example: Proofpoint TRAP: Update List Member.



3. The user is presented with a selection of user defined inputs.

Trap Member Description Trap Member Expiration TRAP Member Duration TRAP Member Duration

- 4. The user inputs values for the the displayed fields and clicks Execute.
- 5. This invokes the Example: Proofpoint TRAP: Update List Member workflow, which calls the Proofpoint TRAP: Update List Member function.

6. The datatable **Proofpoint TRAP List Members** will be refreshed in Resilient with the updated member details for the selected list.

Proofpoint TRAP List Members Search Q Print								Ex	port	
Query execution a	List \Leftrightarrow Id	(a) (b) (b) (c) (d) (d) (d) (d) (d) (d) (d) (d) (d) (d	Member Description	Expiration Date	Created At	Updated At	+ Host	Resolution \Leftrightarrow	TTL ©	Statu
2019-10-0 7 11:54:11	1	8	IP to block	2018-12-18T19:0 8:56Z	2017-01-11T03: 47:15Z	2017-01-11T03:47 :15Z	75.76.13.144	4	0	Enab
2019-10-0 7 11:54:11	1	7	Hostname to block	_	2017-12-29T04: 56:13Z	2018-01-13T00:45 :16Z	trapdemo.evilsc heme.info	4	0	Enab
2019-10-0 7 11:54:11	1	7	Suspect IP to block	2018-12-18T19:0 8:56Z	2017-01-11T03: 47:15Z	2017-01-11T03:47 :15Z	192.168.1.1	4	0	Enab
2019-10-0 7 17:39:15	1	8	Updated IP address	2019-09-30T23:0 0:00Z	2017-01-11T03: 47:15Z	2019-10-07T15:39 :15Z	75.76.13.144	4	5	Enab

► Inputs:

Name	Туре	Required	Example	Tooltip
trap_description	text	No	-	Short description of Proofpoint TRAP list member.
trap_duration	number	No	-	# of milliseconds after which to expire Proofpoint TRAP list membership.
trap_expiration	datetimepicker	No	_	Timestamp to expire Proofpoint TRAP list member.
trap_list_id	number	No	_	Proofpoint TRAP List ID.
trap_member_id	number	No	_	Proofpoint TRAP List member ID.

► Outputs:

```
results = {
    # TODO: Copy and paste an example of the Function Output within this code block.
    # To see view the output of a Function, run resilient-circuits in DEBUG mode and invoke the Function.
    # The Function results will be printed in the logs: "resilient-circuits run --- loglevel=DEBUG"
}
```

► Example Pre-Process Script:

```
inputs.trap_list_id = row.list_id
inputs.trap_member_id = row.member_id
inputs.trap_description = rule.properties.trap_description
inputs.trap_expiration = rule.properties.trap_expiration
inputs.trap_duration = rule.properties.trap_duration
```

```
## ProofPoint Trap - fn_proofpoint_trap_update_list_member ##
# Example result:
```

```
Results {'inputs': {u'trap_list_id': 2, u'trap_expiration': 1567527591000,
u'trap_description': u'Test',
                    u'trap_member_id': 8, u'trap_duration': 10},
         'metrics': {'package': 'fn-proofpoint-trap', 'timestamp': '2019-09-03
17:19:57', 'package_version': '1.0.0',
                     'host': 'myhost.ibm.com', 'version': '1.0', 'execution_time_ms':
33
                     }, 'success': True,
         'content': {u'user_id': None, u'description': u'Updated IP to block',
u'deleted': False,
                     u'created at': u'2017-01-11T03:47:15Z', u'enabled': True,
u'updated_at': u'2017-01-11T03:47:15Z',
                     u'host': {u'created_at': u'2017-01-11T03:47:15Z', u'updated_at':
u'2017-01-11T03:47:15Z',
                               u'host': u'75.76.13.144', u'ttl': 0,
u'resolution state': 4, u'id': 20
                     u'response_id': None, u'expiration': u'2018-12-18T19:08:56Z',
u'list_id': 2, u'host_id': 20,
                     u'hash_reputation_id': None, u'id': 8, u'reverse_user_id': None
         'raw': '{"user id": null, "description": "Updated IP to block", "deleted":
false, "created_at": "2017-01-11T03:47:15Z", '
                 ""enabled": true, "updated_at": "2017-01-11T03:47:15Z", "host":
{"created_at": "2017-01-11T03:47:15Z", '
                 ""updated_at": "2017-01-11T03:47:15Z", "host": "75.76.13.144", "ttl":
0, "resolution_state": 4, "id": 20}, '
                '"response id": null, "expiration": "2018-12-18T19:08:56Z",
"list_id": 2, "host_id": 20, "hash_reputation_id": null, '
                '"id": 8, "reverse user id": null}',
         'reason': None,
         'version': '1.0'
}
\mathbf{n} \mathbf{n} \mathbf{n}
# Globals
# List of fields in datatable "Proofpoint TRAP List Members" for
fn_proofpoint_trap_get_list_members script
DATA_TBL_FIELDS = ["query_execution_time", "member_id", "list_id",
"member_description", "expiration", "created_at", "status"]
DATA_TBL_FIELDS_HOST = ["created_at", "updated_at", "host", "resolution_state",
"ttl"]
FN_NAME = "fn_proofpoint_trap_update_list_member"
WF NAME = "Example: Proofpoint TRAP: Update List Member"
MEMBER = results.content
QUERY_EXECUTION_DATE = results["metrics"]["timestamp"]
# Processing
def main():
    note text = ''
    if MEMBER is not None:
        note_text = "ProofPoint Trap Integration Integration: Workflow <b>{0}</b>:
There was a valid result returned for Resilient function " \
                   "<b>{1}</b>".format(WF_NAME, FN_NAME)
        row.query_execution_date = QUERY_EXECUTION_DATE
        row.member_id = MEMBER["id"]
        row.member description = MEMBER["description"]
        for f in DATA_TBL_FIELDS:
```

```
if f in ["query_execution_time", "member_id", "member_description",
"status"]:
                continue
            if MEMBER[f] is not None:
                  row[f] = MEMBER[f]
        if MEMBER["enabled"]:
            row.status = "Enabled"
        host = MEMBER["host"]
        if host is not None:
            for d in DATA_TBL_FIELDS_HOST:
                row[d] = host[d]
    else:
        noteText += "ProofPoint Trap Integration: Workflow <b>{0}</b>: There were
<b>no</b> results returned for " \
                    "list id <b>{1}</b> for Resilient function <b>{3}
</b>".format(WF_NAME, INPUTS["trap_list_id"], FN_NAME)
    incident.addNote(helper.createRichText(note_text))
if __name__ == "__main__":
    main()
```

