NAPLAN Online:
Results and Reporting API Documentation

Specifications for the extraction of results and reporting data from the National Assessment Platform via API

Release:	1.5
Date of this version:	9-May-22
Author:	NSIP team

Version Control

Results and Reporting Data Set Specification Document Version Control			
Version	Date:	Author/Organization:	Comments
V0.4	15/8/2018	NSIP	Initial Draft
V0.5	23/8/2018	NSIP	Simplified document following internal review, narrowed scope to just results and reporting dataset
V0.7	29/8/2018	NSIP	Added correct paths and HITS details
V0.8	29/8/2018	NSIP	Incorporated feedback from NN, AY, JH
V1.0	10/9/2018	NSIP	Document made final
V1.1	17/9/2018	NSIP	Added section on NAPLAN API purpose/scope (see section 1.2)
V1.2	25/9/2018	NSIP	Added note to section 7 – Choreography noting download of data via API can occur at any time
V1.3	25/2/2019	NSIP	Added section on simultaneous API connections (Section 6) and API daily timings (Section 7)
V1.4	11/1/2021	NSIP	Updated Section 2 – Accessing/Authenticating to the API
V1.5	6/5/2022	NSIP	Timing of API requests

1.   Introduction

1.1 Purpose

This document is designed for use by consumers of results and reporting data resulting in relation to the delivery of NAPLAN Online via the National Assessment Platform. These guidelines are based on the approved NAPLAN Results and Reporting Data Set.

1.2 What is (and isn't) the NAPLAN Results and Reporting API

As discussed later in this document, the API is a simple, read-only REST API which has been established to facilitate the more secure machine to machine transfer of results data. The API uses SIF/XML for its payloads, and it uses the SIF HMAC authentication algorithm. The API should not be confused with a full SIF 3.x endpoint. The REST API is not a SIF 3.x API as specified in the SIF 3.x Infrastructure Specification.

1.3 Terminology

Term	Definition
ACARA	Australian Curriculum and Assessment Authority
ESA	Education Services Australia (tasked with developing the assessment platform)
HITS	Hub Integration Testing Service – A SIF testing service provided by NSIP available here: http://hits.nsip.edu.au/dashboard/
ISR	Individual Student Report (rocket-ship report)
NIAS	NSIP Integration As a Service
NSIP	National Schools Interoperability Program
RRD	Results and Reporting Dataset
SIF	Systems Interoperability Framework
SRM	Student Registration Management System

SSA	School Sector Authority (examples include the Department of Education Victoria, the Department of Education NSW)
SSSR	School Student Summary Report
TAA	Test Administration Authority (examples include the VCAA, NESA, QCAA)
XML	eXtensible Markup Language
XSD	XML Schema Definition

1.4 Additional documentation

This document should be read in conjunction with NAPLAN Online – Jurisdictional Results and Reporting Data Set Technical Specifications (available via NSIP's Github page: https://github.com/nsip/naplan-results-reporting). Particularly relevant to those consuming the Results and Reporting Data Sets are the sections on 1) Timing/ choreography and 2) Impact of ACARA uploads of psychometric data.

1.5 Document distribution

This document will be made available via NSIP's Github page: https://github.com/nsip/naplan-results-reporting

2. Authentication / Access to the APIs

Access to the Results and Reporting API is managed through the Assessment Delivery module of the platform, and only authorised parties can issue credentials to use the API.
In order to obtain credentials to use the Results and Reporting API, please consult the ESA operations team: help@assessform.edu.au , Tel: 1800 620 970. Note that arrangements for issuing credentials may change in the future.

