tvsquared

Measure TV How People Watch It

TVSquared ADvantage Enterprise API User Guide

March 2021

V1.1 Enterprise

Contents

Introduction ... 3
- Purpose ... 3
- Audience ... 3
- Version ... 3
- Conventions ... 3

Overview ... 4
- Attribution API ... 4
- Onboarding API ... 4
- Uplift Metric API ... 4

Authentication ... 5
- Making calls to the API ... 5

Attribution API ... 6
- Setup ... 6
- Parameters ... 6
- Inventory ... 6
- Brand inventory ... 7
- Retrieve spot attribution ... 8
- Retrieve spot columns ... 9
- Advanced TV ... 10
- Retrieve attributed spots ... 11
- Spot metric timeseries ... 13
- Spot attribution timeseries ... 14
- Response attribution timeseries ... 14

Using the Attribution API ... 15
- Data warehousing ... 15
- Formatted spot data ... 17
- Direct vs non-direct response view ... 19

Onboarding API ... 21
- Setup ... 21
- Inventory ... 21

Uplift Metric API ... 24
- Setup ... 24
- Inventory ... 24
- Parameters ... 24

Introduction

Purpose

This document lists and describes the API calls used to obtain attribution, onboarding, and uplift metric data (Enterprise only) for the measurement and optimization of TV campaigns.

Audience

This document is intended for TVSquared ADvantage Enterprise partners who use or want to use the Attribution API, Onboarding API and Uplift Metric API.

Version

Version	Date
1.1	March 2021

Conventions

This document uses the following conventions:

Convention	Identifies
	A warning or mandatory task
	Useful aside information
	Further tasks that you can do or helpful tips

Overview

TVSquared ADvantage, the world's leading TV performance analytics and optimization platform, gives advertisers everything they need to measure, optimize and plan TV campaigns to maximize media spend, generate the greatest response and drive sales and conversions.

Attribution API

After your traffic has been modeled and attributed in your TVSquared ADvantage platform, the TVSquared Attribution API can provide flexible access to the attributed TV spot data and metadata.

Attribution data is extracted from TVSquared ADvantage using the Attribution API at a partner level, where a partner is generally an entity responsible for one or more end clients and their associated brands. In circumstances where an end client deals with TVSquared directly, the partner level is synthesized to allow use of the Attribution API.

Typical workflows with the Attribution API involve multiple calls to different API endpoints, extracting and passing parameterized data, to achieve the required results in the end application. The Attribution API lets users automate the acquisition of attribution data in this manner, at a desired granularity, for inclusion in your own data analysis or presentation to end clients.

Onboarding API

Onboarding data is extracted from TVSquared ADvantage using the Onboarding API.

Uplift Metric API

Uplift metric data is extracted from TVSquared ADvantage (Enterprise only) using the Uplift Metric API.

Authentication

The TVSquared API requires you to authenticate with TVSquared ADvantage so that you can access your client and brand data. You will need to set up and use basic authentication to access the available TVSquared Attribution API endpoints. This requires a Basic HTTP Authorization header to be sent with each request.

For more information on HTTP Authorization header, see https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Authorization.

Making calls to the API

The API uses basic authentication over an https connection. To authenticate, you must send an HTTP Authorization header with every API call, containing a hash of the username and password specified previously for use with the Attribution API.

1. To authenticate with the Attribution API using the `curl` command, use the following command line, substituting the username and password of the API user:

   ```
   curl --user <api_username>:<api_password> https://apixx.tvsquared.com/...
   ```

2. Alternatively, you can use a separate 3rd-party command line tool, or your preferred programming language, to generate the authentication header.

Attribution API

The TVSquared Attribution API provides flexible access to automate the acquisition of attributed TV spot data and metadata from TVSquared ADvantage at a desired granularity.

This section lists all the TVSquared ADvantage endpoints and associated information.

Setup

The TVSquared Attribution API is a standard RESTful HTTP API based on GET, POST requests with JSON formatted responses.

