Omniscien™ TECHNOLOGIES

Language Studio REST API V5.0

Real-Time and Batch Mode Features

31 May 2018

www.omniscien.com
sales@omniscien.com +66 (2) 381 5869

Omniscien Technologies Global HQ 16 Collyer Quay, #22-00, Singapore 049318, Singapore
Omniscien Technologies Europe Stephensonstraat 45, 2723 RM Zoetermeer, The Netherlands
Omniscien Technologies Asia 141 Major Tower Thonglor, 11th Fl., Office 1, Soi Sukhumvit 63 (Ekamai), Sukhumvit Rd., Klongton-Nua, Wattana, Bangkok 10110, Thailand

TABLE OF CONTENTS

Omniscien™ TECHNOLOGIES

Copyright © 2018 Omniscien Technologies. All Rights Reserved

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

1. USING REST API

1.1. Overview

This document describes how to use the common features of the Language Studio REST API version 5 using the RESTful calling style. The result of a request to the Language Studio™ REST API is a simple JSON object.

1.2. How to read this Document

Here is an overview of the following sections:

- Section 2 – How to Access your LSRestAPI Key
  - Details on how to generate or access your LSRestAPI key from Language Studio
- Section 3 – LSRestAPI methods
  - A summary of the LSRestAPI methods
- Section 4 – Job.Submit.Translate
  - Detailed information on how to submit translation
- Section 5 – Job.Submit.Status
  - Detailed information on how to check the status of a job
- Section 6 – Job.Submit.Download
  - Detailed information on how to download a job
- Section 7 – Job.Submit.Delete
  - Detailed information on how to delete a job
- Section 8 – Job.Submit.Cancel
  - Detailed information on how to cancel a job

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

2. HOW TO ACCESS YOUR LSRESTAPI KEY

2.1. Identifying Your Application to Language Studio™

Your application needs to identify itself every time it sends a request to the Language Studio™ REST API, by including an API Key with each request. An API key is a unique identifier that is generated for you by your account manager and can be provided on request. The API key is part of the authorization requirements that are used in conjunction with Account Number. All data in Language Studio™ is transmitted encoded in UTF-8.

To acquire an API key, follow the below steps. More detailed information can be found under Acquired API Key From User Profile.

1. Log into your Language Studio™ account at http://www.languagestudio.com or your own server URL.
2. Select Account>Users from the left side menu.
3. If this is for a new User API, click Add New User, otherwise select the API user account.
4. Click on User Profile and choose the API option from the drop-down list.
5. If this is the first time you have required an API key, or you wish to change API key, click on the Generate API Key button.
6. Copy the API key from the display at the bottom of the form.

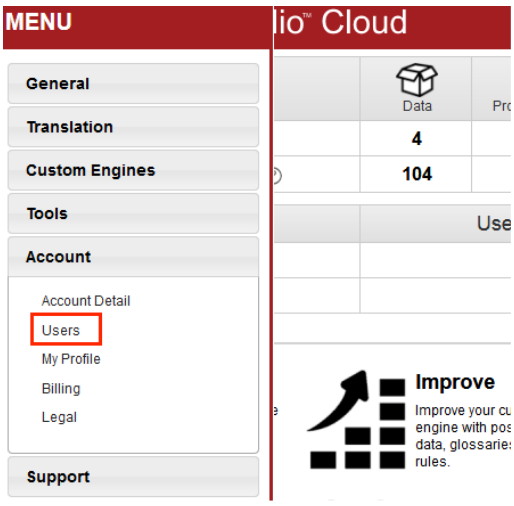
After you have an API key, your application can append the query parameter key=yourAPIKey to all request URLs. The API key is safe for embedding in URLs; it doesn't need any encoding.

The section below shows how the above steps is done in more detail.

2.2. Acquiring API Key from User Profile

To access the API key please follow these steps:

2.2.1. Log into Language Studio and select Users link



2.2.2. Add New User or select existing API user

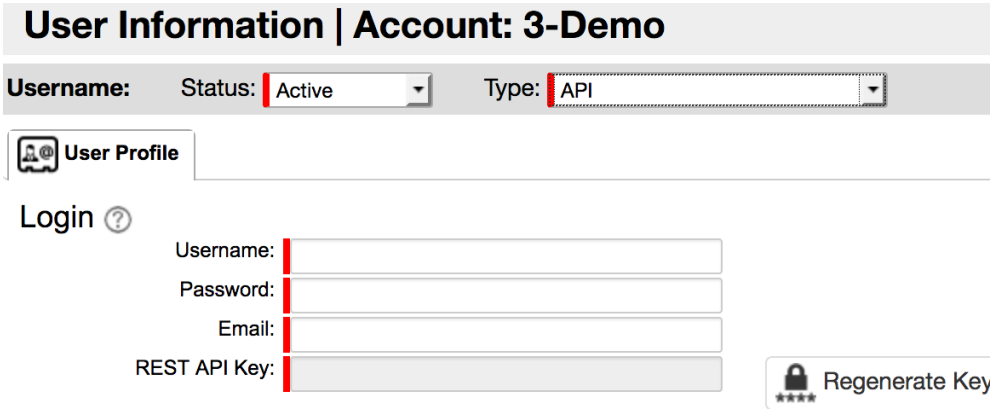
The API user will be clearly labeled as such, under User Type. If one doesn't exist yet, click the **+Add** button.

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

2.2.3. In the User Profile add the credentials.

If a new user. Select **API** in the **Type** field and then add in a **Username**, **Password** and **Email** address.

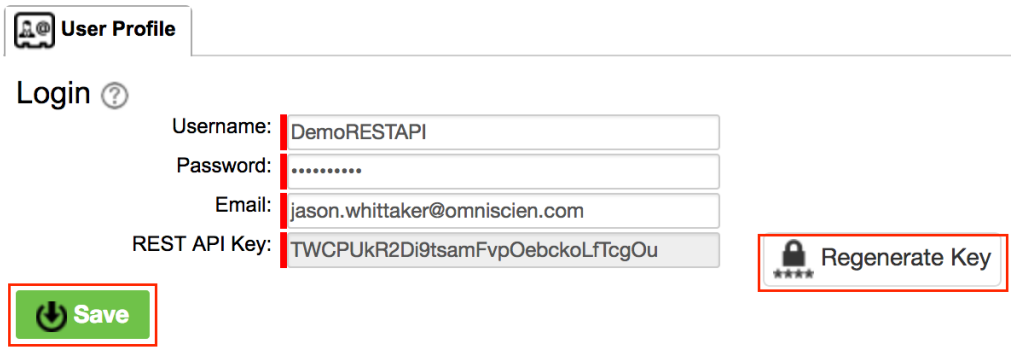
We recommend having API in the Username to distinguish it from a normal user account.



2.2.4. Create a key by clicking Regenerate Key

Press the **Regenerate Key** to produce an API key and then copy the API key from the **REST API Key** field.