Notes on accessing & authenticating to the API

1. The API is a simple, read-only REST API. It does not realise the additional infrastructure services, or query or header parameters specified in the SIF Infrastructure specification. There is for example no provision for flow control or queues. The REST API should not be confused with a SIF 3.x endpoint. In fact the REST API is not a SIF 3.x API as specified in the SIF 3.x Specification.
2. The API will use the SIF_HMACSHA256 authentication method, as documented in the SIF Infrastructure Services Specification, section 4.1.5 (http://specification.sifassociation.org/Implementation/Infrastructure/3.2.1/Documents/InfrastructureServices_3-2-1.pdf page 24/25).
3. No SIF environments are created for the endpoint: the authentication is purely in consumer mode (so the Create Environment part of the SIF specification is to be ignored, along with the *instanceid* and *userToken* parameters).
4. The API does not support URI query parameters: only HTTP header values are recognised.
5. The SIF_HMACSHA256 authentication method involves adding header values to the HTTP GET request.
   a. *timestamp:* date and time according to the ISO 8601 standard. The time must be in ZULU time to avoid time zone issues (e.g. 2018-08-29T04:20:25.019Z).
   b. *authorization:* the authorisation token, which is calculated based on the *applicationKey*, *password* (shared secret) and *timestamp*. The *applicationKey* and *password* are not included as header values.
6. The *applicationKey* uniquely identifies the consumer for authentication purposes. There will be at least 24 of these (e.g. one key for VIC-GOV, VIC-CATH, VIC-IND, NSW-GOV, NSW-CATH… etc). The applicationKey specific to the client will be issued as credentials by the ESA operations team.
7. The algorithm to produce the access token in the authorization HTTP header based on these three values is outlined in the SIF specification cited above, and is as follows. A Javascript script to generate these headers for the Insomnia REST client is available from https://github.com/nsip/naplan-results-reporting/blob/master/naplan_api.zip.
   a. Concatenate the applicationKey and timestamp, and separate them by a ":" .
   b. Calculate the HMAC SHA 256 of this string using the unsent password (this is the primary key sent as base64 decoded binary).
   c. Base64 encode the result.
   d. Concatenate the applicationKey and the new string, separating them again by a ":"
   e. Base64 encode the result.
   f. Prefix the result with the authentication method (SIF_HMACSHA256), separating them by a space.
8. When the API receives a request for NAPLAN Online results, the following process will occur:
   a. Identify the Authentication Method. (SIF_HMACSHA256)
   b. Base 64 decode to obtain the applicationKey and the Base64 value of the HMAC SHA 256.
   c. Look up the applicationKey in the list of pre-stored authorized clients, to retrieve the pre-stored shared secret (password) specific to that client.

d. Retrieve the timestamp that was used from the message header.

e. Use the applicationKey and timestamp and calculate the SIF_HMACSHA256 value using the shared secret for this client. Compare the result with the string provide in the request.

f. The authentication of the issuing application has been confirmed.

Sample request

```
GET /sifapi/some_endpoint_name HTTP/1.1
Host: example.org
Authorization: SIF_HMACSHA256
bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg==
timestamp: 2013-06-22T23:52-07Z
```

Composition of headers for request

Component	Source/construction	Example
GET /sifapi/some_endpoint_name HTTP/1.1	Specified by Janison/ESA	GET /sifapi/testdata HTTP/1.1 GET /sifapi/schoollist HTTP/1.1 GET /sifapi/schooldata HTTP/1.1
Host: example.org	Specified by Janison/ESA	Host: naplanassessment.edu.au
Authorization: SIF_HMACSHA256 bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadX BmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg= =	Created via application Key, password and timestamp. Refer step 7 above	For: applicationKey = RamseyPortal, password = a1b2c398, timestamp = 2013-06-22T23:52-07Z a. RamseyPortal:2013-06-22T23:52-07 b. 6TVgYwbAhmQc2zALda8ZupfrzfqZ+XD7f2b MA0AzWRo= (*HMAC SHA 256 value coded using a1b2c398 as shared secret, then Base64-encoded*) c. RamseyPortal:6TVgYwbAhmQc2zALda8Zup frzfqZ+XD7f2bMA0AzWRo= d. SIF_HMACSHA256 bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg== (*HMAC SHA 256 value coded using a1b2c398 as shared secret, then Base64-encoded, then prefixed with SIF_HMACSHA256*)
Timestamp	Date/time of the request as per ISO 8601 format	2013-06-22T23:52-07Z

3.   Authorisation

NAPLAN data on the National Assessment Platform is held in tenancies, segregated by state/territory and by sector. (This means there are 24 tenancies for NAPLAN data on the National Assessment Platform: Victoria – Government, Victoria – Catholic, Victoria – Independent; NSW – Government, NSW – Catholic,

NSW – Independent; and so on.) Tenanted data on the platform is strictly segregated: parties are only authorised to access data on a per-tenancy basis. That means that TAAs need to be authorised to access their three tenancies explicitly; and that when a sector is authorised to access a tenancy, it is not granted access to the other sectors' data in the same state or territory.

In this document, "The Authority" refers to the party authorised to use the API on behalf of schools (currently the TAA).

4. SIF Data Model

The National Assessment exposes a REST-based API for NAPLAN-related data. The **Results and Reporting** data set (used to access the results of NAPLAN testing) is captured in the SIF AU data model (refer link below).