The Attribution API exclusively uses the HTTPS protocol for encryption and data security - plain HTTP requests are blocked and will not be successful.

All Attribution API endpoints share the same base URL, accessed at the partner level:

```
https://<partnerdomain>.tvsquared.com/api/v2/
```

where `<partnerdomain>` is a unique identifier for the API in the format `api-xx`. Use your `<partnerdomain>` for the base URL for all Attribution API endpoints.

Parameters

The TVSquared Attribution API follows several conventions for parameters passed and returned in API calls. All date and time parameters are formatted in the ISO standard as follows, unless otherwise stated:

- Date: `YYYY-MM-DD`
- Datetime: `YYYY-MM-DDThh:mm:ss`

Client ID and brand ID are used in several Attribution API endpoints, identified by `<clientid>` and `<brandid>` respectively.

These are unique identifiers for the required client and brand and can be derived from the TVSquared ADvantage portal or from the response of an Inventory request.

Inventory

Performing a **GET** request at the following endpoint returns basic inventory information regarding any partners, clients, or brands that you have access to.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution
```

Response format

```
{
"partners": [list of partners with details],
"clients": [list of clients with associated brand details]
}
```

The partner details include the title, id, API domain and partner domain. Client details include title, id and its associated brand titles and ids. This endpoint can be used to get the appropriate client and brand ids for subsequent API requests.

Additional information

No additional information.

Brand inventory

Performing a **GET** request to this endpoint returns details regarding a brand's spot and action metrics as well as time series splits.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/<brandid>
```

Response format

```
{
"media": <list of names of media types>,
"tz": <the timezone for all dates for this brand in the API>,
"spots": {
"count": <total number of spots>,
"max": <maximum spot date>,
"min": <minimum spot date>
},
"smetrics": <list of names of spot metrics>,
"tsgrans": <list of supported timeseries granularities>,
"umetrics": <list of names of response metrics>,
"tsumsplits": <dictionary containing lists of splits and all the
splitvalues>,
"tsrtmtypes": "tsrtmtypes": <list of names of timeseries RTM types>,
"uactions": <list of names of response actions>,
"tsrefs": <list of names of referrers>
}
```

Parameters

Parameter	Description
MEDIA	A list of the media types, for example, **web**, **app**, **phone**, **sms**.
SMETRICS	A list of the available spot metrics, for example, **audience**, **cost**.
TSGRANS	A list of the timeframes you can use to explore your data for example, **minute**, **week**, **month**.
UMETRICS	A list of the response metrics, for example, **session**, **revenue**.
TSUMSPLITS	A list of time series response metric splits, for example, product, lag, region. These parameters are generally dictionary keys with lists of more specific data as values. For example, a **medium** time response metric split may contain **web** or **phone** as values.
TSRTMTYPES	These parameters allow access to timeseries data for different groupings of the data, as follows: • Baseline: the baseline response • Unattributed: the unattributed baseline response • nondr: the non-direct response, that is the sum of the baseline and unattributed responses • dr: the direct response

Parameter	Description
UACTIONS	A list of available response actions, for example, **allresponse**, **signup**, **subscription**.
TSREFS	A list of referrers names, for example, **search**, **paysearch**, **tvwebsite**, **banner**

Additional information

No additional information.

Retrieve spot attribution

Performing a **GET** request will retrieve a day's worth of attributed spots for the day specified through the URL parameters.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/<brandid>/
spotsjson/<year>/<month>/<day>
```

The returned data includes details about the spot time, daypart, the program the spot appeared in, as well as specific spot and response metric data.

Response format

