



LUCY REST API

documentation

For v4.8.3 and newer

September 2021

Table of Contents

General Information	4
Authentication	4
Resources	4
Language	5
Client	5
Scenario Template	5
Attachment Template	6
Awareness Template.....	6
Report Template	7
Campaign Template	7
Campaign	7
Campaign Errors.....	8
Campaign Event	9
Campaign Export.....	9
Campaign Statistics	9
Campaign Report	10
Campaign Custom Field	10
Campaign Schedule Rule.....	10
Campaign Schedule Plan	12
Scenario.....	12
Awareness.....	13
Recipient Group	14
Recipient	14
Recipient Custom Field	15
Victim	15
Awareness Certificate	15
Incident	16
Benchmark	17
Domain.....	17
Endpoint List	18
/api/auth	18
/api/languages	19
/api/clients	19
/api/clients/:id	20
/api/recipient-groups.....	21
/api/recipient-groups/:id	21
/api/recipient-groups/:id/recipients.....	22
/api/recipients/:id.....	23
/api/recipient-custom-fields	24
/api/recipient-custom-fields/:id	25
/api/scenario-templates	26
/api/scenario-templates/:id.....	26

/api/awareness-templates.....	27
/api/awareness-templates/:id	27
/api/attachment-templates	28
/api/attachment-templates/:id	28
/api/report-templates.....	29
/api/campaigns/templates	29
/api/campaigns	30
/api/campaigns/:id.....	31
/api/campaigns/:id/recipient-groups	32
/api/campaigns/:id/recipient-groups/status	33
/api/campaigns/:id/status	33
/api/campaigns/:id/copy	34
/api/campaigns/:id/stats	34
/api/campaigns/:id/report.....	34
/api/campaigns/:id/errors	35
/api/campaigns/:id/victims.....	35
/api/campaigns/victims/:victim_id/re-send/:message_type	35
/api/campaigns/:id/scenarios.....	36
/api/campaigns/:id/awareness.....	37
/api/campaigns/:id/schedule-rules.....	38
/api/campaigns/:id/schedule-plan/generate	41
/api/campaigns/:id/schedule-plan/status	41
/api/campaigns/:id/schedule-plan	41
/api/schedule-rules/:id	42
/api/schedule-rules/:id/recipient-groups	42
/api/schedule-rules/:id/scenarios.....	43
/api/campaign-custom-fields.....	43
/api/campaign-custom-fields/:id	44
/api/scenarios/:id.....	44
/api/scenarios/:id/recipient-groups	45
/api/scenarios/:id/recipient-groups/:recipient_group_id.....	46
/api/scenarios/:id/victims.....	46
/api/incidents.....	47
/api/incidents/:id	47
/api/export/generate.....	48
/api/awareness-certificates/generate	51
/api/awareness-certificates/:id.....	53
/api/awareness-certificates/download/:id.....	53
/api/benchmarks.....	53
/api/benchmarks/:id	54
/api/domains.....	55
/api/domains/:id	56
/api/export/:id	56

General Information

Lucy API is a RESTful web service. The API does not keep the client's state and all requests can be performed in any order.

Some general requirements are:

- The service uses JSON as the only data interchange format, all API responses are in JSON, most request parameters are expected to be in JSON. All requests to API should have a "Content-Type" header set to "application/json".
- All requests should be performed strictly over HTTPS protocol. Plain HTTP requests to API will be rejected by server.

Authentication

Lucy API uses JWT (JSON web tokens) for authentication. The token should be obtained by sending an Authentication request before issuing any other requests (see Authentication request description below for details).

The obtained authentication token should be specified in the "Authorization" HTTP header with "Bearer" scheme, for all subsequent requests. The only request that doesn't require authentication is the authentication request itself, all other requests require a token. Example:

```
Authorization: Bearer  
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoiYWRtaW4iOnRydWV9.TJVA95OrM7E2cBab30RMHrHDcEfxjoYZgeFONFh7HgQ
```

The client should store the token locally and keep it until the token expires. The expiration date and time is specified in the token header. Please check out <https://jwt.io/> for JWT token formation description and libraries for various programming languages.

Resources

Resources are used when getting some data from Lucy API or creating/changing some objects in Lucy. Resource structure is always the same and does not depend on the type of action you do - whether you are getting information from Lucy or creating an object, the object representation doesn't change. For example, campaigns - when you get a list of campaigns or get a single campaign, the structure doesn't differ from what the server expects when you are creating a new campaign.

The only difference is links - when you are posting/putting something to API, the server does not expect any links (and ignores them if they are present in request). The links are only used for exposing relations in GET requests. For specifying relations with other existing objects in system, please use integer IDs.

Language

```
{
  "id": int,
  "name": str,
  "default": boolean,
  "image": "/link/to/language/icon"
}
```

Client

```
{
  "id": int,
  "name": str,
  "country": str,
  "state": str,
  "city": string,
  "address": str,
  "postcode": str,
  "website": str,
  "email": str,
  "phone": str,
  "fax": str,
  "contact": str
}
```

Scenario Template

```
{
  "id": int,
  "created_at": datetime,
  "updated_at": datetime,
  "type": "data|file|mixed|hyperlink|malware_test|usb_attack|awareness",
  "client_id": int,
  "name": str,
  "description": str,
  "no_landing": boolean,
  "category": str,
  "audience": str,
  "message_type": "email|sms",
  "lure_message_type": "email|sms",
  "license": str,
  "license_priority": int,
  "enable": boolean,
  "forbid": boolean,

  "links": [{
    "rel": "client",
    "href": "/link/to/client"
  }, {
    "rel": "icon",
    "href": "/link/to/template/icon"
  }],
}
```

```
"languages": [{
  "id": int,
  "name": str,
  "image": "/link/to/language/icon"
  "preview": "/link/to/template/preview"
},
...
]
```