The API uses the SIF AU data model (3.4.2+), which is expressed in XML. This document describes how to use the API at a high level, including the endpoints and objects involved. It does not document the elements used within the objects.

For details of the objects sent and returned through the API, including the validation rules they are expected to satisfy, refer to the SIF AU specification and the NAPLAN technical specifications:

- https://github.com/nsip/naplan-results-reporting
- http://specification.sifassociation.org/Implementation/AU/3.4.2/

5. Optimising Performance: Compression

The API for Results & Reporting supports providing payload responses either uncompressed or gzip encoded.

Requesting compressed payload responses is strongly recommended due to the significant reduction in time to transfer data achievable.

To utilise this feature the client must send an Accept-Encoding header to advertise that it supports gzip encoding.

If the Accept-Encoding header is not present payload responses will be returned uncompressed.

Syntax

```
Accept-Encoding: gzip
```

Sample request

```
GET /sifapi/some_endpoint_name HTTP/1.1
Host: example.org
Accept-Encoding: gzip
Authorization: SIF_HMACSHA256
bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg==
timestamp: 2013-06-22T23:52-07Z
```

6.   Optimising Performance: Parallel access

The bulk of the data available from the Results & Reporting API is through the /SchoolData endpoint, with a separate URL instance for each school represented in the tenancy. In order to speed up the total time needed to download the data, it is possible for a client to access several URLs simultaneously, through a pool of clients. However, for a number of reasons including security, the National Assessment Platform constrains the amount of traffic it handles. Parallelized access is subject to the following constraints:

- No client can issue more than 10 API requests simultaneously. The constraint applies to per client, and not just per tenancy: that means that a single TAA can only issue 10 API requests simultaneously, even though they access data from three different tenancies (sectors).
    - Because of the performance impact on the platform, current testing indicates that 10 simultaneous accesses results in data being downloaded only three times faster than through a single connection. These performance metrics are subject to change, but users should plan accordingly.
    - The system is specified to deliver the /SchoolData results for a single school within 30 sec. Current testing indicates that the system at peak performance can deliver the /SchoolData results for a single school within 3 sec; however there is no guarantee that higher throughput will be maintained under operational conditions.

7.   Timing – When to use the API to request data

When configuring the timing of API requests, developers should note the following:

- Clients should only access the API outside of the operational days of test administration during the test window.
- The operational days are from Monday through Friday.
- The test window is typically a 3 week window from mid to late May, and it may be longer if special circumstances obtain. For the exact dates in the current year, please consult the PMO schedule.
- Developers should consult with the ESA Operations team, via the Helpdesk, to request suitable windows for running the API during NAPLAN.

8. Results & Reporting API - Detail

The size of data involved in the Results & Reporting data set for a sector can be large, given the amount of detail for an individual student's results: the results for a full cohort in a sector can amount to up to 50 GB. One way of mitigating this is by using gzip to stream payload; even so, the data set is large enough that the API breaks it up into more tractable chunks.

Three endpoints are exposed by the Results & Reporting API:

1. **Test content**: A nationwide endpoint, for **test content**
   This is the same for all consumers of the API, and is exposed as a single endpoint for all schools.

2. **Schools list**: A sector-specific endpoint, for the **schools list**
   In order to enable clients to iterate through schools, a separate endpoint enumerates the **schools which the client has access to**, and their associated identifiers.

3. **Test results and test registrations:** A school-specific endpoint, for the **test results** for students registered for tests. Data about **test results** and **test registrations** are segregated in the API, with one endpoint for each school involved. Clients make a request to a different endpoint for each school they represent.

The HITS test harness (see below) provides sample data as responses for its representation of the Results & Reporting API. The sample data being served by HITS are also available for download from https://github.com/nsip/naplan-results-reporting/blob/master/NAPLANAPITestData.zip.

Endpoint purpose	1. **Test content**
URL	https://administration.assessform.edu.au/testdata
Inputs	None
Outputs	Returns all NAPCodeFrame, NAPTest, NAPTestlet and NAPTestItem objects for the current NAPLAN assessment cycle.
Scope	The same results are returned to all clients: the endpoint has national scope, as the same tests are administered across the nation.
Notes	N/A

Sample request	`GET /naplan/sifapi/testdata HTTP/1.1`
	`Host: administration.assessform.edu.au`
	`Accept-Encoding: gzip`
	`Authorization: SIF_HMACSHA256`
	`bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg==`
	`timestamp: 2013-06-22T23:52-07Z`