Click **Save** to store you're the Rest API key to that user.



Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

3. REST API METHODS FOR TRANSLATION

3.1. API URL

This is the API URL you will need for connecting with Language Studio.

https://lsapi.languagestudio.com/lsrestapi/LSRESTAPI.V5.jsp

3.2. Method Overview

Below is a list of all those methods and what they do.

Method	Comment
Job.Submit	Submit the job(s) to Language Studio™ for processing.
Job.Status	Retrieve all job statuses from Language Studio™.
Job.Download	Download data related to the specified job from Language Studio™.
Job.Cancel	Cancel a job or set of jobs from Language Studio™.
Job.Delete	Delete a job or set of jobs from Language Studio™.

3.3. Translation Request Workflow

3.3.1. Batch Mode – Asynchronous

1. Job.Submit.Translate
    a. Submit a Translation Request, Retrieve Job ID(s) .
2. Job.Status
    a. Check status periodically using Job ID list.
    b. If you have more than 1 job running, you can send all IDs in 1 request.
    c. If you send too many requests in 1 minute, the system will temporarily block you to prevent request flooding. Requests for status should be a maximum of no more than 3 times per minute.
    d. Loop checking status until status is Complete.
3. Job.Download
    a. Download the resulting translation output for each job as it completes.

3.3.2. Real-Time Mode – Synchronous

1. Job.Submit.Translate
    a. Submit a Translation Request.
    b. Request is Synchronous so will not return until translation is complete.
    c. Translation output is in the response.

NOTE: This option requires a dedicated translation server. If you would like to use this option, please contact Omniscien Technologies for more details.

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

4. JOB.SUBMIT.TRANSLATE

4.1. Job.Submit.Translate Overview

Submits a translation job for processing in real-time or batch mode.

4.2. Call

HTTP Method	POST
Content Type	multipart/form-data

4.3. Parameters

Parameter Name	Data Type	Optional/Required	Description
requesttype	String	Required	The type of call being made (case insensitive). This should be set to "**Job.Submit.Translate**".
apikey	String	Required	Your API Key. See **Acquiring API Key from User Profile** for more details.
domaincode	Integer	Required	The domain code of the custom engine to be used for this translation
datatoprocess	String	Required	The data that will be processed. If DataType = 1 (File) A list of file paths delimited by a | (pipe). **Note:** This is only needed in REST if you are submitting files that already exist in the Work Project folder. Use the ***file*** parameter If DataType = 2 (Text) A string to process
projectno	Integer	Optional	The number of Work Project in account to run the job in. (Default value= First project in the account.)
advancedsettingsxml	String	Optional	Contains an XML structure with additional settings. (Default value= Null)
file	File	Optional	Contains 1 or more files that are being uploaded to a Work Project folder. Multiple files Only used to upload files from local storage to a Work Project folder. Use the ***datatoprocess*** parameter to specify files that are already in the Work Project folder. The ***datatoprocess*** parameter will be ignored if files are uploaded. (Default value= Null)

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

Advanced Settings XML Structure

```
<AdvancedSettings-Translate>
  <BatchName></BatchName>
  <ClientMetaData></ClientMetaData>
  <ClientRef></ClientRef>
  <ConfidenceScore></ConfidenceScore>
  <DataType></DataType>
  <DomainVersionNo></DomainVersionNo>
  <GenerateTMX></GenerateTMX>
  <GenerateXLIFF></GenerateXLIFF>
  <JobDetail></JobDetail>
  <JobMode>1</JobMode>
  <JobXML><![CDATA[]]></JobXML>
  <KeepUnknownTags></KeepUnknownTags>
  <LogJobSteps></LogJobSteps>
  <NotifyEmail></NotifyEmail>
  <NotifyURL></NotifyURL>
  <Priority></Priority>
  <ResultDetail></ResultDetail>
  <SentenceSegment></SentenceSegment>
  <TextDataTypeFileExtension>txt</TextDataTypeFileExtension>
  <OutputFiles>
    <File source="" target=""/>
  </OutputFiles>
</AdvancedSettings-Translate>
```

4.3.1. All parameters are optional.

Name	Default Value	Description
BatchName	Null	The name of batch. This is a user specified label that can be used to group files together. It can be very useful in reports.
ClientMetaData	Null	Client meta data that can be used by other applications and carried with the translations. The same data will be applied to each file. Only valid for JobMode=1
ClientRef	Null	Client reference number or ID that can be used to associated data with external applications. Only valid for JobMode=1
ConfidenceScore	False	If True will return confidence scores.
DataType	1	Determines the type of data contained in DataToProcess. Valid values: 1 - File 2 - Text string
DomainVersionNo	The version number specified in the custom engine domain settings.	The version number of the custom engine to use when translating. Only valid for JobMode=1

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

GenerateTMX	False	If True, a separate TMX file will be created when non-TMX documents are submitted for translation. Only valid for JobMode=1
GenerateXLIFF	False	If True, a separate XLIFF file will be created when non-XLIFF documents are submitted for translation.
JobDetail	1	Determines the level of detail for the job when JobMode=2 (Real-Time) 1. Standard 2. Enhanced Only valid for JobMode=2
JobMode	1	The mode to run the job in. Valid values: 1. Batch - Jobs are submitted to the job queue and processed progressively. 2. Real-time - Jobs are processed and a response is immediate. **See notes below. Notes: • Diagnostic Review jobs can only be run from the Web Interface. • Real-time jobs require infrastructure deployed 24/7. This option is only available if you have made prior arrangements for real-time infrastructure. • Jobs will error if this Domain Code is not configured for real-time access on the server and real-time is specified.
JobXML	Null	Additional custom advanced configuration settings. Talk to your Omniscien Technologies account manager for specific non-standard settings.
LogJobSteps	False	If True then an additional log file will be produced with detailed step by step information that is useful for debugging. If JobMode = 1 (Batch) A detailed log file is produced. If JobMode = 2 (Real-Time) A summary is produced in the JSON format.
NotifyEmail	Null	When specified will send an email on job completion or error. 3 values will be included in the email subject: • JobID • JobType • Status Only valid for JobMode=1
NotifyURL	Null	When specified will connect to the specified URL on job completion or error. 3 values will be included on the QueryString: • JobID • JobType • Status Only valid for JobMode=1

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

Priority	99	The value of priority of this job (Requires permissions) Standard jobs run at priority 99. Jobs with a smaller number will take priority. Jobs with a higher number will be lower priority. Only valid for JobMode=1
ResultDetail	1	Determines the level of detail for results when JobMode=2 (Real-Time). 1. Translation text only 2. Translation text + Segment text (source and target) 3. Translation text + Segment text (source and target) + Word Alignment Only valid for JobMode=2
OutputFiles	Null	Determines the file names of the output files when written to the project folder. A list of files with their paired source and target name. If not defined, then the file will be saved in the project folder with the naming convention: <filename>.out.<fileextension> Only valid for JobMode=1
KeepUnknownTags	False (remove unknown tags)	If True, XML tags <unknown>word</unknown> will be kept in the output and not removed.
SentenceSegment	True	(Optional) Set "true" = Run segment the sentence before translating.
TextDataTypeFileExtension	txt	If DataType = 2 (Text String), this value will be used to determine how the text will be processed.