Attachment Template

```
{
  "id": int,
  "created_at": datetime,
  "updated_at": datetime,
  "name": str,
  "enable": boolean,
  "description": str,
  "variables": [
    str
  ],
  "configurations": [
    str
  ],
  "scenarioType": "data|file|mixed|hyperlink|malware_test|usb_attack|awareness"
}
```

Awareness Template

```
{
  "id": int,
  "created_at": datetime,
  "updated_at": datetime,
  "name": str,
  "description": str,
  "email_sender_name": str,
  "email_sender": str,
  "quiz": Boolean,
  "license": str,
  "license_priority": int,
  "enabled": Boolean,

  "links": [{
    "rel": "icon",
    "href": "/link/to/template/icon"
  }],

  "languages": [{
    "id": int,
    "name": str,
    "image": "/link/to/language/icon"
```

```
    "preview": "/link/to/template/preview"
  },
  ...
]
```

Report Template

```
{
  "id": int,
  "name": str,

  "sections": [{
    "id": str,
    "title": str
  }
  ...
]
```

Campaign Template

```
{
  "id": int,
  "name": str
}
```

Campaign

```
{
  "id": int,
  "created_at": datetime,
  "started_at": datetime,
  "stopped_at": datetime,
  "client_id": int,
  "benchmark_sector_id": int,
  "name": str,
  "status": "idle|running|waiting_approval|stopped",
  "stop_date": datetime,
  "recipients": int,
  "submitted": int,
  "lure_submitted": int,
  "opened": int,
  "clicked": int,
  "succeeded": int,
  "vulnerable": int,
  "errors": int,
  "reported": int,
  "invalid_submits": int,
  "awareness_submitted": int,
  "awareness_opened": int,
```

```
"awareness_completed": int,
"awareness_average_score": float,
"awareness_average_percent": float,
"awareness_average_time": float,
"awareness_rescheduled": int,
"custom_fields": [
  str
],

"links": [{
  "rel": "client",
  "href": "/link/to/client"
}, {
  "rel": "benchmark_sector",
  "href": "/link/to/benchmark_sector"
}, {
  "rel": "awareness_template",
  "href": "/link/to/awareness_template"
}, {
  "rel": "language",
  "href": "/link/to/language"
}]

"stop_report": boolean,
"stop_report_format": "pdf|html|docx",
"stop_report_font_family": str,
"stop_report_font_size": int,
"stop_report_template": int,
"stop_report_sections": [
  int
],
"stop_report_scenarios": [
  int
],
"stop_report_email": str
}
```

Campaign Errors

```
{
  "victim_id": int,
  "recipient_id": int,
  "name": str,
  "email": str,
  "error_text": str,
  "error_time": datetime,
  "error_type": "attack|awareness|lure"
}
```


Campaign Event

```
{
  "type": int,
  "time": datetime,
  "event": str
}
```

- Type:
 - 1 => Campaign created
 - 2 => Campaign started
 - 3 => Campaign stopped
 - 10 => Attack scenario added
 - 11 => Attack scenario removed
 - 20 => Recipient group added
 - 21 => Recipient group removed
 - 50 => Awareness scenario added
 - 51 => Awareness Scenario removed

Campaign Export

```
{
  "name": str,
  "type": int,
  "extension": "csv|xml",
  "status": int,
  "created_at": datetime,
  "id": int,
  "filter": str
}
```

Campaign Statistics

```
{
  "victims": int,
  "submitted": int,
  "clicks": int,
  "success": int,
  "vulnerable": int,
  "awareness_submitted": int,
  "trained": int,
  "report": int,
  "errors": int,
  "timeline": [
    "Resource: Campaign Event",
    "Resource: Campaign Event",
    ...
  ]
}
```

Campaign Report

```
{
  "id": int,
  "campaign_id": int,
  "status": 10|20|30|100,
  "type": "pdf|html|docx",
  "created_at": datetime,

  "links": [{
    "rel": "campaign",
    "href": "/link/to/campaign"
  }]
}
```

- Status:
 - 10 => Processing
 - 20 => Ready
 - 30 => Pending for send
 - 100 => Error

Campaign Custom Field

```
{
  "id": int,
  "campaign_id": int,
  "name": str
}
```

Campaign Schedule Rule

```
{
  "id": int,
  "campaign_id": int,
  "created_at": datetime,
  "processed": boolean,
  "type": 0|1|2,
  "emails_type": 0|1|2|3,
  "weekend": boolean,
  "week_days": array,
  "timezone": str|"default",

  "links": [{
    "rel": "campaign",
    "href": "/link/to/campaign"
  }, {
    "rel": "scenarios",
    "href": "/link/to/schedule/scenarios"
  }, {
    "rel": "recipient_groups",
    "href": "/link/to/schedule/recipient_groups"
  }],
}
```

```

"start_time": datetime,
"stop_time": datetime,
"start_hour": time,
"stop_hour": time,
"sort_type": 0|1|2,
"only_successfully_phished": boolean,
"repeats": int,
"repeat_interval": 0|1|2,
"run_days": array,

"yearly_months": [{
  "month": 1|2|3|4|5|6|7|8|9|10|11|12,
  "scenarios": array
},
...
],

"only_new_victims": boolean,
"scenarios": array,
"groups": array
}