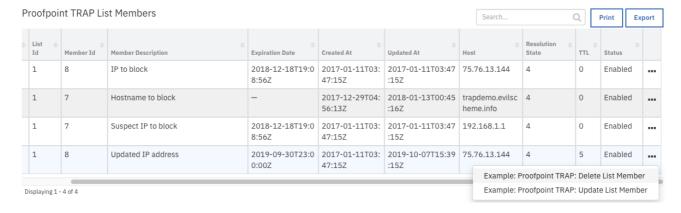
Function - Proofpoint TRAP: Delete List Member

Delete the member of a list.

- Deletes an member of a Proofpoint TRAP list.
- Updates the deleted Proofpoint TRAP list member details for the selected incident datatable Proofpoint TRAP List Members row in Resilient.
- Example workflows which use this Resilient Function includes Example: Proofpoint TRAP: Delete List
- The workflow is applicable for incident datatable Proofpoint TRAP List Members populated rows in Resilient.

The workflow is initiated by the datatable rule, Example: Proofpoint TRAP: Delete List Member.

- 1. Open an incident and select the row of datatable **Proofpoint TRAP List Members** corresponding to the list member to delete.
- 2. From the selected row actions select-> Example: Proofpoint TRAP: Delete List Member.



- 3. Click on the action to execute.
- 4. This invokes the Example: Proofpoint TRAP: Delete List Member workflow, which calls the Proofpoint TRAP: Delete List Member function.

5. The datatable Proofpoint TRAP List Members will be refreshed in Resilient for the deleted member with Status set to Deleted.

Ρ	roofpoir	nt TRAP Lis	st Members				Search	(Q	Print E	Export
40	List \oplus	() Member Id	Member Description	Expiration Date	© Created At	Updated At	Host	Resolution =	TTL 0	Status	
	1	8	IP to block	2018-12-18T19:0 8:56Z	2017-01-11T03: 47:15Z	2017-01-11T03:47 :15Z	75.76.13.144	4	0	Enabled	•••
	1	7	Hostname to block	-	2017-12-29T04: 56:13Z	2018-01-13T00:45 :16Z	trapdemo.evilsc heme.info	4	0	Enabled	•••
	1	7	Suspect IP to block	2018-12-18T19:0 8:56Z	2017-01-11T03: 47:15Z	2017-01-11T03:47 :15Z	192.168.1.1	4	0	Enabled	•••
	1	8	Updated IP address	2019-09-30T23:0 0:00Z	2017-01-11T03: 47:15Z	2019-10-07T15:39 :15Z	75.76.13.144	4	5	Deleted	•••

► Inputs:

Name	Type	Required	Example	Tooltip
trap_list_id	number	No	_	Proofpoint TRAP List ID.
trap_member_id	number	No	_	Proofpoint TRAP List member ID.