```
{
"spots": [
{
"aggchannel": "<channel the spot showed on>",
"country": "<country the spot showed in>",
"region": "<region the spot showed in>",
"length": "<length of the spot in seconds>",
"saleshouse": "<saleshouse the spot was bought through>",
"datetime": "<date the spot aired>",
"spotid": "<spotid>",
"daypart": "<daypart the spot showed in>",
"genre": "<genre of the program>",
"programme": "<programme the spot showed in>",
"clocknumber": "<clocknumber associated with this spot>",
"channel": "<channel the spot showed on>",
"um": "<response metrics and values>",
"sm": "<spot metric and values>"
}, ...
]
}
```

Additional information

No additional information.

Retrieve spot columns

This **GET** request will retrieve a list of the column names for spots in a supplied date range.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/
<brandid>/spotheaders/<y1>/<m1>/<d1>/<y2>/<m2>/<d2>
```

The `<y1>`, `<y2>` parameters represent the start and end dates of the date range you want to view spot column names for.

Response format

```
{
"headers": [...]
}
```

The response format is a simple list of spot column names. These column names can then be used to retrieve specific information in subsequent calls. Note that the column types vary across date ranges depending on what data was supplied in the spots and responses during this time.

Additional information

A full list of spot column names.

Column Name	Type	Description
datetime	Ordinal	Date time the spot was transmitted: YYYY-MMDDThh:mm:ss
daypart	Ordinal	Daypart the spot was transmitted during
length	Ordinal	Length of spot in seconds
saleshouse	Ordinal	Saleshouse this spot was bought via
programme	Ordinal	Program the spot was transmitted during
genre	Ordinal	Genre of the program the spot was transmitted during
country	Ordinal	Country the spot was transmitted in
region	Ordinal	Region the spot was transmitted in
clocknumber	Ordinal	Clocknumber
clockgroup	Ordinal	Clocknumber group name
chgenre	Ordinal	Genre of the channel/station the spot was transmitted on
spotid	Ordinal	Our internal ID for the spot

Column Name	Type	Description
feed	Ordinal	Spot feed (US only)
sm.audience.v	Spot metric	Total audience for the spot
sm.cost.v	Spot metric	Total cost of the spot
sm.spots.v	Spot metric	Number of spots composing the spot (for pre-aggregated spots)
um.\<action>\.m.session	Response metric	Total number of response \<action>\ sessions attributed to the spot
um.\<action>\.m.revenue	Response metric	Total response \<action>\ revenue attributed to the spot
um.\<action>\.region.\<regionid>\.m.session	Response metric	Number of response \<action>\ sessions attributed to the spot in \<regionid>\
um.\<action>\.m.region\<regionid>\.m.revenue	Response metric	Total revenue of response \<action>\ attributed to the spot in \<regionid>\

The response `\<action\>` includes actions such as a `signup`, `sale`, or `subscription`.

Advanced TV

A full list of OTT-specific column names. Use these to split the data in any combination.

Note: We have provided the typical use for c_deviceclass, c_placementgroup and c_publisher. The others can vary from partner to partner and can be used for whatever suits. Ensure the column names are exactly the same format as in the table below:

Column Name	Type	Description
c_deviceclass	Ordinal	Type of device the impression was served on
c_placementgroup	Ordinal	Placement group of the impression
c_publisher	Ordinal	Publisher that delivered the ad
c_contentname	Ordinal	Misc
c_contentrating	Ordinal	Misc
c_deliveryplatform	Ordinal	Misc
c_externalname	Ordinal	Misc
c_externalplacementname	Ordinal	Misc

Retrieve attributed spots

Performing a **POST** request returns detailed spot metadata and attribution metrics for a specific date. With the request you should **POST** a list of column names for spot data you are interested in.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/<brandid>/
spotsarray/<year>/<month>/<day>
```

Request and response format

```
{
"headers": [COLUMN NAMES]
}
```

`COLUMN NAMES` should be an ordered list of the spot column values you wish to retrieve. The available column names can be obtained from the `[Retrieve Spot Columns endpoint](#attr-spotColumns)`. Any columns that do not exist in the database will return `Null`. Note that values from the `datetime` column are always returned, and if this column is not included in `COLUMN NAMES`, the datetime values are inserted at the start of each item of spot data.