Sample response headers	`content-encoding: gzip`
	`content-type: application/xml`
	`status: 200`

Sample response payload	<NAPResultsReporting> <NAPCodeFrame RefId="…"> … </NAPCodeFrame> … <NAPCodeFrame RefId="…"> … </NAPCodeFrame> <NAPTest RefId="…"> … </NAPTest> … <NAPTest RefId="…"> … </NAPTest> <NAPTestlet RefId="…"> … </NAPTestlet> … <NAPTestlet RefId="…"> … </NAPTestlet> <NAPTestItem RefId="…"> … </NAPTestItem> … <NAPTestItem RefId="…"> … </NAPTestItem> </NAPResultsReporting>

Endpoint purpose	2. **Schools list** - Schools whose NAPLAN results the client has access to
URL	https://administration.assessform.edu.au/schoollist
Inputs	None
Outputs	Returns all SchoolInfo objects for the current NAPLAN assessment cycle whose results the client is authorised to access.
Notes	The data returned by the Schools List endpoint will mostly be drawn from the Australian Schools List, maintained by ACARA as the national source of truth about schools. The endpoint data may be more up to date than the Australian Schools List; but its main purpose is to help iterate through test results endpoints.

Scope	Because NAPLAN data is segmented by sector into tenancies, the endpoint is meant to represent a single tenancy's worth of data: so in the first instance, the endpoint will return all schools in a tenancy. If the authorisation model for the API is refined in future, the output of the endpoint will be refined to reflect the current client authorisation.
Sample request	``` GET /naplan/sifapi/schoollist HTTP/1.1 Host: administration.assessform.edu.au Accept-Encoding: gzip Authorization: SIF_HMACSHA256 bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg== timestamp: 2013-06-22T23:52-07Z ```
Sample response headers	``` content-encoding: gzip content-type: application/xml status: 200 ```
Sample response payload	<NAPResultsReporting> <SchoolInfo RefId="…"> … </SchoolInfo> … <SchoolInfo RefId="…"> … </SchoolInfo> </NAPResultsReporting>

Endpoint purpose	3. **Test results** and **test registrations**
URL	https://administration.assessform.edu.au/SchoolData/{School RefId}
Inputs	School RefId: GUID for the school being queried.
Outputs	Returns the SchoolInfo object for the school queried, as well as the following objects associated with the school for the current NAPLAN assessment cycle: • Test registrations: StudentPersonal, NAPEventStudentLink • Student responses: NAPStudentResponseSet • Test score summaries: NAPTestScoreSummary
Notes	If a student sits some NAPLAN tests in one school and some in another, their registrations and results will appear in the endpoint outputs for both schools. The School RefId is obtained by calling the School List endpoint, and using the RefId attribute of the corresponding SchoolInfo object. The SchoolInfo object also contains the name of the school, and other identifiers for the school, including the Australian Schools List identifier ("ACARA ID").
Scope	The data returned is specific to a school.
Sample request	``` GET /naplan/sifapi/SchoolData/1234 HTTP/1.1 Host: administration.assessform.edu.au Accept-Encoding: gzip Authorization: SIF_HMACSHA256 bmV3OjZUVmdZd2JBaG1RYzJ6QUxkYThadXBmcnpmcVorWEQ3ZjJiTUEwQXpXUm89Cg== timestamp: 2013-06-22T23:52-07Z ```
Sample response headers	``` content-encoding: gzip content-type: application/xml status: 200 ```

Sample response payload	```<NAPResultsReporting>```
	 ``<SchoolInfo RefId="…"> … </SchoolInfo>``
	 …
	 ``<SchoolInfo RefId="…"> … </SchoolInfo>``
	 ``<StudentPersonal RefId="…"> … </StudentPersonal>``
	 …
	 ``<StudentPersonal RefId="…"> … </StudentPersonal>``
	 ``<NAPEventStudentLink RefId="…"> … </NAPEventStudentLink>``
	 …
	 ``<NAPEventStudentLink RefId="…"> … </NAPEventStudentLink>``
	 ``<NAPStudentResponseSet RefId="…"> … </NAPStudentResponseSet>``
	 …
	 ``<NAPStudentResponseSet RefId="…"> … </NAPStudentResponseSet>``
	 ``<NAPTestScoreSummary RefId="…"> … </NAPTestScoreSummary>``
	 …
	 ``<NAPTestScoreSummary RefId="…"> … </NAPTestScoreSummary>``
	 ``</NAPResultsReporting>``