```

- **Type:**
 - 0 => One Shot
 - 1 => Repeating
 - 2 => Yearly Campaign
- **Emails Type:**
 - 2 => All
 - 0 => Lure
 - 1 => Attack
 - 3 => Awareness
- **Weekdays:**
 - 1 => Monday
 - 2 => Tuesday
 - 3 => Wednesday
 - 4 => Thursday
 - 5 => Friday
 - 6 => Saturday
 - 7 => Sunday
- **Sort Type:**
 - 0 => Default
 - 1 => Full Random
 - 2 => By Scenarios
- **Repeat Interval:**
 - 0 => Daily
 - 1 => Weekly
 - 2 => Monthly
- The format of the attributes "start_time" and "stop_time": "YYYY-MM-DD HH:mm"
- The format of the attributes "start_hour" and "stop_hour": "HH:mm"
- The attribute "only_successfully_phished" is applicable for "emails_type"=3 only.
- The attributes "scenarios" and "groups" are used in POST\PUT requests only.

Campaign Schedule Plan

```
{
  "rule_id": int,
  "time": datetime,
  "victim_id": int,
  "status": 0|10|20,
  "web_proxy_copied": boolean,

  "links": [{
    "rel": "victim",
    "href": "/link/to/victim"
  }]
}
```

- Status:
 - 0 => New
 - 10 => Processed
 - 20 => Error

Scenario

```
{
  "id": int,
  "created_at": datetime,
  "started_at": datetime,
  "stopped_at": datetime,
  "campaign_id": int,
  "scenario_template_id": int,
  "language_id": int,
  "name": str,
  "domain": str,
  "subdomain": str,
  "full_domain": str,
  "sender_name": str,
  "sender_email": str,
  "random_email": boolean,
  "redirect_url": str,
  "collect_data": "no|partial|full",
  "login_regexp": str,
  "password_regexp": str,
  "forward_email": str,
  "message_type": "email|sms",
  "double_barrel": boolean,
  "lure_sender_name": str,
  "lure_sender_email": str,
  "lure_random_email": boolean,
  "lure_delay": int,
  "lure_message_type": "email|sms",
  "status": "idle|running",
  "send_awareness_link": boolean,
  "awareness_delay": int,
  "no_landing": boolean,
  "track_opened_emails": boolean,
  "use_reply_to": boolean,
}
```

```

"anonymous_mode": boolean,
"success_action": "click|post|download|file_data",
"recipient_count": int,
"messages_sent": int,
"clicks": int,
"successful_attacks": int,
"vulnerable_victims": int,
"errors": int,

"links": [{
  "rel": "campaign",
  "href": "/link/to/campaign"
}, {
  "rel": "scenario_template",
  "href": "/link/to/scenario_template"
}, {
  "rel": "language",
  "href": "/link/to/language"
}]
}

```

Awareness

```

{
  "id": int,
  "name": str,
  "campaign_id": int,
  "template_id": int,
  "language_id": int,
  "website_online": true,
  "sender_email": str,
  "sender_name": str,
  "subject": str,
  "domain": str,
  "quiz": boolean,
  "risk_level": int,
  "public_id": str,
  "website_type": 1|2,
  "course_id": int,

  "links": [{
    "rel": "campaign",
    "href": "/link/to/campaign"
  }, {
    "rel": "awareness_template",
    "href": "/link/to/awareness_template"
  }, {
    "rel": "language",
    "href": "/link/to/language"
  }],

  "scenarios": [{
    "scenario_id": int,
    "risk_level": int
  }],
}

```

```
...
]
}
```

- Website_type:
 - 1 => Regular awareness
 - 2 => Adapt Learning

Recipient Group

```
{
  "id": int,
  "name": str,
  "usb_attack": boolean
}
```

Recipient

```
{
  "id": int,
  "last_tested": datetime,
  "recipient_group_id": int,
  "language_id": int,
  "email": str,
  "name": str,
  "phone": str,
  "staff_type": str,
  "location": str,
  "division": str,
  "comment": str,
  "link": str,
  "gender": str,
  "ldap_based": boolean,
  "firstname": str,
  "lastname": str,
  "objectguid": str,
  "external_id": str,

  "links": [{
    "rel": "recipient_group",
    "href": "/link/to/recipient_group"
  }, {
    "rel": "language",
    "href": "/link/to/language"
  }]

  "custom_fields": [
    str
  ]
}
```

Recipient Custom Field

```
{
  "id": int,
  "name": str,
  "ldap_attribute": str
}
```

Victim

```
{
  "id": int,
  "recipient_id": int,
  "lure_submitted_at": datetime,
  "submitted_at": datetime,
  "awareness_submitted_at": datetime,
  "opened_at": datetime,
  "clicked_at": datetime,
  "succeeded_at": datetime,
  "trained_at": datetime,
  "reported_at": datetime,
  "os": str,
  "browser": str,
  "ip": str,
  "proxy_ip": str,
  "country": str,
  "plugins": str,
  "clicks": int,
  "email": str,
  "name": str,
  "phone": str,
  "staff_type": str,
  "location": str,
  "division": str,
  "comment": str,
  "link": str,
  "average_quiz_score": int,
  "error": boolean,
  "error_time": datetime,
  "error_text": str,

  "links": [{
    "rel": "recipient",
    "href": "/link/to/recipient"
  }]
}
```