```
{
"daterange": {
"min": "<min datetime of returned spots>",
"max": "<max datetime of returned spots>"
},
"ordinals": {
"<columnname>": {
"date": <boolean to indicate whether this is a date>,
"values": [
<optional list of values for this ordinal>
],
"title": "<title text for this ordinal>"
},...
},
"headers": {
"<columnname>": <index of per-spot array for this column name>,
...
},
"smetrics": [
<list of spot metrics present in the data>
],
"umetrics": [
<list of attributed response metrics present in this data>
],
"spots": [
[<list of spot column values in the order requested>],
[<list of spot column values in the order requested>],
...
]
}
```

Additional information

No additional information.

Unattributed response data

Performing a **GET** request returns unattributed response timeseries data of a specified granularity between two dates inclusively.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/
<brandid>/ts/raw/<uaction>/<umetric>/<reftype>/<gran>/<medium>/<y1>/
<m1>/<d1>/<y2>/<m2>/<d2>
```

Parameters

Parameter	Description
UACTION	Action to retrieve, for example, **allresponse**, **signup**.
UMETRIC	Metric to retrieve, for example, **session**, **response**.
REFTYPE	Referrer type of timeseries to retrieve, for example, **direct**, **search**.
GRAN	Granularity to retrieve, for example, **day**, **month**.
MEDIUM	Medium to retrieve, for example, **web**, **mobile**.

Possible values for these parameters can be found from performing an `[inventory](#inventory)` request and the subsequent response data.

Response format

```
{
"group": "raw",
"action": "UACTION",
"metric": "UMETRIC",
"type": "REFTYPE",
"gran": "GRAN",
"medium": "MEDIUM",
"date1": "DATE1",
"date2": "DATE2",
"values": [VALUE0, VALUE1, ...]
}
```

Additional information

No additional information.

Spot metric timeseries

Performing a **GET** request retrieves spot metric timeseries data of a specified granularity between two dates inclusively.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/
<brandid>/ts/sm/<smetric>/<gran>/<y1>/<m1>/<d1>/<y2>/<m2>/<d2>
```

Response format

```
{
"group": "sm",
"metric": "SMETRIC",
"gran": "GRAN"
"date1": "DATE1",
"date2": "DATE2",
"values": [VALUE0, VALUE1, ...]
}
```

Additional information

No additional information.

Spot attribution timeseries

Performing a **GET** request returns the attribution timeseries for spots at the granularity requested between two dates inclusively.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/
<brandid>/ts/um/<uaction>/<umetric>/<split>/<splitname>/<gran>/<y1>/
<m1>/<d1>/<y2>/<m2>/<d2>
```

Parameters

Parameter	Description
SPLIT	Data split, for example, **medium**.
SPLITNAME	Data splitname to retrieve, for example, **web**.

Response format

```
[VALUE0, VALUE1, ...]
```

Additional information

No additional information.

Response attribution timeseries

Performing a **GET** request returns the attribution timeseries for response at the granularity requested between two dates inclusively.

URL

```
https://<partnerdomain>.tvsquared.com/api/v2/attribution/<clientid>/
<brandid>/ts/rtm/<uaction>/<umetric>/<tsrtmtype>/<gran>/<medium>/
<y1>/<m1>/<d1>/<y2>/<m2>/<d2>
```

Parameters

Parameter	Description
TSRTMTYPE	Breakdown type of timeseries to receive.

For a full list, see ***Brand Inventory***.

Response format

```
{
"group": "rtm",
"action": "UACTION",
"metric": "UMETRIC",
"type": "TSRTMTYPE",
"gran": "GRAN",
"medium": "MEDIUM",
"date1": "DATE1",
"date2": "DATE2",
"values": [VALUE0, VALUE1, ...]
}
```

Additional information

No additional information.

Using the Attribution API

While Attribution API calls can be used individually to retrieve data from the TVSquared ADvantage, you can also get utility from multiple calls used in sequence.