The error payload for all endpoints is derived from the SIF error payload:

```
<error id="{GUID}">
  <Code>{HTTP STATUS CODE}<Code>
  <Scope>{Attempted operation}</Scope>
  <Message>{Simple description of error (max length: 1024)}</Message>
  <Description>{Optional more complete description of error}</Description>
</error>
```

Important notes:
- The API for Results & Reporting uses gzip to compress the payloads in query responses.
- The endpoints cater to read-only request/response calls.
- No endpoint is expected to return a response to a request greater than 5MB in size, gzipped.
- The Results & Reporting API deviates from SIF convention in allowing multiple object classes in a single payload, for consistency. This allows the number of endpoints to be reduced from nine to three.
- The responses for all three endpoints are wrapped in a <NAPResultsReporting> wrapper.

9. Choreography

The expected usage scenario for the results and reporting API is as follows:
- Students complete the NAPLAN assessment
- ACARA completes upload of psychometric data and releases results
- The Authority uses the API to access the results and reporting data. The data is accessed in three stages:
  - One endpoint exposes the test codeframes.
  - One endpoint exposes the list of schools which the Authority has access to.

o Using this list of schools, the Authority requests results and reporting data via a separate endpoint for each school. The Authority may choose to execute these requests in parallel, subject to the constraints already mentioned.

Note: In past years, psychometric data (and therefore student results) were updated more than once by central authorities. When consuming data from the platform via the API, consumers should be prepared to be able to consume the data more than once if needed.

Important: The download of results data via the API can occur at any time and is NOT bound to any of the current controls which exist around extracting the RRD (as a zip file) via the web ui (ie all schools are in the results phase). This means that consumers of data via the API should ensure that they are aware of TAA and ACARA / platform business processes and timings and only consume the data when it is appropriate to do so.

10. Testing the APIs

Test endpoints to consume sample results and reporting data are available via the NSIP HITS service. Request a HITS account here (http://hits.nsip.edu.au/dashboard/).
Note that HITS is intended only as a test harness to establish that clients can connect to SIF endpoints, consume data, and process data. It is *not* intended as a test harness for load testing, and is not built to cope with stress testing.
The HITS test data involves a full set of NAPLAN test data, 10 schools, and 500 students per school. All data on HITS is made up, and does not reflect any real tests.
- Sample test content endpoint in HITS: http://hitstest.nsip.edu.au/api/naplanresuls/TestData
- Sample school list endpoint in HITS: http://hitstest.nsip.edu.au/api/naplanresuls/SchoolList
- Sample school results endpoint in HITS: http://hitstest.nsip.edu.au/api/naplanresuls/SchoolData/{RefID}

**The following differences should be noted between the test APIs/results and production APIs/results:**

- The applicationKeys recognised by HITS are: {state}{sector}, where {state} is one of vic, nsw, act, qld, nt, wa, sa, tas, and {sector} is one of gov, cath, ind. These will not be the same as those issued by the platform.
- The password recognised by HITS for each of those applicationKeys will be issued to users by the NSIP team as they register for HITS. This will not be the same as that issued by the platform.
- HITS will return data through gzip streaming, whether the Accept-Encoding header is set to "gzip" or not.

HITS Testing
The following are instructions for accessing the test instance of the API on HITS using the Insomnia HTTP client (https://insomnia.rest).

1. Install Insomnia: https://insomnia.rest/download , using these instructions https://support.insomnia.rest/article/23-installation
2. Download https://github.com/nsip/naplan-results-reporting/blob/master/naplan_api.zip. This is a Node package for populating Insomnia HTTP requests with the SIF HMAC authorisation headers.
3. Uncompress the Node package, and place the naplan_api folder into the Insomnia plugin directory (see https://support.insomnia.rest/article/26-plugins. To locate the plugin folder, go into Insomnia, click on *Application* then *Preferences*. In the Insomnia preferences screen click on the *Plugins* tab. Click *Show Plugins Folder*. Place the naplan.api folder into this folder.