4.4. Response

String	JSON Response with job submission information.

4.5. Comments

Translation jobs can be submitted as either batch-mode (default) or real-time.

Batch-Mode:

- One or more jobs can be submitted in a single request.
- Jobs will be queued and translated as soon as resources are available.
- Multiple concurrent jobs can translate at the same time.
- This is the most common option and is suitable for all kinds of translation except those that require a real-time response (i.e. chat).
- To check job status, use the REST API Job.Status.
- To download a complete job, use the REST API Job.Download.
- Batch-mode may have a slight delay on the first document as a result of the time needed to load an engine on demand.

Real-Time:

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

- This option requires a real-time server to be allocated and always be on so that it can respond immediately without loading an engine. **Please discuss this with Omniscien Technologies first.**
- Only one request for real-time translation can be submitted per request.
- A request can be multiple sentences, but in general should not be a large number of sentences as this feature is designed for use cases like chat rather than documents.
- A response will be returned immediately.

4.6. Examples

4.6.1. Example Requests

4.6.1.1. Example 1 - Submission of a single job in Batch mode

CURL

```
curl
-H "Content-Type: multipart/form-data"
-X POST
-F "requesttype=Job.Submit.Translate"
-F "apikey=Your_API_KEY"
-F "domaincode=1234"
-F "datatoprocess="
-F "file=@/path/to/file/TH-EN-1.txt"
 https://<SERVER URL>
```

Query String

```
Not Supported
```

Java

```
private void JobSubmitTranslate(String APIKey, String DomainCode, String
DataToProcess) {
        org.apache.http.client.HttpClient client = new
org.apache.http.impl.client.DefaultHttpClient();
        org.apache.http.client.methods.HttpPost post = new
org.apache.http.client.methods.HttpPost(https://<SERVER URL>);
        org.apache.http.entity.mime.MultipartEntity entity = new
org.apache.http.entity.mime.MultipartEntity();
        java.nio.charset.Charset chars = java.nio.charset.Charset.forName("UTF-8");
        // Set normal parameter
        org.apache.http.entity.mime.content.StringBody param1 = new
org.apache.http.entity.mime.content.StringBody("Job.Submit.Translate", chars);
        entity.addPart("requesttype", param1);

        org.apache.http.entity.mime.content.StringBody param2 = new
org.apache.http.entity.mime.content.StringBody(APIKey, chars);
        entity.addPart("apikey", param2);

        org.apache.http.entity.mime.content.StringBody param3 = new
org.apache.http.entity.mime.content.StringBody(DomainCode, chars);
        entity.addPart("domaincode", param3);

        // Set file to upload
        java.io.File oFile = new java.io.File(DataToProcess);
        if (oFile.exists()) {
                entity.addPart("file", new
org.apache.http.entity.mime.content.FileBody(oFile));
        }
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
        //Set entity
        post.setEntity(entity);

        // Post parameters to a server
        org.apache.http.HttpResponse response = client.execute(post);

        // Get response text
        org.apache.http.HttpEntity entityRes = response.getEntity();
        String returnValue = org.apache.http.util.EntityUtils.toString(entityRes,
"UTF-8");
        System.out.println("Output=" + returnValue);
}
```

4.6.1.2. Example 2 - Submission of a multiple jobs in Batch mode

CURL

```
curl
-H "Content-Type: multipart/form-data"
-X POST
-F "requesttype=Job.Submit.Translate"
-F "apikey=Your_API_KEY"
-F "domaincode=1234"
-F "datatoprocess="
-F "file=@/path/to/file/TH-EN-1.txt"
-F "file=@/path/to/file/TH-EN-2.txt"
https://<SERVER URL>
```

Query String

```
Not supported
```

Java

```
private void JobSubmitTranslate(String APIKey, String DomainCode, String
DataToProcess) {
        org.apache.http.client.HttpClient client = new
org.apache.http.impl.client.DefaultHttpClient();
        org.apache.http.client.methods.HttpPost post = new
org.apache.http.client.methods.HttpPost(https://<SERVER URL>);
        org.apache.http.entity.mime.MultipartEntity entity = new
org.apache.http.entity.mime.MultipartEntity();
        java.nio.charset.Charset chars = java.nio.charset.Charset.forName("UTF-8");

        // Set normal parameter
        org.apache.http.entity.mime.content.StringBody param1 = new
org.apache.http.entity.mime.content.StringBody("Job.Submit.Translate", chars);
        entity.addPart("requesttype", param1);

        org.apache.http.entity.mime.content.StringBody param2 = new
org.apache.http.entity.mime.content.StringBody(APIKey, chars);
        entity.addPart("apikey", param2);

        org.apache.http.entity.mime.content.StringBody param3 = new
org.apache.http.entity.mime.content.StringBody(DomainCode, chars);
        entity.addPart("domaincode", param3);

        // Set file to upload
        String[] Files = DataToProcess.split("[|]");
        for (String sFilePath : Files) {
                java.io.File oFile = new java.io.File(sFilePath);
                if (oFile.exists()) {
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
                        entity.addPart("file", new
org.apache.http.entity.mime.content.FileBody(oFile));
                }
        }

        // Set entity
        post.setEntity(entity);

        // Post parameters to a server
        org.apache.http.HttpResponse response = client.execute(post);

        // Get response text
        org.apache.http.HttpEntity entityRes = response.getEntity();
        String returnValue = org.apache.http.util.EntityUtils.toString(entityRes,
"UTF-8");
        System.out.println("Output=" + returnValue);
}
```

4.6.1.3. Example 3 - Submission of a job in Real-Time mode

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Submit.Translate"
-d "apikey=Your_API_KEY"
-d "domaincode=1234"
-d "datatoprocess=This is the first sentence. This is the second sentence."
-d "advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode></AdvancedSettings-Translate>"
https://<SERVER URL>
```

Query String

```
GET https://<SERVER
URL>?requesttype=Job.Submit.Translate&apikey=YOUR_API_KEY&domaincode=YOUR_DOMAIN_COD
E&datatoprocess=This is the first sentence. This is the second
sentence.&advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode></AdvancedSettings-Translate>
```

Java