This section provides end to end examples describing how to use multiple Attribution API calls to retrieve targeted data for typical scenarios.

Attribution API GET and POST calls in the following examples can be made using simple cURL command lines, as described in **Making calls to the API**, or using the libraries of your preferred programming language.

Data warehousing

This example explores a typical early scenario which can be used to familiarize with the Attribution API - data warehousing of spot attribution data on a daily cadence.

To extract spot attribution data from TVSquared ADvantage requires a client and brand ID, and appropriate date parameters, to be passed for data retrieval into the data warehousing application.

1. Retrieve the top-level inventory from the Attribution API using the base Inventory GET request, for example for an api partner domain of `api-123`:

   ```
   https://api-123.tvsquared.com/api/v2/attribution
   ```

   All inventory data available under your assigned permissions is returned, for example:

   ```
   {
   "partners": [
   {
   "api": "api-123.tvsquared.com",
   "portal": "example-partner.tvsquared.com",
   "id": "example-partner",
   "title": "Example Partner"
   }, ...
   ],
   "clients": [
   {
   "brands": [
   {
   "id": "default",
   "title": "default"
   }, ...
   ],
   "id": "example-client",
   "title": "Example Client"
   }, ...
   ]
   }
   ```

2. Inside the `clients` array, find the required client id value which is paired with the `id` key - in this case `example-client`. This client id is static data and can be stored for later use.
3. Retrieve the inventory from the Attribution API filtered down for this client, by passing the client id as a parameter to the Inventory request, for example:

   ```
   https://api-123.tvsquared.com/api/v2/attribution/exampleclient
   ```

4. Within the `brands` array for the specified client data returned, find the required brand id value which is paired with the id key - in this case default. This brand id is static data and can be stored for later use.
5. Retrieve the brand inventory from the Attribution API using the Brand inventory GET request, passing the client id and brand id as parameters, for example:

```
https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
default
```

All brand inventory data for the specified client/brand is returned, for example:

```
{
"media": [
"web",
"app",
"sms",
"phone"
],
"tz": "Europe/London",
"spots":
{
"count": 1000,
"max": "2018-03-04T23:00:00",
"min": "2017-01-02T00:10:00:"
},
...
}
```

6. Use the time range `max` and `min` values inside the `spots` document to determine the next date required for the data warehousing application in YYYY, MM and DD format.
7. Retrieve the spot attribution data for the data warehousing application using the ***Retrieve spot attribution*** GET request, passing the client and brand id, and the required date, for example:

```
https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
default/spotsjson/2018/03/03
```

All attributed spot data for the specified date is returned for use in the application, for example:

```
{
"spots": [
{
"aggchannel": "DRAMA",
"country": "GB",
"region": "national",
"saleshouse": "SALES",
"datetime": "2018-03-03T20:00:57",
"um": {
"all response": {
"medium": {
"web": {
"m": {
"session": 0.0,
"revenue": 0.0,
}
},
"m": {
"session": 0.0,
"revenue": 0.0,
}
},
"m": {
"session": 0.0,
"revenue": 0.0,
}
```

```
},
},
"spotid": 12345,
"length": 30,
"daypart": "breakfast",
"sm": {
"audience": {
"v": 0.0,
},
"cost": {
"v": 0.0,
}
},
"genre": "None",
"programme": "COFFEE PANIC",
"clocknumber": "CREATIVE1",
"channel": "DRAMA"
}, ...
]
}
```

Formatted spot data

This example demonstrates using dynamic data with the Attribution API - producing formatted, attributed spot data based on the available spot data columns, suitable for use in further analysis.

To extract formatted, attributed spot data from TVSquared ADvantage for further analysis requires a client and brand ID, and an appropriate date range in order to identify the available data columns for formatting.