Awareness Certificate

```
{
  "id": int,
  "type": 1001|1002|1003,
  "format": 0|1,
  "object_id": int,
}
```

```
"status": 10|20|100,  
"email": str,  
"sendTo": str,  
"error": str  
}
```

- Type:
 - 1001 => Victim
 - 1002 => Recipient
 - 1003 => Campaign
- Format – format of the file:
 - 0 => PDF
 - 1 => JPG\JPEG
- Object_id – one of the following values depending on the Type:
 - ID of victim (Type = 1001)
 - ID of campaign (Type = 1003)
 - Leave empty for Type = 1002
- Email:
 - Recipient email address => only used for Type = 1002
- Status:
 - 10 => Processing
 - 20 => Ready
 - 100 => Error

Incident

```
{  
  "id": int,  
  "email": str,  
  "status": "open|in_progress|dismissed|simulation|real_phishing|closed",  
  "time": datetime,  
  "score": float,  
  "more_analysis": boolean,  
  "client_id": int,  
  "campaign_id": int,  
  
  "links": [{  
    "rel": "self",  
    "href": "/api/incidents/:id"  
  }, {  
    "rel": "delete",  
    "href": "/api/incidents/:id"  
  }],  
  
  "message_url": "/admin/incident/:id/message",  
  "subject": str,  
  
  "headers": [{  
    "from": str,  
    "ip": str,  
    "by": str,  
    "with": str,
```



```
    "time": str,  
    "blacklisted": boolean,  
    "score": float  
  },  
  ...  
],  
  
"domains": [{  
  "source": str,  
  "domain": str,  
  "score": float,  
  "checks": [{  
    "service_name": str,  
    "flagged": boolean  
  },  
  ...  
]  
},  
...  
],  
  
"body_analysis": [{  
  "rule": str,  
  "score": float  
},  
...  
]  
}
```

Benchmark

```
{  
  "id": int,  
  "name": str  
}
```

Domain

```
{  
  "id": int,  
  "name": str  
}
```

Endpoint List

A list below contains all endpoints exposed by Lucy API. Every endpoint corresponds to some system resource or action. If endpoint returns some system resources or expects some common resource as an input, the specification will refer to that resource using "Resource:Resource Name" mnemonic. For example, if you see "Resource:Campaign" in request/response body, that means you should refer to this document's Resources section and find the corresponding resource there.

List endpoints support 3 optional GET parameters, that you can use to implement ordering and limits:

- `sort` - column to sort by. The default sorting works in ascending order, if you wish to sort in descending order, specify a minus sign before the column name (Example: `sort=-created_at`)
- `offset` - offset results to the specified number of entries
- `limit` - max number of entries

Example:

```
GET /api/campaigns?sort=-created_at&offset=10&limit=10
```

/api/auth

POST

- Client authentication
- Request:

```
{
  "email": "user email",
  "password": "user password"
}
```

- Response:

```
{
  "token": "authentication token",
  "timezone": str,
  "expire": int
}
```

token => Bearer Token

timezone => current time zone configured in the system

expire => datetime in seconds since Unix Epoch when the token will expire

/api/languages

GET

- Get a list of languages
- Resources:
 - [Language](#)
- Response:

```
{
  "languages": [
    "Resource:Language",
    "Resource:Language",
    ...
  ]
}
```

/api/clients

GET

- Get a list of clients
- Resources:
 - [Client](#)
- Response:

```
{
  "clients": [
    "Resource:Client",
    "Resource:Client",
    ...
  ]
}
```

PUT

- Create a client
- Resources:
 - [Client](#)
- Request:

```
"Resource:Client"
```

Mandatory fields:

- name
- Response:

```
{
  "client":
    "Resource:Client"
}
```

/api/clients/:id

GET

- Get client
- Resources:
 - [Client](#)
- Response:

```
{
  "client":
    "Resource:Client"
}
```

POST

- Save client
- Resources:
 - [Client](#)
- Request:

```
"Resource:Client"
```

Mandatory fields:

- id
- name
- Response:

```
{
  "client":
    "Resource:Client"
}
```

DELETE

- Delete client
- Response: N/A

/api/recipient-groups

GET

- Get a list of recipient groups
- Resources:
 - [Recipient Group](#)
- Response:

```
{
  "recipient-groups": [
    "Resource:Recipient Group",
    "Resource:Recipient Group",
    ...
  ]
}
```

PUT

- Create a recipient group
- Resources:
 - [Recipient Group](#)
- Request:

```
"Resource:Recipient Group"
```

Mandatory fields:

- name
- usb_attack
- Response:

```
{
  "recipient-group":
    "Resource:Recipient Group"
}
```

/api/recipient-groups/:id

GET

- Get recipient group
- Resources:
 - [Recipient Group](#)
- Response:

```
{
  "recipient-group":
    "Resource:Recipient Group"
}
```

POST

- Save recipient group
- Resources:
 - [Recipient Group](#)
- Request:

```
"Resource:Recipient Group"
```

Mandatory fields:

- Name
 - usb_attack
- Response:

```
{  
  "recipient-group":  
    "Resource:Recipient Group"  
}
```

DELETE

- Delete recipient group
- Response: N/A

/api/recipient-groups/:id/recipients

GET

- Get a list of recipients in group
- Resources:
 - [Recipient](#)
- Response:

```
{  
  "recipients": [  
    "Resource:Recipient",  
    "Resource:Recipient",  
    ...  
  ]  
}
```

PUT

- Create a recipient
- Resources:
 - [Recipient](#)
- Request:

```
"Resource:Recipient"

OR

[
  "Resource:Recipient",
  "Resource:Recipient",
  ...
]
```