```
private void JobSubmitTranslate (String APIKey, String DomainCode, String
DataToProcess, String AdvancedSettingsXML)  {
        // set parameters
        String sParameters = "requesttype=Job.Submit.Translate&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&domaincode=" +
java.net.URLEncoder.encode(DomainCode, "UTF-8") + "&datatoprocess=" +
java.net.URLEncoder.encode(DataToProcess, "UTF-8") + "&advancedsettingsxml=" +
java.net.URLEncoder.encode(AdvancedSettingsXML, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
        StringBuilder output = new StringBuilder();
        java.io.OutputStreamWriter writer = null;
        java.io.BufferedReader reader = null;
        java.io.InputStreamReader isReader = null;
        try {
                // Send data
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
                java.net.URL url = new java.net.URL(webUrl);
                java.net.URLConnection conn = url.openConnection();
                conn.setDoOutput(true);
                writer = new java.io.OutputStreamWriter(conn.getOutputStream());
                writer.write(parameters);
                writer.flush();
                // Get the response
                isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-
8");
                reader = new java.io.BufferedReader(isReader);
                String line;
                while ((line = reader.readLine()) != null) {
                        output.append(line);
                }
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        } catch (Exception e) {
        e.printStackTrace();
        } finally {
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        }
    return output.toString();
}
```

4.6.1.4. Example 4 - Submission of a job in Real-Time mode with ResultDetail=2 and JobDetail=2

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Submit.Translate"
-d "apikey=Your_API_KEY"
-d "domaincode=1234"
-d "datatoprocess=This is a test. This is the second sentence."
-d "advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode><ConfidenceScore>true</Confidenc
eScore><JobDetail>2</JobDetail><ResultDetail>2</ResultDetail></AdvancedSettings-
Translate>"
https://<SERVER URL>
```

Query String

```
POST https://<SERVER
URL>?requesttype=Job.Submit.Translate&apikey=YOUR_API_KEY&domaincode=1234&datatoproc
ess=This is the first sentence. This is the second
sentence.&advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode><ConfidenceScore>true</Confidenc
eScore><JobDetail>2</JobDetail><ResultDetail>2</ResultDetail></AdvancedSettings-
Translate>
```

Java

```
private void JobSubmitTranslate (String APIKey, String DomainCode, String
DataToProcess, String AdvancedSettingsXML)  {
        // set parameters
        String sParameters = "requesttype=Job.Submit.Translate&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&domaincode=" +
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
java.net.URLEncoder.encode(DomainCode, "UTF-8") + "&datatoprocess=" +
java.net.URLEncoder.encode(DataToProcess, "UTF-8") + "&advancedsettingsxml=" +
java.net.URLEncoder.encode(AdvancedSettingsXML, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
        StringBuilder output = new StringBuilder();
        java.io.OutputStreamWriter writer = null;
        java.io.BufferedReader reader = null;
        java.io.InputStreamReader isReader = null;
        try {
                // Send data
                java.net.URL url = new java.net.URL(webUrl);
                java.net.URLConnection conn = url.openConnection();
                conn.setDoOutput(true);
                writer = new java.io.OutputStreamWriter(conn.getOutputStream());
                writer.write(parameters);
                writer.flush();
                // Get the response
                isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-
8");
                reader = new java.io.BufferedReader(isReader);
                String line;
                while ((line = reader.readLine()) != null) {
                        output.append(line);
                }
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        } catch (Exception e) {
        e.printStackTrace();
        } finally {
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        }
    return output.toString();
}
```

4.6.1.5. Example 5 - Submission of a job in Real-Time mode with ResultDetail=3 and JobDetail=2

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Submit.Translate"
-d "apikey=Your_API_KEY"
-d "domaincode=1234"
-d "datatoprocess=This is the first sentence. This is the second sentence."
-d "advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode><ConfidenceScore>true</Confidenc
eScore><JobDetail>2</JobDetail><ResultDetail>3</ResultDetail></AdvancedSettings-
Translate>"
https://<SERVER URL>
```

Query String

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
POST https://<SERVER
URL>?requesttype=Job.Submit.Translate&apikey=YOUR_API_KEY&domaincode=1234&datatoproc
ess=This is the first sentence. This is the second
sentence.&advancedsettingsxml=<AdvancedSettings-
Translate><DataType>2</DataType><JobMode>2</JobMode><ConfidenceScore>true</Confidenc
eScore><JobDetail>2</JobDetail><ResultDetail>3</ResultDetail></AdvancedSettings-
Translate>
```

Java

```
private void JobSubmitTranslate (String APIKey, String DomainCode, String
DataToProcess, String AdvancedSettingsXML)  {
        // set parameters
        String sParameters = "requesttype=Job.Submit.Translate&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&domaincode=" +
java.net.URLEncoder.encode(DomainCode, "UTF-8") + "&datatoprocess=" +
java.net.URLEncoder.encode(DataToProcess, "UTF-8") + "&advancedsettingsxml=" +
java.net.URLEncoder.encode(AdvancedSettingsXML, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
        StringBuilder output = new StringBuilder();
        java.io.OutputStreamWriter writer = null;
        java.io.BufferedReader reader = null;
        java.io.InputStreamReader isReader = null;
        try {
                // Send data
                java.net.URL url = new java.net.URL(webUrl);
                java.net.URLConnection conn = url.openConnection();
                conn.setDoOutput(true);
                writer = new java.io.OutputStreamWriter(conn.getOutputStream());
                writer.write(parameters);
                writer.flush();
                // Get the response
                isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-
8");
                reader = new java.io.BufferedReader(isReader);
                String line;
                while ((line = reader.readLine()) != null) {
                        output.append(line);
                }
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        } catch (Exception e) {
        e.printStackTrace();
        } finally {
                if (reader != null) { reader.close(); }
                if (isReader != null) { isReader.close();}
                if (writer != null) { writer.close();}
        }
    return output.toString();
}
```

4.6.2. Example Responses

4.6.2.1. Example 1 - Submission of a single job in Batch mode

```
{
  "result": {
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
    "jobs": [
      {
        "job": {
          "file":"TH-EN-1.txt",
          "status": "1",
          "description": "Job successfully added to queue.",
          "jobid": "191260"
        }
      }
    ]
  },
  "duration": "134",
  "startdate": "20171221 08:37:33:934",
  "errortext": "",
  "errorno": "",
  "enddate": "20171221 08:37:34:068",
  "requestid": "20171221-083733.378-105.172.000001"
}
```

4.6.2.2. Example 2 - Submission of a multiple jobs in Batch mode

```
{
  "result": {
    "jobs": [
      {
        "job": {
          "file": "TH-EN-1.txt",
          "status": "1",
          "description": "Job successfully added to queue.",
          "jobid": "191260"
        }
      },
      {
        "job": {
          "file": "TH-EN-2.txt",
          "status": "0",
          "description": "Duplicate job in queue.",
          "jobid": ""
        }
      }
    ]
  },
  "duration": "134",
  "startdate": "20171221 08:37:33:934",
  "errortext": "",
  "errorno": "",
  "enddate": "20171221 08:37:34:068",
  "requestid": "20171221-083733.378-105.172.000001"
}
```