4. Edit the src.js file (contained in the naplan.api folder) to set the applicationKey and the password variable to the applicationKey and password you have been assigned for HITS. Keep the authMethod variable as is.
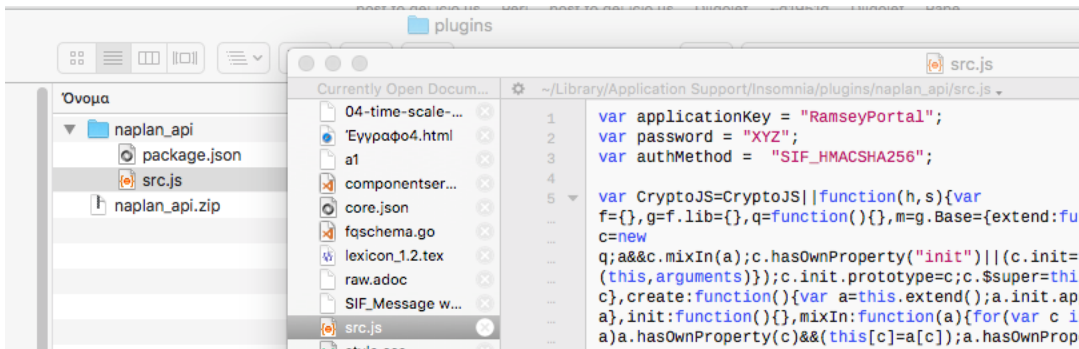
5. Make sure the plugin is recognised: in Insomnia, navigate to *Application > Preferences >* Plugin (pc) or *Insomnia > Preferences >* Plugin (mac).
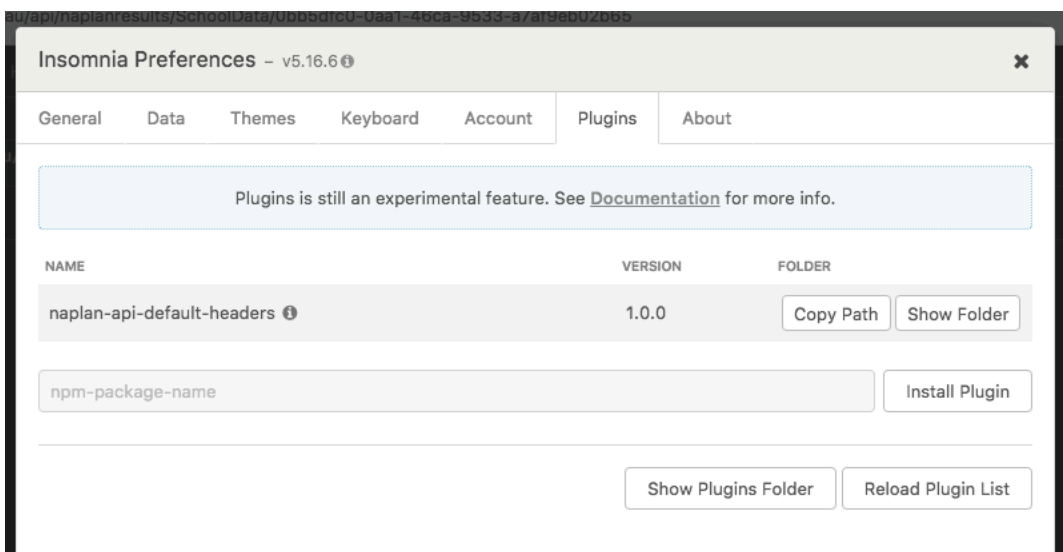
6. This has the effect of computing and adding values for the "Authorization" and "timestamp" headers to all requests coming out of Insomnia, which thereby aligns Insomnia with SIF HMAC256.
7. Issue a request for test data to HITS. In Insomnia, click on *New Request* or *Ctrl+N*. type in a name for the request, GET should be displayed as the method. Click *Create*. Paste the following link in the URL bar http://hits.nsip.edu.au/api/naplanresults/large/TestData, then click *Send*. If successful, the status will show as 200 OK. Click *Save to File* in the *Preview* tab, and choose a location you can find easily later. You may need to change the file name to have an .xml suffix so it can be opened. Open this file to ensure that it contains expected data.
8. Issue a request for the schools list to HITS: HTTP GET http://hits.nsip.edu.au/api/naplanresults/large/SchoolList simply by changing the URL in Insomnia and click *Send*. The resulting school list will display the schools in your jurisdiction. This list can be used to access the RefId for any particular school required for use in the next step.
9. To issue a request for results and reporting data from one school to HITS: HTTP GET , change the URL in Insomnia to http://hits.nsip.edu.au/api/naplanresults/large/SchoolData/**0bb5dfc0-0aa1-46ca-9533-a7af9eb02b65**. The RefId (in bold above) is what would change to access another school's results. Click *Save to File* or *Show Anyway* to view the results.
10. Note that Insomnia will not update the timestamp for a saved request, so if you use the same URL it will expire after 5 minutes. Closing Insomnia and reopening it should correct this issue.