Mandatory fields:

- name
 - email
-
- Response:

```
{
  "recipient-group":
    "Resource:Recipient Group"
}
```

/api/recipients/:id

GET

- Get recipient
- Resources:
 - [Recipient](#)
- Response:

```
{
  "recipient":
    "Resource:Recipient"
}
```

POST

- Save recipient
- Resources:
 - [Recipient](#)
- Request:

```
"Resource:Recipient"
```

Mandatory fields:

- name
- Response:

```
{  
  "recipient":  
    "Resource:Recipient"  
}
```

DELETE

- Delete recipient
- Response: N/A

/api/recipient-custom-fields

GET

- Get a list of recipient custom fields
- Resources:
 - [Recipient Custom Field](#)
- Response:

```
{  
  "recipient-custom-fields": [  
    "Resource:Recipient Custom Field",  
    "Resource:Recipient Custom Field",  
    ...  
  ]  
}
```


PUT

- Create a recipient custom field
- Resources:
 - [Recipient Custom Field](#)
- Request:

```
{  
  "name": str  
}
```

- Response:

```
{  
  "recipient-custom-field":  
    "Resource:Recipient Custom Field"  
}
```

/api/recipient-custom-fields/:id

GET

- Get recipient custom fields
- Resources:
 - [Recipient Custom Field](#)
- Response:

```
{  
  "recipient-custom-field":  
    "Resource:Recipient Custom Field"  
}
```

POST

- Save recipient custom field
- Resources:
 - [Recipient Custom Field](#)
- Request:

```
"Resource:Recipient Custom Field "
```

- Response:

```
{  
  "recipient-custom-field":  
    "Resource:Recipient Custom Field"  
}
```

DELETE

- Delete recipient custom field
- Response: N/A

/api/scenario-templates

GET

- Get a list of scenario templates
- Resources:
 - [Scenario Template](#)
- Response:

```
{
  "scenario-templates": [
    "Resource:Scenario Template",
    "Resource:Scenario Template",
    ...
  ]
}
```

/api/scenario-templates/:id

GET

- Get scenario template
- Resources:
 - [Scenario Template](#)
- Response:

```
{
  "scenario-template":
    "Resource:Scenario Template"
}
```

DELETE

- Delete scenario template
- Response: N/A

/api/awareness-templates

GET

- Get a list of awareness templates
- Resources:
 - [Awareness Template](#)
- Response:

```
{
  "awareness-templates": [
    "Resource:Awareness Template",
    "Resource:Awareness Template",
    ...
  ]
}
```

/api/awareness-templates/:id

GET

- Get awareness template
- Resources:
 - [Awareness Template](#)
- Response:

```
{
  "awareness-template":
    "Resource:Awareness Template"
}
```

DELETE

- Delete awareness template
- Response: N/A

/api/attachment-templates

GET

- Get a list of attachment templates
- Resources:
 - [Attachment Template](#)
- Response:

```
{
  "attachment-templates": [
    "Resource:Attachment Template",
    "Resource:Attachment Template",
    ...
  ]
}
```

/api/attachment-templates/:id

GET

- Get attachment template
- Resources:
 - [Attachment Template](#)
- Response:

```
{
  "attachment-template":
    "Resource:Attachment Template"
}
```

DELETE

- Delete attachment template
- Response: N/A

/api/report-templates

GET

- Get a list of report templates
- Resources:
 - [Report Template](#)
- Response:

```
{
  "report-templates": [
    "Resource:Report Template",
    "Resource:Report Template",
    ...
  ]
}
```

/api/report-templates/:id

GET

- Get report template
- Resources:
 - [Report Template](#)
- Response:

```
{
  "report-template":
    "Resource:Report Template"
}
```

/api/campaigns/templates

GET

- Get a list of campaign templates
- Resources:
 - [Campaign Template](#)
- Response:

```
{
  "campaign-templates": [
    "Resource:Campaign Template",
    "Resource:Campaign Template",
    ...
  ]
}
```

/api/campaigns

GET

- Get a list of campaigns
- Resources:
 - [Campaign](#)
- Response:

```
{
  "campaigns": [
    "Resource: Campaign",
    "Resource: Campaign",
    ...
  ]
}
```

PUT

- Create a campaign
- Resources:
 - [Campaign](#)
- Request:

```
"Resource: Campaign"
```

Mandatory fields:

- name
 - client_id
- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

PUT

- Create a campaign with default template
- Resources:
 - [Campaign](#)
- Request:

```
{
  "Resource: Campaign",
  "mode": 3,
  "campaign_template_id": int
}
```

Mandatory fields:

- name
- client_id
- mode => always equal to 3
- campaign_template_id => ID of the campaign template from */api/campaigns/templates*

- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

/api/campaigns/:id

GET

- Get campaign
- Resources:
 - [Campaign](#)
- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

POST

- Save campaign
- Resources:
 - [Campaign](#)
- Request:

```
"Resource: Campaign"
```

Mandatory fields:

- name
- client_id
- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

DELETE

- Delete campaign
- Response: N/A

/api/campaigns/:id/recipient-groups

GET

- Get a list of recipient groups in campaign
- Resources:
 - [Recipient Group](#)
- Response:

```
{
  "recipient-groups": [
    "Resource:Recipient Group",
    "Resource:Recipient Group",
    ...
  ]
  "status": "running|completed"
}
```