4.6.2.3. Example 3 - Submission of a job in Real-Time mode

```
{
  "result": {
    "jobs": [
      {
        "job": {
          "status": "1",
          "description": "Job successfully translated.",
          "jobid": "195198",
          "resulttext": "Ini adalah kalimat pertama. Ini adalah kalimat kedua."
        }
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
      }
    ]
  },
  "duration": "68417",
  "startdate": "20180418 09:43:53:428",
  "errortext": "",
  "errorno": "",
  "enddate": "20180418 09:45:01:845",
  "requestid": "20180418-094352.807-105.172.000001"
}
```


4.6.2.4. Example 4 - Submission of a job in Real-Time mode with ResultDetail=2 and JobDetail=2


```
{
  "result": {
    "jobs": [
      {
        "job": {
          "status": "1",
          "description": "Job successfully translated.",
          "resultsegments": [
            {
              "segment": {
                "source": "This is the first sentence.",
                "target": "ini kalimat pertama .",
                "confidencescore": "",
                "status": "1"
              }
            },
            {
              "segment": {
                "source": "This is the second sentence.",
                "target": "ini kalimat kedua .",
                "confidencescore": "",
                "status": "1"
              }
            }
          ],
          "jobid": "195507",
          "resulttext": "Ini kalimat pertama. Ini kalimat kedua."
        }
      }
    ]
  },
  "duration": "58233",
  "startdate": "20180419 13:38:34:114",
  "errortext": "",
  "errorno": "",
  "enddate": "20180419 13:39:32:347",
  "requestid": "20180419-133833.452-105.104.000001"
}
```


4.6.2.5. Example 5 - Submission of a job in Real-Time mode with ResultDetail=3 and JobDetail=2


```
{
  "result": {
    "jobs": [
      {
        "job": {
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
          "status": "1",
          "description": "Job successfully translated.",
          "resultsegments": [
            {
              "segment": {
                "source": "This is the first sentence.",
                "target": "ini |0-1,0, 0-0 1-0| kalimat pertama |2-4,0, 0-1 1-1 2-0|
. |5-5,0, 0-0| ",
                "confidencescore": "",
                "status": "1"
              }
            },
            {
              "segment": {
                "source": "This is the second sentence.",
                "target": "ini |0-1,0, 0-0 1-0| kalimat kedua |2-4,0, 0-1 1-1 2-0| .
|5-5,0, 0-0| ",
                "confidencescore": "",
                "status": "1"
              }
            }
          ],
          "jobid": "195504",
          "resulttext": "Ini kalimat pertama. Ini kalimat kedua."
        }
      }
    ]
  },
  "duration": "58876",
  "startdate": "20180419 12:52:11:258",
  "errortext": "",
  "errorno": "",
  "enddate": "20180419 12:53:10:134",
  "requestid": "20180419-125210.613-105.104.000001"
```


Omniscien™ TECHNOLOGIES

Copyright © 2018 Omniscien Technologies. All Rights Reserved

5. JOB.STATUS

5.1. Job.Status Overview

Returns the job status of one or more Job IDs.

5.2. Call

HTTP Method	GET
Content Type	text/html

5.3. Parameters

Name	Default Value	Description
requesttype	String	The type of call being made (case insensitive). This should be set to "**Job.Status**".
apikey	String	Your API Key. See **Acquiring API Key from User Profile** for more details.
jobidlist	String	A list of Job IDs delimited by a comma (e.g. "1234,1236,1237")
detaillevel	Integer	(Optional Default=1) The level of detail to be returned from the request. 1=Basic (status only) 2=Detailed Information about the job. 3=Extended Detailed Information about the job.

5.4. Response

String	JSON Response with job status information.

5.5. Comments

Different detail levels can be returned by specifying the detail level parameter.

In each job, a type parameter is specified which specifies if the job is a **JSJob** or a **Translation**.

5.6. Examples

5.6.1. Example Requests

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Status"
-d "apikey=Your_API_KEY"
-d "jobidlist=123"
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
-d "detaillevel=1"
https://<SERVER URL>
```

Query String

```
GET https://<SERVER
URL>?requesttype=Job.Status&apikey=YOUR_API_KEY&jobidlist=123&detaillevel=1
```

Java

```
private void JobStatus(String APIKey, String JobIDList, String DetailLevel)  {
        // set parameters
        String sParameters = "requesttype=Job.Status&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&detaillevel=" +
java.net.URLEncoder.encode(DetailLevel, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
    StringBuilder output = new StringBuilder();
    java.io.OutputStreamWriter writer = null;
    java.io.BufferedReader reader = null;
    java.io.InputStreamReader isReader = null;
    try {
        // Send data
        java.net.URL url = new java.net.URL(webUrl);
        java.net.URLConnection conn = url.openConnection();
        conn.setDoOutput(true);
        writer = new java.io.OutputStreamWriter(conn.getOutputStream());
        writer.write(parameters);
        writer.flush();
        // Get the response
        isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-8");
        reader = new java.io.BufferedReader(isReader);
        String line;
        while ((line = reader.readLine()) != null) {
            output.append(line);
        }
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    }
    return output.toString();
}
```

5.6.2. Example Responses

DetailLevel = 1

```
{
   "result":{
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
      "jobs":[
         {
            "job":{
               "jobid":"123",
               "type":"translation",
               "status":"queued",
               "statuscode":"9"
            }
         }
      ]
   },
   "requestid":"20170314-092048.406-1.1.000001",
   "duration":"56057",
   "startdate":"20170314 13:48:07.534",
   "enddate":"20170314 13:49:03.591",
   "errortext":"",
   "errorno":""
}
```


DetailLevel = 2


```
{
  "result": {
    "jobs": [
      {
        "job": {
          "header": {
            "jobid": "123",
            "type": "translation",
            "status": "Complete",
            "statuscode": "2",
            "jobtypecode": "2",
            "jobtypename": "plaintext",
            "projectno": "8381",
            "languagepairname": "EN-ZH",
            "domaincode": "1301971",
            "batchname": "",
            "priority": "99",
            "inputfilename": "TESTFile_019_11.txt",
            "outputfilename": "TESTFile_019_11_1301971.out.txt"
          },
          "downloads": [
            {
              "download": {
                "date": "20171127 13:00:20:070",
                "accountno": "1",
                "user": "TestAPI",
                "filetypes": "TX|LOG|SRC"
              }
            }
          ],
          "jobactivity": {
            "datecreated": "2017-11-22 08:01:07.000",
            "dateassigned": "2017-11-22 08:01:07.000",
            "datelastupdated": "2017-11-22 08:01:27.000",
            "durationqueue:": "123454",
            "durationprocessing": "12345",
            "durationtotal": "12345",
            "createdby": "TestAPI",
            "updatedby": "TestAPI",
            "processing": {
              "totalsteps": "27",
              "tucount": "2",
              "tuwarnings": "",
              "sourcewordscount": "4",
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
              "targetwordscount": "4",
              "currentstep": {
                "no": "27",
                "name": "js12",
                "title": "Postprocess Custom - Postprocess Javascript (JS12) ",
                "message": "The job has been successfully completed.",
                "durationprocessing": "",
                "timeremaining": "",
                "percentcomplete": "100",
                "tucomplete": "",
                "wordscomplete": "13",
                "wordsperminute": "156000"
              }
            }
          }
        }
      }
    ]
  },
  "requestid": "20170314-115809.298-105.172.000001",
  "duration": "56057",
  "startdate": "20170314 13:48:07.534",
  "enddate": "20170314 13:49:03.591",
  "errortext": "",
  "errorno": ""
}
```


Detail Level = 3