1. Identify an appropriate client and brand id, for example from those stored previously in ***Data warehousing***, and a date range of interest from the ***Brand inventory*** response.
2. Retrieve the spot columns for your date range of interest from the Attribution API by passing the client id, brand id and date parameters to the ***Retrieve spot columns*** GET request, for example:

   ```
   https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
   default/spotheaders/2017/08/02/2017/08/05
   ```

   Spot columns can vary across date ranges depending on what data was supplied in the spots and responses during this time.

   ```
   {
   "headers": [
   "datetime",
   "daypart",
   "chgenre",
   "saleshouse",
   "aggchannel",
   "genre",
   "spotid",
   "um.all.response.m.session",
   "um.sale.m.revenue",
   "sm.cost.v",
   ...]
   }
   ```

3. Retrieve the formatted, attributed spot data for use in further analysis using the ***Retrieve attributed spots*** POST request.

   Send the required headers, as determined by your analysis needs, for retrieval in the POST data:

```
{
"headers": [
"datetime",
"daypart",
"channel",
"um.all.response.m.session",
"um.install.m.session",
"um.install.m.revenue"
]
}
```

   Pass the client id, brand id, and date for the data you are interested in to the URL, for example:

```
https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
default/spotsarray/2017/08/03
```

   All spot metadata and attribution metrics for the date specified are returned for use in further analysis, for example:

```
{
"daterange": {
"max": "2017-08-03T23:31:00",
"min": "2017-08-03T00:18:00",
},
"ordinals": {
"channel": {
"date": false,
"values": [
"SCIENCE",
"HISTORY CHANNEL"
],
"title": "Channel"
},
"daypart": {
"date": true,
"values": [
"breakfast",
"coffee",
"day",
"earlypeak",
"latepeak",
"postpeak",
"bob",
"latenight"
],
"title": "Daypart"
}
},
"headers": {
"datetime": 0,
"daypart": 1,
"channel": 2,
"um.all response.m.session": 3,
"um.install.m.session": 4,
"um.install.m.revenue": 5
},
"spots": [
["2017-08-03T00:18:00", 6, 0, 13.423, 9.576, 0.0],
["2017-08-03T00:33:57", 0, 1, 15.572, 8.944, 0.0]
],
"smetrics": [
"audience",
```

```
"cost",
"spots"
],
"umetrics": [
"session",
"revenue"
],
"uactions": [
"all response",
"sale",
"install"
]
}
```

Direct vs non-direct response view

This example demonstrates extracting timeseries data using the Attribution API, to create a direct versus non-direct response view of the data for comparison.
To extract timeseries data from TVSquared ADvantage to create views for comparison requires a date range and several additional parameters. These determine what metrics (values) and splits (data groupings) are present in the returned data.

1. Identify an appropriate client and brand id, for example from those stored previously in ***Data warehousing***, and the following additional parameters from the Brand inventory response:
    - Response action (for example, `signup` or `all response`)
    - Response metric (for example, `session`)
    - Time granularity (for example, `day`)
    - Medium (for example, `web`)
    - Date range of interest
2. To get the non-direct response timeseries data (sum of baseline response and non-attributed response), pass these parameters to the ***Response attribution timeseries*** GET request, with a `ts_rtm_type` of `nondr`, for example:

    ```
    https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
    default/ts/rtm/all%20respons/session/nondr/day/
    web/2018/6/1/2018/7/1
    ```

    Non-direct response timeseries data for the dates, metrics, and splits supplied is returned. For example, the following results, with a value returned for each day in the date range specified:

    ```
    {
    "group": "rtm",
    "action": "all response",
    "metric": "session",
    "type": "nondr",
    "gran": "day",
    "medium": "web",
    "date1": "2018-06-01T00:00:00",
    "date2": "2018-07-01T00:00:00",
    "values": [
    890.7431672131167,
    634.0000000000001,
    ...
    568.0000000000001,
    401.00000000000006,
    405.39431611011844
    ```

```
]
}
```

3. To get the direct response timeseries data for comparison, pass the same parameters to the ***Response attribution timeseries*** GET request with a `ts_rtm_type` of `dr`, for example:

```
https://api-123.tvsquared.com/api/v2/attribution/exampleclient/
default/ts/rtm/all%20respons/session/dr/day/
web/2018/6/1/2018/7/1
```