► Outputs:

```
results = {
    # TODO: Copy and paste an example of the Function Output within this code block.
    # To see view the output of a Function, run resilient-circuits in DEBUG mode and invoke the Function.
    # The Function results will be printed in the logs: "resilient-circuits run -- loglevel=DEBUG"
}
```

► Example Pre-Process Script:

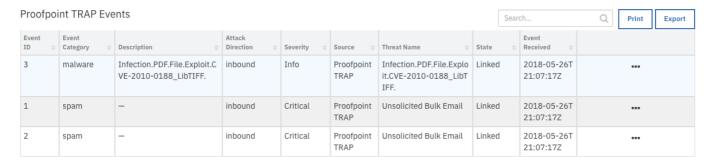
```
inputs.trap_list_id = row.list_id
inputs.trap_member_id = row.member_id
```

```
0.00
# Globals
# List of fields in datatable "Proofpoint TRAP List Members" for
fn_proofpoint_trap_delete_list_member script
DATA_TBL_FIELDS = ["query_execution_time", "status"]
FN_NAME = "fn_proofpoint_trap_delete_list_member"
WF_NAME = "Example: Proofpoint TRAP: Delete List Member"
STATUS = results.content
INPUTS = results.inputs
QUERY_EXECUTION_DATE = results["metrics"]["timestamp"]
# Processing
def main():
    note_text = ''
    if STATUS is not None:
        note_text = "ProofPoint Trap Integration Integration: Workflow <b>{0}</b>:
There was a valid result returned for Resilient function " \
                   "<b>{1}</b>".format(WF_NAME, FN_NAME)
        row.query execution date = QUERY EXECUTION DATE
        if STATUS.lower() == "ok":
            row.status = "Deleted"
    else:
        noteText += "ProofPoint Trap Integration: Workflow <b>{0}</b>: There were
<b>no</b> results returned for " \
                    "list id <b>{1}</b> for Resilient function <b>{3}
</b>".format(WF NAME, INPUTS["trap list id"], FN NAME)
    incident.addNote(helper.createRichText(note_text))
if __name__ == "__main__":
    main()
```

Data tables:

Data Table - Proofpoint TRAP Events

This data table is populated by the **poller** for each Resilient incident and has an entry for each event detected in the corresponding Proofpoint incident.



API Name:

proofpoint_trap_events

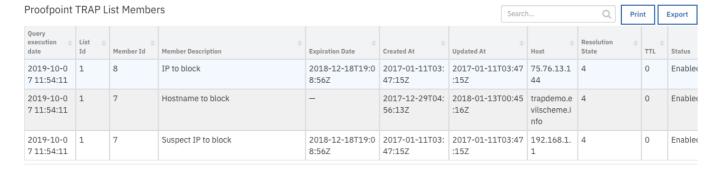
Columns:

Column Name	API Access Name	Туре	Tooltip
Attack Direction	event_attackdirection	text	-
Event Category	event_category	text	-
Description	event_description	text	-
Event Id	event_id	number	-
Event Received	event_received	text	-
Severity	event_severity	text	-
Source	event_source	text	-
State	event_state	text	-
Threat Name	event_threatname	text	-

Data Table - Proofpoint TRAP List Members

This data table is typically initially populated by a workflow using the Function Proofpoint TRAP: Get List Members. An entry is created for all members selected in the workflow.

The datatable is also be updated by functions Proofpoint TRAP: Add Members to List, Proofpoint TRAP: Update List Member and Proofpoint TRAP: Delete List Member.



API Name:

trap_list_members

Columns:

Column Name	API Access Name	Туре	Tooltip
Created At	created_at	text	-
Expiration Date	expiration	text	-
Host	host	text	-
List Id	list_id	number	-
Member Description	member_description	text	-
Member Id	member_id	number	-
-			

Column Name	API Access Name	Туре	Tooltip
Query Execution Date	query_execution_date	text	-
Resolution State	resolution_state	text	-
Status	status	text	-
ΠL	ttl	number	-
Updated At	updated_at	text	-

Custom Fields

Label	API Access Name	Туре	Prefix	Placeholder	Tooltip
proofpoint_trap_incident_id	<pre>proofpoint_trap_incident_id</pre>	number	properties	-	Proofpoint TRAP incidet id.

Rules

Rule Name	Object	Workflow Triggered
Example: Proofpoint Trap: Get Incident Details	incident	wf_proofpoint_trap_get_incident_details
Example: Proofpoint TRAP: Get List Members	incident	wf_proofpoint_trap_get_list_members
Example: Proofpoint TRAP: Add Member to List	artifact	wf_proofpoint_trap_add_member_to_list
Example: Proofpoint TRAP: Update List Member	trap_list_members	wf_proofpoint_trap_update_list_member
Example: Proofpoint TRAP: Delete List Member	trap_list_members	wf_proofpoint_trap_delete_list_member