```
{
  "result": {
    "jobs": [
      {
        "job": {
          "header": {
            "jobid": "123",
            "type": "translation",
            "status": "Complete",
            "statuscode": "2",
            "jobtypecode": "2",
            "jobtypename": "plaintext",
            "projectno": "8381",
            "languagepairname": "EN-ZH",
            "domaincode": "1301971",
            "batchname": "",
            "priority": "99",
            "inputfilename": "TESTFile_019_11.txt",
            "outputfilename": "TESTFile_019_11_1301971.out.txt",
            "inputfilepath": "00001/08381/TESTFile_019_11.txt",
            "inputfiledesc": "",
            "outputfilepath": "00001/08381/TESTFile_019_11_1301971.out.txt",
            "domainname": "General",
            "domainversionno": "1",
            "capitalizeversionno": "",
            "jobtypedesc": "Translate Plain Text",
            "projectname": "P-DropFolder",
            "jobxml":
"<jobxml><jobflags><KeepUnknownTags>false</KeepUnknownTags><SentenceSegment>true</Se
ntenceSegment><GenerateXLIFF>false</GenerateXLIFF><GenerateTMX>false</GenerateTMX><L
ogJobSteps>false</LogJobSteps></jobflags></jobxml>",
            "languagepaircode": "1",
            "decodertype": "SMT3",
            "nmtdecodertype": "2",
            "clientref": "",
            "clientmetadata": ""
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
    },
    "downloads": [
      {
        "download": {
          "date": "20171127 13:00:20:070",
          "accountno": "1",
          "user": "TestAPI",
          "filetypes": "TX"
        }
      }
    ],
    "jobactivity": {
      "datecreated": "20171122 08:01:07.000",
      "dateassigned": "20171122 08:01:07.000",
      "datelastupdated": "20171122 08:01:27.000",
      "durationqueue:": "123454",
      "durationprocessing": "12345",
      "durationtotal": "12345",
      "createdby": "TestAPI",
      "updatedby": "TestAPI",
      "processing": {
        "totalsteps": "27",
        "tucount": "2",
        "tuwarnings": "",
        "sourcewordscount": "4",
        "targetwordscount": "4",
        "currentstep": {
          "no": "27",
          "name": "js12",
          "title": "Postprocess Custom - Postprocess Javascript (JS12) ",
          "message": "",
          "durationprocessing": "",
          "timeremaining": "",
          "percentcomplete": "100",
          "tucomplete": "",
          "wordscomplete": "13",
          "wordsperminute": "156000"
        }
      },
      "unknownwords": {
        "resolve": "Google|Bing|None",
        "unknowncount": "",
        "unknownuniquecount": "",
                  "resolvedcount": ""
      },
      "rules": {
        "basic": "pta,glo,ntt,ptc",
        "advanced": "js2,js7,js9"
      },
      "steptime": {
        "01:gjp": "20171127 12:57:45:594,20171127 12:57:45:688,94",
        "03:rts": "20171127 12:57:45:736,20171127 12:57:45:742,6",
        "05:etu": "20171127 12:57:45:783,20171127 12:57:48:066,02283",
        "13:tok": "20171127 12:57:48:156,20171127 12:57:55:490,7334",
        "15:etm": "20171127 12:57:55:554,20171127 12:57:55:952,398",
        "16:tx": "20171127 12:57:56:058,20171127 12:58:50:877,819",
        "17:txu": "20171127 12:58:51:015,20171127 12:58:51:015,0",
        "19:cap": "20171127 12:58:51:055,20171127 13:00:17:706,651",
        "21:rim": "20171127 13:00:17:741,20171127 13:00:17:869,128",
        "22:det": "20171127 13:00:17:903,20171127 13:00:19:941,2:038",
        "26:mrg": "20171127 13:00:20:070,20171127 13:00:20:186,116"
      }
    }
  }
}
```


Omniscien™ TECHNOLOGIES

Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
  ]
},
"requestid": "20170314-092048.406-1.1.000001",
"duration": "56057",
"startdate": "20170314 13:48:07.534",
"enddate": "20170314 13:49:03.591",
"errortext": "",
"errorno": ""
```


Copyright © 2018 Omniscien Technologies. All Rights Reserved

6. JOB.DOWNLOAD

6.1. Job.Download Overview

Downloads the files or download history from one or more jobs.

6.2. Call

HTTP Method	GET
Content Type	text/html

6.3. Parameters

Name	Default Value	Description
requesttype	String	The type of call being made (case insensitive). This should be set to "**Job.Download**".
apikey	String	Your API Key. See **Acquiring API Key from User Profile** for more details.
jobidlist	String	(Optional. Default=Null) A list of Job IDs delimited by a comma (e.g. "1234,1236,1237") to delete. All files associated with each job will be located and deleted.
filetypes	String	(Optional. Default = "TX") A list of file types to download, delimited by a pipe (|) Valid Options: • TX = Translation • SRC = Source • LOG = Debug Log File • TMX = TMX file if available • XLF = XLIFF file if available
zipformat	String	(Optional. Default="ZIP") Valid Values: • NONE - Only valid for single file. • ZIP - Standard ZIP Algorithm (See https://en.wikipedia.org/wiki/Zip_(file_format)) • 7Z - 7-Zip provides a notably higher compression ratio, but takes longer to compress/decompress. This is useful for very large sets of files when download bandwidth is limited. A smaller file will result in faster downloads. (See https://en.wikipedia.org/wiki/7-Zip) • GZ - GNU Zip Format (See https://en.wikipedia.org/wiki/Gzip)
downloadkey	String	(Optional. Default=Null) A key to use to download package.
downloadhistory	Boolean	(Optional. Default = False) If True, all other arguments except for JobIDList will be ignored.

Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

6.4. Responses

There are 3 possible Responses:

JSON - Job File Data

String	If the DownloadKey is not specified and more than 1 file is requested, then the JSON response showing the data that is to be returned

JSON - Job Download History

String	If the History parameter is specified, then the download history for each job will be returned in JSON format.

Binary File Download

Binary	In the case of passing the DownloadKey or a single file is requested, then the binary file will be returned that matches the key.

6.5. Comments

This API is designed for downloading translation job outputs and history only. Other non-translation job types have different download mechanisms.