PUT

- Add a recipient group to a campaign (recipients will be added to all scenarios in campaign)
- Resources:
 - [Recipient Group](#)
- Request:

```
{
  "id": int
}
```

- Response:

```
{
  "recipient-group":
    "Resource:Recipient Group",
  "status": "running|completed"
}
```


/api/campaigns/:id/recipient-groups/status

GET

- Get current status of campaign
- Resources:
 - [Recipient Group](#)
- Response:

```
{
  "running": true
}
```

/api/campaigns/:id/recipient-groups/:recipient_group_id

DELETE

- Delete recipient group from campaign
- Response: N/A

/api/campaigns/:id/status

POST

- Change campaign status.
This endpoint intended to run, stop or restart a campaign; therefore it supports only POST request and only three statuses
- Resources:
 - [Campaign](#)
- Request:

```
{
  "status": "running|stopped|restart"
}
```

- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

/api/campaigns/:id/copy

POST

- Copy a campaign with all scenarios.
- Resources:
 - [Campaign](#)
- Request: N/A
- Response:

```
{
  "campaign":
    "Resource: Campaign"
}
```

/api/campaigns/:id/stats

GET

- Get campaign statistics
- Resources:
 - [Campaign Statistics](#)
- Response:

```
{
  "Resource: Campaign Statistics"
}
```

/api/campaigns/:id/report

POST

- Create a campaign report.
- Resources:
 - [Campaign Report](#)
- Request:

```
{
  "template_id": int,
  "format": "pdf|html|docx",
  "font_size": int,
  "font_family": str,
  "scenarios": [
    int
  ],
  "sections": [
    int
  ]
}
```

- Response:

```
{
  "report":
    "Resource: Campaign Report"
}
```

/api/campaigns/:id/errors

GET

- Get campaign errors
- Resources:
 - [Campaign Errors](#)
- Response:

```
{
  "errors": [
    "Resource: Campaign Errors",
    "Resource: Campaign Errors",
    ...
  ]
}
```

/api/campaigns/:id/victims

GET

- Get campaign victims
- Resources:
 - [Victim](#)
- Response:

```
{
  "victims": [
    "Resource: Victim",
    "Resource: Victim",
    ...
  ]
}
```

/api/campaigns/victims/:victim_id/re-send/:message_type

POST

- Resend email for the specified victim and message type
- Request:
Possible values for ":message_type":
 - attack
 - lure
 - awareness

- Response:

```
{
  "status": "ok"
}
```

/api/campaigns/:id/scenarios

GET

- Get a list of scenarios in campaign
- Resources:
 - [Scenario](#)
- Response:

```
{
  "scenarios": [
    "Resource:Scenario",
    "Resource:Scenario",
    ...
  ]
}
```

PUT

- Create scenario in campaign
- Resources:
 - [Scenario](#)
- Request:

```
"Resource:Scenario"
```

Mandatory fields:

- name
- domain
- language_id
- scenario_template_id

- Response:

```
{
  "scenario":
    "Resource:Scenario"
}
```

/api/campaigns/:id/awareness

GET

- Get a list of awareness in campaign
- Resources:
 - [Awareness](#)
- Response:

```
{
  "awareness": [
    "Resource:Awareness",
    "Resource:Awareness",
    ...
  ]
}
```

PUT

- Create awareness in campaign
- Resources:
 - [Awareness](#)
- Request:

```
"Resource:Awareness"
```

Mandatory fields:

- name
 - domain
 - language_id
 - template_id
- Response:

```
{
  "awareness":
    "Resource:Awareness"
}
```

/api/campaigns/:id/schedule-rules

GET

- Get a list of schedule rules in campaign
- Resources:
 - [Campaign Schedule Rule](#)
- Response:

```
{
  "schedule-rules": [
    "Resource: Campaign Schedule Rule",
    "Resource: Campaign Schedule Rule",
    ...
  ]
}
```

PUT

- Create a One-Shot schedule rule
- Resource:
 - [Campaign Schedule Rule](#)
- Request:

```
{
  "type": 0,
  "emails_type": 0|1|2|3,
  "timezone": str|"default",
  "weekend": boolean,
  "week_days": array|[],
  "start_time": datetime,
  "stop_time": datetime,
  "sort_type": 0|1|2,
  "only_successfully_phished": boolean,
  "scenarios": [
    int,
    ...
  ],
  "groups": [
    int,
    ...
  ]
}
```

- Response:

```
{
  "schedule-rule":
    "Resource: Campaign Schedule Rule"
}
```

PUT

- Create a Repeating schedule rule
- Resource:
 - [Campaign Schedule Rule](#)
- Request:

```
{
  "type": 1,
  "emails_type": 0|1|2|3,
  "repeats": int,
  "repeat_interval": 0|1|2,
  "timezone": str|"default",
  "weekend": boolean,
  "week_days": array|[],
  "start_time": datetime,
  "start_hour": time,
  "stop_hour": time,
  "sort_type": 0|1|2,
  "only_successfully_phished": boolean,
  "run_days": array,
  "scenarios": [
    int,
    ...
  ],
  "groups": [
    int,
    ...
  ]
}
```

- Response:

```
{
  "schedule-rule":
    "Resource: Campaign Schedule Rule"
}
```

PUT

- Create a Yearly Campaign schedule rule
- Resource:
 - [Campaign Schedule Rule](#)
- Request:

```
{
  "type": 2,
  "timezone": str|"default",
  "weekend": boolean,
  "week_days": array|[],
  "start_hour": time,
  "stop_hour": time,
  "run_days": array,
  "only_new_victims": boolean,
  "yearly_months": [{
    "month": 1|2|3|4|5|6|7|8|9|10|11|12,
    "scenarios": [
      int
    ]
  },
  ...
]
"scenarios": [
  int,
  ...
],
"groups": [
  int,
  ...
]
}
```

