REDCap API & DET

What is DET ?

The Data Entry Trigger is an advanced feature. It provides a way for REDCap to trigger a call to a remote web address (URL), in which it will send a HTTP Post request to the specified URL whenever *any* record or survey response has been created or modified on *any* data collection instrument or survey in this project (it is *not* triggered by data imports but only by normal data entry on surveys and data entry forms). Its main purpose is for notifying other remote systems outside REDCap at the very moment a record/response is created or modified, whose purpose may be to trigger some kind of action by the remote website, such as making a call to the REDCap API.

In the HTTP Post request, the following parameters will be sent by REDCap in order to provide a context for the record that has just been created/modified:

• **project_id** - The unique ID number of the REDCap project (i.e. the 'pid' value found in the URL when accessing the project in REDCap).
• **username** - The username of the REDCap user that is triggering the Data Entry Trigger. Note: If it is triggered by a survey page (as opposed to a data entry form), then the username that will be reported will be '[survey respondent]'.
• **instrument** - The unique name of the current data collection instrument (all your project's unique instrument names can be found in column B in the data dictionary).
• **record** - The name of the record being created or modified, which is the record's value for the project's first field.
• **redcap_event_name** - The unique event name of the event for which the record was modified (for longitudinal projects only).
• **redcap_data_access_group** - The unique group name of the Data Access Group to which the record belongs (if the record belongs to a group).
• **[instrument]_complete** - The status of the record for this particular data collection instrument, in which the value will be 0, 1, or 2. For data entry forms, 0=Incomplete, 1=Unverified, 2=Complete. For surveys, 0=partial survey response and 2=completed survey response. This parameter's name will be the variable name of this particular instrument's status field, which is the name of the instrument + '_complete'.
• **redcap_url** - The base web address to REDCap (URL of REDCap's home page).
i.e., https://redcap.bcahsn.ca/
• **project_url** - The base web address to the current REDCap project (URL of its Project Home page).
i.e., https://redcap.bcahsn.ca/redcap_v6.5.20/index.php?pid=XXXX

What is an API?

The acronym "API" stands for "Application Programming Interface". An API is just a defined way for a program to accomplish a task, usually retrieving or modifying data.

REDCap API to make applications, websites, widgets, and other projects that interact with REDCap. Programs talk to the REDCap API over HTTP, the same protocol that your browser uses to visit and interact with web pages.

API security : Best Practices:

Although API requests to REDCap are done using SSL (HTTPS), which means that the traffic to and from the REDCap server is encrypted, there is still more that can be done to ensure the highest level of security when using the API. This is especially important if you are moving sensitive data into or out of REDCap. One thing that is *highly* recommended is for your API script/program (i.e. the thing making the request to the REDCap API) to validate the SSL certificate of the REDCap web server when it makes the API request.

How to prevent Man in the Middle attacks:
Preventing MiM attacks is pretty simple. Essentially all you need to do is to force your API script to validate the SSL certificate of the REDCap server. REDCap's SSL certificate will always be valid, but the hacker's fake certificate can never be determined to be valid if you attempt to validate it. In many programs or programming languages that can make API requests, validating an SSL certificate is often as easy as setting a flag. For example, cURL is popularly used by many API scripts in programming languages such as PHP, R, SAS, and many more in order to make the web request to REDCap. **So if your API script is utilizing cURL, all you need to do is modify your script so that it sets the cURL option named CURLOPT_SSL_VERIFYPEER to have a value of TRUE.** Once done, your API script will attempt to make the API request to REDCap *only* if it can validate REDCap's SSL certificate. **Thus by adding the SSL certificate check, you have completely prevented the possibility of MiM attacks and are using the most secure form of communication with the REDCap API.** If you are not using cURL, there are plenty of other examples on the web for how to validate an SSL certificate in different programming languages. Such examples can be found simply by Googling the name of your programming language + "verify ssl certificate" (e.g., "Java verify ssl certificate"), which should provide you with many helpful results.

REMINDER: Please remember that while REDCap itself has many security layers to help protect you and to ensure the highest level of security

and data integrity, **it is *your* responsiblity to ensure that you are using the most secure methods and best practices when using the REDCap API.**

Supported Actions currently in BCCHR REDCap API (LTS version 6.10)

Supported Actions:
Export Records
Export Reports
Import Records
Export Metadata (i.e. Data Dictionary)
Export List of Export Field Names (i.e. variables used during exports and imports)
Export a File
Import a File
Delete a File
Export Instruments (i.e., Data Entry Forms)
Export PDF file of Data Collection Instruments (either as blank or with data)
Export a Survey Link for a Participant
Export a Survey Queue Link for a Participant
Export a Survey Return Code for a Participant
Export a Survey Participant List
Export Events
Export Arms
Export Instrument-Event Mappings
Export Users
Export Project Information
Export REDCap Version

Demo:

I'm going to demonstrate a simple API by using PHP programming code -

Exporting records for a project

Connecting to Redcap - You'll need to know your API token (issued by REDCap administrator) and URL of REDCap. If you use PHP program language, their is inbuilt REDCap class file called 'RestCall Request.php' - which performs the API operations Call automatically.

```php
<?php

  //
  // Export Screening Data
  //

# the class that performs the API call
require_once('RestCallRequest.php');

# arrays to contain elements you want to filter results by
# example: array('item1', 'item2', 'item3');
$records = array();
$events = array();
$fields = array();
$forms = array();

$token = "INSERT YOUR API token here";

# an array containing all the elements that must be submitted to the API
$data = array('content' => 'record', 'type' => 'flat', 'format' => 'csv', 'records' => $records, 'events' => $events,
          'fields' => $fields, 'forms' => $forms, 'token' => $token); //

# create a new API request object
$request = new RestCallRequest("https://rc.cfri.ca/redcap/api/", 'POST', $data);

# initiate the API request
$request->execute();

/*...*/
# OPTION 1: for testing purposes and small datasets you can just output the data to screen

# get the content type of the data being returned
$response = $request->getResponseInfo();
$type = explode(";", $response['content_type']);
$contentType = $type[0];

# set the content type of page
//header("Content-type: $contentType; charset=utf-8");
```