This API has multiple functions:

1. Download a single file directly
    1. Specify a single JobID in the JobIDList and a single value in FileTypes.
    2. The file will be returned immediately as a binary response.
2. Prepare to download
    1. Specify the ZipFormat and one or more JobIDs in JobIDList and one or more file types in FileTypes
    2. The files will be prepared and zipped in the format specified.
    3. On completion of zipping, a JSON response will be sent with the details of the files in that are in the Zip file and a DownloadKey value. Use the key to download the Zip.
3. Download
    1. Specify the DownloadKey
    2. A zip file containing all files requested will be returned in the previously specified Zip format.
4. Download Job Download History
    1. If Download history is True, then the JSON response will contain data for the download history of the job.

6.6. Examples

6.6.1. Example Requests

6.6.1.1. Example 1 - Download a single file directly

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
-d "requesttype=Job.Download"
-d "apikey=Your_API_KEY"
-d "jobidlist=191224"
-d "filetypes=TX"
https://<SERVER URL>
```

Query String

```
GET https://<SERVER
URL>?requesttype=Job.Download&apikey=YOUR_API_KEY&jobidlist=191224&filetypes=TX
```

Java

```
private void JobDownload(String APIKey, String JobIDList, String FileTypes, String
SaveFilePath)  {
    // set parameters
    String sParameters = "requesttype=Job.Download&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&filetypes=" +
java.net.URLEncoder.encode(FileTypes, "UTF-8");
        sParameters = "https://<SERVER URL>?" + sParameters;
        //call RestAPI and save file
        java.net.URL oURL = new java.net.URL(sParameters);
        java.io.File fSavePath = new java.io.File(SaveFilePath);
        org.apache.commons.io.FileUtils.copyURLToFile(oURL, fSavePath);
}
```

6.6.1.2. Example 1 - Download a single file directly

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Download"
-d "apikey=Your_API_KEY"
-d "jobidlist=191224"
-d "filetypes=TX|LOG"
-d "zipformat=ZIP"
 https://<SERVER URL>
```

Query String

```
GET https://<SERVER
URL>?requesttype=Job.Download&apikey=YOUR_API_KEY&jobidlist=191224&filetypes=TX|LOG&
zipformat=ZIP
```

Java

```
private void JobDownload(String APIKey, String JobIDList, String FileTypes, String
ZipFormat)  {
    // set parameters
    String sParameters = "requesttype=Job.Download&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&filetypes=" +
java.net.URLEncoder.encode(FileTypes, "UTF-8") + "&zipformat=" +
java.net.URLEncoder.encode(ZipFormat, "UTF-8");
    //call RestAPI and receive output
    String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

6.6.1.3. Example 3 - Download Package

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Download"
-d "apikey=Your_API_KEY"
-d "downloadkey=9c720efd-bcfe-4ec3-a50a-c0f780341814"
https://<SERVER URL>
```

Query String

```
GET https://<SERVER
URL>?requesttype=Job.Download&apikey=YOUR_API_KEY&downloadkey=9c720efd-bcfe-4ec3-
a50a-c0f780341814
```

Java

```
private void JobDownload(String APIKey, String DownloadKey, String SaveFilePath)  {
    // set parameters
    String sParameters = "requesttype=Job.Download&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&downloadkey=" +
java.net.URLEncoder.encode(DownloadKey, "UTF-8");
        sParameters = "https://<SERVER URL>?" + sParameters;
        //call RestAPI and save file
        java.net.URL oURL = new java.net.URL(sParameters);
        java.io.File fSavePath = new java.io.File(SaveFilePath);
        org.apache.commons.io.FileUtils.copyURLToFile(oURL, fSavePath);
}
```

6.6.1.4. Example 4 - Download Job Download History

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Download"
-d "apikey=Your_API_KEY"
-d "jobidlist=191224"
-d "downloadhistory=true"
https://<SERVER URL>
```

Query String

GET https://<SERVER URL>?requesttype=Job.Download&apikey=YOUR_API_KEY&jobidlist=191224&downloadhistory=true

Java

```
private void JobDownload(String APIKey, String JobIDList, String DownloadHistory)  {
    // set parameters
    String sParameters = "requesttype=Job.Download&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&downloadhistory=" +
java.net.URLEncoder.encode(DownloadHistory, "UTF-8");
        //call RestAPI and receive output
    String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
}
```

6.6.2. Example Responses

6.6.2.1. Example 1 - Download a single file directly

This is a binary response

6.6.2.2. Example 2 - Prepare package to download


```
{
  "result": {
    "downloadkey": "9c720efd-bcfe-4ec3-a50a-c0f780341814",
    "downloadkeyexpires": "20180109 18:31:41:809",
    "jobcount": "1",
    "jobs": [
      {
        "job": {
          "tmx": "",
          "translation": "/home/data/lse-data/account/00001/08388/Case7-2.out2.txt",
          "source": "",
          "status": "1",
          "description": "Request job download successfully.",
          "log": "/home/data/lse-data/account/00001/08388/logs/191224.txt.log",
          "jobid": "191224",
          "xlf": ""
        }
      }
    ]
  },
  "duration": "3137",
  "startdate": "20180109 18:01:39:713",
  "errortext": "",
  "errorno": "",
  "enddate": "20180109 18:01:42:850",
  "requestid": "20180109-180135.834-1.1.000001"
}
```


6.6.2.3. Example 3 - Download Package

This is a binary response

6.6.2.4. Example 4 - Download Job Download History


```
{
  "result": {
    "jobs": [
      {
        "job": {
          "jobid": "191224",
          "downloads": [
            {
              "download": {
                "date": "20180105 16:01:47:065",
                "accountno": "1",
                "user": "aoweb",
                "filetypes": "TX"
              }
            },
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
            {
              "download": {
                "date": "20180109 18:11:04:963",
                "accountno": "1",
                "user": "aoweb",
                "filetypes": "TX"
              }
            }
          ]
        }
      }
    ]
},
"duration": "4246",
"startdate": "20180109 18:18:38:951",
"errortext": "",
"errorno": "",
"enddate": "20180109 18:18:43:197",
"requestid": "20180109-181833.002-1.1.000001"
```


Copyright © 2018 Omniscien Technologies. All Rights Reserved

7. JOB.DELETE

7.1. Job.Delete Overview

Deletes all files associated with a job or jobs, with an option to securely delete files using the methods defined by the United States Department of Defense (DoD 5220.22-M) Data Wipe Standard.

7.2. Call

HTTP Method	GET
Content Type	text/html

7.3. Parameters

Name	Default Value	Description
requesttype	String	The type of call being made (case insensitive). This should be set to "**Job.Delete**".
apikey	String	Your API Key. See **Acquiring API Key from User Profile** for more details.
jobidlist	String	A list of Job IDs delimited by a comma (e.g. "1234,1236,1237") to delete. All files associated with each job will be located and deleted.
shred	Boolean	(Optional. Default=False) If True, will shred all files using the Dod 5220.22-M methodology for secure deletion.
shredoverwrites	Integer	(Optional Default=5) The number of overwrites when shredding a document.

7.4. Response

String	JSON Response with confirmation that the files in each job that were deleted or shredded.

7.5. Comments

In general, Shred should not be needed. All files are deleted permanently. However, should your process require that you confirm to security standards issues by the US Department of Defence, then this option can be used to ensure compliance.

7.6. Examples

7.6.1. Example Request

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
-d "requesttype=Job.Delete"
-d "apikey=Your_API_KEY"
-d "jobidlist=123"
https://<SERVER URL>
```