- Response:

```
{
  "schedule-rule":
    "Resource: Campaign Schedule Rule"
}
```


/api/campaigns/:id/schedule-plan/generate

POST

- Request to generate schedule plan
- Resources:
 - [Campaign Schedule Plan](#)
- Request: N/A
- Response:

```
{
  "status": "running"
}
```

/api/campaigns/:id/schedule-plan/status

GET

- Get the status of a request for the generation of the schedule plan.
- Resources:
 - [Campaign Schedule Plan](#)
- Response:

```
{
  "running": boolean
}
```

/api/campaigns/:id/schedule-plan

GET

- Get a list of schedule plan events
- Resources:
 - [Campaign Schedule Plan](#)
- Response:

```
{
  "schedule-plan": [
    "Resource: Campaign Schedule Plan",
    "Resource: Campaign Schedule Plan",
    ...
  ]
}
```

/api/schedule-rules/:id

GET

- Get schedule rule
- Resources:
 - [Campaign Schedule Rule](#)
- Response:

```
{
  "schedule-rule":
    "Resource: Campaign Schedule Rule"
}
```

POST

- Save schedule rule
- Resources:
 - [Campaign Schedule Rule](#)
- Request:

```
"Resource: Campaign Schedule Rule"
```

- Response:

```
{
  "schedule-rule":
    "Resource: Campaign Schedule Rule"
}
```

DELETE

- Delete schedule rule
- Response: N/A

/api/schedule-rules/:id/recipient-groups

GET

- Get a list of the recipient groups in schedule rule
- Resources:
 - [Campaign Schedule Rule](#)
- Response:

```
{
  "recipient-groups": [
    "Resource: Recipient Group",
    "Resource: Recipient Group",
    ...
  ]
}
```

/api/schedule-rules/:id/scenarios

GET

- Get a list of the scenarios in schedule rule
- Resources:
 - [Campaign Schedule Rule](#)
- Response:

```
{
  "scenarios ": [
    "Resource:Scenario",
    "Resource:Scenario",
    ...
  ]
}
```

/api/campaign-custom-fields

GET

- Get a list of campaign custom fields
- Resources:
 - [Campaign Custom Field](#)
- Response:

```
{
  "campaign-custom-fields": [
    "Resource:Campaign Custom Field",
    "Resource:Campaign Custom Field",
    ...
  ]
}
```

PUT

- Create a campaign custom field
- Resources:
 - [Campaign Custom Field](#)
- Request:

```
{
  "campaign_id": int,
  "name": str
}
```

- Response:

```
{
  "campaign-custom-field":
    "Resource:Campaign Custom Field"
}
```

/api/campaign-custom-fields/:id

GET

- Get campaign custom fields
- Resources:
 - [Campaign Custom Field](#)
- Response:

```
{
  "campaign-custom-field":
    "Resource: Campaign Custom Field"
}
```

POST

- Save campaign custom field
- Resources:
 - [Campaign Custom Field](#)
- Request:

```
"Resource: Campaign Custom Field"
```

- Response:

```
{
  "campaign-custom-field":
    "Resource: Campaign Custom Field"
}
```

DELETE

- Delete campaign custom field
- Response: N/A

/api/scenarios/:id

GET

- Get scenario
- Resources:
 - [Scenario](#)
- Response:

```
{
  "scenario":
    "Resource: Scenario"
}
```

POST

- Save scenario
- Resources:
 - [Scenario](#)
- Request:

```
"Resource:Scenario"
```

Mandatory fields:

- name
 - domain
 - language_id
 - scenario_template_id
- Response:

```
{  
  "scenario":  
    "Resource:Scenario"  
}
```

DELETE

- Delete scenario from campaign
- Response: N/A

/api/scenarios/:id/recipient-groups

GET

- Get a list of recipient groups in scenario
- Resources:
 - [Recipient Group](#)
- Response:

```
{  
  "recipient-groups": [  
    "Resource:Recipient Group",  
    "Resource:Recipient Group",  
    ...  
  ]  
}
```

PUT

- Add a recipient group to a scenario
- Resources:
 - [Recipient Group](#)
- Request:

```
{
  "id": int
}
```

- Response:

```
{
  "recipient-group":
    "Resource:Recipient Group"
}
```

`/api/scenarios/:id/recipient-groups/:recipient_group_id`

DELETE

- Delete recipient group from a scenario
- Response: N/A

`/api/scenarios/:id/victims`

GET

- Get scenario victims
- Resources:
 - [Victim](#)
- Response:

```
{
  "victims": [
    "Resource:Victim",
    "Resource:Victim",
    ...
  ]
}
```

/api/incidents

GET

- Get a list of phishing incident reports
- Resources:
 - [Incident](#)
- Response:

```
{
  "incidents": [
    "Resource:Incident",
    "Resource:Incident",
    ...
  ]
}
```

/api/incidents/:id

GET

- Get incident
- Resources:
 - [Incident](#)
- Response:

```
{
  "incident":
    "Resource:Incident"
}
```

DELETE

- Delete incident
- Response: N/A

/api/export/generate

POST

- Export campaign data
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,

  "campaign_id": int,
  "awareness_id": int,
  "role_group": "admin|view|user|supervisor|enduser",
  "group_id": int,
  "template_id": int,
  "language_id": int