Result : when there is no API token or invalid token in the source code

ERROR: You do not have permissions to use the API

```
RestCallRequest Object
(
    [url:protected] => https://rc.cfri.ca/redcap/api/
    [verb:protected] => POST
    [requestBody:protected] => content=record&type=flat&format=csv&token=INSERT+YOUR+API+token+here
    [requestLength:protected] => 0
    [username:protected] =>
    [password:protected] =>
    [acceptType:protected] => text/xml
    [responseBody:protected] => ERROR: You do not have permissions to use the API
    [responseInfo:protected] => Array
        (
            [url] => https://rc.cfri.ca/redcap/api/
            [content_type] => text/html; charset=utf-8
            [http_code] => 403
            [header_size] => 351
            [request_size] => 198
            [filetime] => -1
            [ssl_verify_result] => 0
            [redirect_count] => 0
            [total_time] => 0.125461
            [namelookup_time] => 0.004569
            [connect_time] => 0.007045
            [pretransfer_time] => 0.035511
            [size_upload] => 68
            [size_download] => 49
            [speed_download] => 390
            [speed_upload] => 542
            [download_content_length] => -1
            [upload_content_length] => 0
            [starttransfer_time] => 0.125357
            [redirect_time] => 0
            [certinfo] => Array
                (
                )

        )

    [usingFiles:protected] =>
)
```

Successful Export window page

```
RestCallRequest Object
(
    [url:protected] => https://rc.cfri.ca/redcap/api/
    [verb:protected] => POST
    [requestBody:protected] => content=record&type=flat&format=csv&token=DF3492792186348F058F8D6176037A28
    [requestLength:protected] => 0
    [username:protected] =>
    [password:protected] =>
    [acceptType:protected] => text/xml
    [responseBody:protected] => record_id,interviewer_code_screen,arrive_household_time,answer_door,no_answer_time_left,no_answer_le
1,,,,,0,0,0,0,0,0,0,,,,,,,,,,,,0,,,,0,,,,,,,,,,0,,,,,,,,,,,0,,0,,,,,,,,,,,,,,,,,,,,,,,,,,1,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

    [responseInfo:protected] => Array
        (
            [url] => https://rc.cfri.ca/redcap/api/
            [content_type] => text/html; charset=utf-8
            [http_code] => 200
            [header_size] => 344
            [request_size] => 204
            [filetime] => -1
            [ssl_verify_result] => 0
            [redirect_count] => 0
            [total_time] => 0.191997
            [namelookup_time] => 0.004591
            [connect_time] => 0.005209
            [pretransfer_time] => 0.034311
            [size_upload] => 74
            [size_download] => 23004
            [speed_download] => 119814
            [speed_upload] => 385
            [download_content_length] => -1
            [upload_content_length] => 0
            [starttransfer_time] => 0.191707
            [redirect_time] => 0
            [certinfo] => Array
                (
                )

        )

    [usingFiles:protected] =>
)
```

Importing Records: PHP code

```php
<?php

  // Example : Importing records through REDCap API
  //

# the class that performs the API call
require_once('RestCallRequest.php');

$input_filename_path = '';

# or OPTION 2: fill the variable with data from a file
$YOUR_DATA = file_get_contents($input_filename_path);

$records = array();
$events = array();
$fields = array();
$forms = array();

$token = "INSERT YOUR API TOKEN here ";

# an array containing all the elements that must be submitted to the API
$data = array('content' => 'record', 'type' => 'flat', 'format' => 'csv', 'records' => $records, 'events' => $events,
          'fields' => $fields, 'forms' => $forms, 'token' => '', 'data' => $YOUR_DATA); // Doug

# create a new API request object

$request = new RestCallRequest("https://rc.cfri.ca/redcap/API/", 'POST', $data);

# initiate the API request
$request->execute();

# the following line will print out the entire HTTP request object
# good for testing purposes to see what is sent back by the API and for debugging
echo '<pre>' . print_r($request, true) . '</pre>';

# print the output from the API
echo $request->getResponseBody();

?>
```

HTTP Status Codes:

The REDCap API attempts to return appropriate HTTP status codes for every request.

- **200 OK:** Success!
- **400 Bad Request:** The request was invalid. An accompanying message will explain why.
- **401 Unauthorized:** API token was missing or incorrect.

- **403 Forbidden:** You do not have permissions to use the API.
- **404 Not Found:** The URI you requested is invalid or the resource does not exist.
- **406 Not Acceptable:** The data being imported was formatted incorrectly.
- **500 Internal Server Error:** The server encountered an error processing your request.
- **501 Not Implemented:** The requested method is not implemented.

API examples

The REDCap API can be called from a variety of clients using any popular client-side or web development language that you are able to implement (e.g .NET, Python, PHP, Java). Below you may download a ZIP file containing several examples of how to call the API using various software languages. The files contained therein may be modified however you wish.

NOTE: The files included in the ZIP file below are *not* officially sanctioned REDCap files but are merely examples of how one might make API requests using specific software languages. Please be aware that the files in the ZIP could potentially change from one REDCap version to the next.

redcap_api_examples.zip

If you need any assistance on how to use REDCap API to connect external application, please contact us at redcap@bcahsn.ca