Omniscien TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

Query String

```
POST https://<SERVER URL>?requesttype=Job.Delete&apikey=YOUR_API_KEY&jobidlist=12
```

Java

```
private void JobDelete(String APIKey, String JobIDList, String Shred, String
ShredOverwrites)  {
        // set parameters
        String sParameters = "requesttype=Job.Delete&apikey=" +
URLEncoder.encode(APIKey, "UTF-8")  + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&shred=" +
java.net.URLEncoder.encode(Shred, "UTF-8") + "&shredoverwrites=" +
java.net.URLEncoder.encode(ShredOverwrites, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
    System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
    StringBuilder output = new StringBuilder();
    java.io.OutputStreamWriter writer = null;
    java.io.BufferedReader reader = null;
    java.io.InputStreamReader isReader = null;
    try {
        // Send data
        java.net.URL url = new java.net.URL(webUrl);
        java.net.URLConnection conn = url.openConnection();
        conn.setDoOutput(true);
        writer = new java.io.OutputStreamWriter(conn.getOutputStream());
        writer.write(parameters);
        writer.flush();
        // Get the response
        isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-8");
        reader = new java.io.BufferedReader(isReader);
        String line;
        while ((line = reader.readLine()) != null) {
            output.append(line);
        }
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    }
    return output.toString();
}
```

7.6.2. Example Response

Heading

```
{
  "result": {
    "action": "Shredded",
    "jobs": [
      {
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
        "job": {
          "jobid": "191215",
          "files": {
            "file": [
              {
                "filename": "2044cded-6dab-4a8c-8f25-762c0824f854.txt",
                "status": "The file has been successfully deleted."
              },
              {
                "filename": "2044cded-6dab-4a8c-8f25-762c0824f854.out.txt",
                "status": "The file has been successfully deleted."
              }
            ]
          },
          "status": "1",
          "description": "The job has been successfully deleted."
        }
      },
      {
        "job": {
          "description": "Job does not exist in system.",
          "status": "0",
          "jobid": "991210"
        }
      }
    ]
  },
  "duration": "5245",
  "startdate": "20171219 17:12:47:651",
  "errortext": "",
  "errorno": "",
  "enddate": "20171219 17:12:52:896",
  "requestid": "20171220-092048.406-1.1.000001"
}
```


Omniscien™ TECHNOLOGIES

Copyright © 2018 Omniscien Technologies. All Rights Reserved

8. JOB.CANCEL

8.1. Job.Cancel Overview

Cancels a job that was queued or currently processing. Optionally, the files associated with each job can also be deleted or shredded.

8.2. Call

HTTP Method	GET
Content Type	text/html

8.3. Parameters

Name	Default Value	Description
requesttype	String	The type of call being made (case insensitive). This should be set to "Job.Cancel".
apikey	String	Your API Key. See **Acquiring API Key from User Profile** for more details.
jobidlist	String	A list of Job IDs delimited by a comma (e.g. "1234,1236,1237") to delete. All files associated with each job will be located and deleted.
deleteshred	Integer	(Optional. Default=0) Determines if the files associated with a job are deleted, shredded or left on the server. Valid values: • 0 = Leave files on server. • 1 = Delete files. • 2 = Shred files using the DOD 5220.22-M methodology for secure deletion.
shredoverwrites	Integer	(Optional Default=5) The number of overwrites when shredding a document.

8.4. Response

String	JSON Response with job status information and confirmation of file deletion/shredding as determined by the parameters.

8.5. Examples

8.5.1. Example Request

CURL

```
curl
-H "Content-Type: application/x-www-form-urlencoded"
-X POST
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
-d "requesttype=Job.Cancel"
-d "apikey=Your_API_KEY"
-d "jobidlist=123"
https://<SERVER URL>
```

Query String

```
GET https://<SERVERURL>?requesttype=Job.Cancel&apikey=YOUR_API_KEY&=123
```

Java

```
private void JobCancel(String APIKey, String JobIDList, String DeleteShred, String
ShredOverwrites)  {
        // set parameters
        String sParameters = "requesttype=Job.Cancel&apikey=" +
java.net.URLEncoder.encode(APIKey, "UTF-8") + "&jobidlist=" +
java.net.URLEncoder.encode(JobIDList, "UTF-8") + "&deleteshred=" +
java.net.URLEncoder.encode(DeleteShred, "UTF-8") + "&shredoverwrites=" +
java.net.URLEncoder.encode(ShredOverwrites, "UTF-8");
        //call RestAPI and receive output
        String returnValue = HttpPost("https://<SERVER URL>", sParameters);
        System.out.println("Output=" + returnValue);
}

private String HttpPost(String webUrl, String parameters) throws Exception {
    StringBuilder output = new StringBuilder();
    java.io.OutputStreamWriter writer = null;
    java.io.BufferedReader reader = null;
    java.io.InputStreamReader isReader = null;
    try {
        // Send data
        java.net.URL url = new java.net.URL(webUrl);
        java.net.URLConnection conn = url.openConnection();
        conn.setDoOutput(true);
        writer = new java.io.OutputStreamWriter(conn.getOutputStream());
        writer.write(parameters);
        writer.flush();
        // Get the response
        isReader = new java.io.InputStreamReader(conn.getInputStream(), "UTF-8");
        reader = new java.io.BufferedReader(isReader);
        String line;
        while ((line = reader.readLine()) != null) {
            output.append(line);
        }
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        if (reader != null) { reader.close(); }
        if (isReader != null) { isReader.close();}
        if (writer != null) { writer.close();}
    }
    return output.toString();
}
```

8.5.2. Example Response

Heading

```
{
```


Omniscien™ TECHNOLOGIES
Copyright © 2018 Omniscien Technologies. All Rights Reserved

```
"result":{
   "jobs":[
      {
         "job":{
            "description":"The job has been successfully cancelled.",
            "status":"1",
            "jobid":"112724"
         }
      }
   ]
},
"requestid":"20170321-092048.406-1.1.000001",
"duration":"300686",
"startdate":"20170321 11:17:29.693",
"errortext":"",
"errorno":"",
"enddate":"20170321 11:22:30.379"
```


Omniscien™ TECHNOLOGIES

Copyright © 2018 Omniscien Technologies. All Rights Reserved