  "extension": "csv|xml"
}
```

Possible values in "type":

- 10 => All Victims
 - 11 => Succeeded Victims
 - 12 => Not Succeeded Victims
 - 13 => Clicked Victims
 - 14 => Started Training
 - 15 => Victims Opened Emails
 - 16 => Not Started Training
 - 17 => Completed Training
 - 20 => Downloaded File
 - 21 => Received Training Certificate
 - 22 => All Victims By Recipient Group
 - 30 => Collected Data
 - 40 => Visits
 - 41 => Scenario Visits
 - 42 => Awareness Visits
 - 50 => Quiz Answers
 - 60 => Schedule Plan
 - 80 => Campaign Comparisons
 - 90 => Campaign Benchmarks
 - 121 => Campaign Scenarios
 - 210 => Mail & Web Results
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```


POST

- Export campaign awareness template to SCORM package
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,
  "awareness_id": int,
  "extension": "csv|xml"
}
```

Possible values in "type":

- 200 => SCORM template of Campaign Awareness
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```

POST

- Export user SAML links
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,
  "role_group": "admin|view|user|supervisor|enduser",
  "extension": "csv|xml"
}
```

Possible values in "type":

- 400 => SAML links
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```

POST

- Export recipients of the specified recipient group
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,
  "group_id": int,
  "extension": "csv|xml"
}
```

Possible values in "type":

- 300 => Recipients
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```

POST

- Export awareness template to SCORM package
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,
  "template_id": int,
  "language_id": int
  "extension": "csv|xml"
}
```

Possible values in "type":

- 201 => SCORM template of Awareness Template
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```

POST

- Export common data
- Resources:
 - [Campaign Export](#)
- Request:

```
{
  "name": str,
  "type": int,
  "extension": "csv|xml"
}
```

Possible values in "type":

- 70 => System Status Log
 - 100 => Clients
 - 110 => Campaigns
 - 120 => Scenarios
 - 500 => Endusers
 - 600 => Overall Recipients Statistics
- Response:

```
{
  "export":
    "Resource: Campaign Export"
}
```

/api/awareness-certificates/generate

POST

- Request a certificate for all victims of the campaign
- Resources:
 - [Awareness Certificate](#)
- Request:

```
{
  "type": 1003,
  "format": 0|1,
  "object_id": int
}
```

object_id => ID of campaign

- Response:

```
{
  "awareness-certificate":
    "Resource: Awareness Certificate"
}
```

POST

- Request a certificate for the specified victim
- Resources:
 - [Awareness Certificate](#)
- Request:

```
{
  "type": 1001,
  "format": 0|1,
  "object_id": int,
  "sendTo": str
}
```

object_id => ID of victim

sendTo => an email address to which the certificate file will be sent

- Response:

```
{
  "awareness-certificate":
    "Resource:Awareness Certificate"
}
```

POST

- Request a certificate for the specified recipient
- Resources:
 - [Awareness Certificate](#)
- Request:

```
{
  "type": 1002,
  "format": 0|1,
  "email": str
}
```

email => email address of the recipient

- Response:

```
{
  "awareness-certificate":
    "Resource:Awareness Certificate"
}
```

/api/awareness-certificates/:id

GET

- Check awareness certificate status
- Resources:
 - [Awareness Certificate](#)
- Response:

```
{
  "awareness-certificate":
    "Resource:Awareness Certificate"
}
```

/api/awareness-certificates/download/:id

GET

- Download the archive with awareness certificates
- Resources:
 - [Awareness Certificate](#)
- Response:

```
zip file
```

/api/benchmarks

GET

- Get a list of benchmarks
- Resources:
 - [Benchmark](#)
- Response:

```
{
  "benchmarks": [
    "Resource:Benchmark",
    "Resource:Benchmark",
    ...
  ]
}
```

PUT

- Create a benchmark
- Resources:
 - [Benchmark](#)
- Request:

```
"Resource: Benchmark"
```

Mandatory fields:

- name
- Response:

```
{  
  "benchmark":  
    "Resource: Benchmark"  
}
```

/api/benchmarks/:id

GET

- Get benchmark
- Resources:
 - [Benchmark](#)
- Response:

```
{  
  "benchmark":  
    "Resource: Benchmark"  
}
```

POST

- Save benchmark
- Resources:
 - [Benchmark](#)
- Request:

```
"Resource: Benchmark"
```

Mandatory fields:

- id
- name

- Response:

```
{
  "benchmark":
    "Resource: Benchmark"
}
```

DELETE

- Delete benchmark
- Response: N/A

/api/domains

GET

- Get a list of domains
- Resources:
 - [Domain](#)
- Response:

```
{
  "domains": [
    "Resource: Domain",
    "Resource: Domain",
    ...
  ]
}
```

PUT

- Create a domain
- Resources:
 - [Domain](#)
- Request:

```
"Resource: Domain"
```

Mandatory fields:

- name
- Response:

```
{
  "domain":
    "Resource: Domain"
}
```

/api/domains/:id

DELETE

- Delete domain
- Response: N/A

/api/export/:id

GET

- Get an Export for the specified ID
- Resources:
 - [Campaign Export](#)
- Response:

```
{
  "export": {
    "id": int,
    "name": str,
    "type": int,
    "extension": str,
    "status": int,
    "created_at": str,
    "filter": str,
  }
}
```

Possible values in "status":

- 10 - Processing
- 20 - Ready
- 100 - Error