Direct response timeseries data for the dates, metrics and splits supplied is returned for comparison with the non-direct response, for example:

```
{
"group": "rtm",
"action": "all response",
"metric": "session",
"type": "dr",
"gran": "day",
"medium": "web",
"date1": "2018-06-01T00:00:00",
"date2": "2018-07-01T00:00:00",
"values": [
35.14378426195513,
0.0,
...
315.60518845883036
]
}
```

Onboarding API

This section of the API allows retrieval of client onboarding. The base URL is appended with `onboarding` so that the base URL for the attribution endpoints becomes:

```
https://<partnerdomain>/api/v2/onboarding
```

All URLs in this section start with this base URL.

Setup

The TVSquared Onboarding API is a standard RESTful HTTP API based on GET, POST requests with JSON formatted responses.

The API exclusively uses the HTTPS protocol for encryption and data security - plain HTTP requests are blocked and will not be successful.

All API endpoints share the same base URL, accessed at the partner level:

```
https://<partnerdomain>.tvsquared.com/api/v2/
```

where `<partnerdomain>` is a unique identifier for the API in the format `api-xx`. Use your `<partnerdomain>` for the base URL for all API endpoints.

Inventory

Performing a GET request at the following endpoint returns basic inventory information regarding any partners, clients, or brands that you have access to.

URL

Users who have partner-level access permissions can call this endpoint:

```
https://<partnerdomain>/api/v2/onboarding
```

Response format

```
{
  partners: [
    {

      clients: [
        { client details
          Onboarding: { details }
        },...

      ]

    },...

  ]
}
```

The partner details include the title, id, api domain and partner domain. Client details include title, id and its associated brand's titles and ids. This endpoint can be used to get the appropriate client and brand ids for subsequent API requests.

Example response format

```
{
  "partners": [
    {
      "title": "Example Partner",
      "id": "example-partner",
```

```
"api": "api-123.tvsquared.com",
"portal": "example-partner.tvsquared.com",
"clients": [
  {
    "title": "Example Client 1",
    "id": "example-client-1",
    "brands": [
      {
        "title": "default",
        "id": "default"
      }
    ],
    "onboarding": {
      "metadata": {
        "webmaster": "example@tvsquared.com",
        "timecreated": null,
        "duasignedstamp": null,
        "tagconfirmedstatus": null,
        "tagconfirmedstamp": null,
        "volumesconfirmedstatus": true,
        "volumesconfirmedstamp": "2020-04-16T00:00:00",
        "qastamp": null,
        "market": null,
        "campaignstartdate": "2020-04-20T00:00:00",
        "campaignenddate": "2020-05-19T00:00:00",
        "websiteurl": "www.tvsquared.com",
        "advertisercontact": "example@tvsquared.com",
        "spotsconfirmedstamp": "2020-05-06T00:00:00",
        "spotsconfirmedstatus": true,
        "advertiserid": EXAMPLE-001,
        "actions": [
          "action1",
          "action2"
        ],
        "tag1": "value",
        "tag2": "value",
        "tag3": "value"
      },
      "status": {
        "id": 3,
        "title": "Data Validation"
      }
    }
  },
  {
    "title": "Example Client 2",
    "id": "example-client-2",
    "brands": [
      {
        "title": "default",
        "id": "default"
      }
    ],
    "onboarding": {
      "metadata": {
        "webmaster": "example@tvsquared.com",
        "timecreated": "2020-01-10T20:39:48",
        "duasignedstamp": "2020-01-11T15:46:19",
        "tagconfirmedstatus": true,
        "tagconfirmedstamp": "2020-02-19T00:00:00",
        "volumesconfirmedstatus": true,
        "volumesconfirmedstamp": "2020-04-16T00:00:00",
        "qastamp": "2020-01-28T13:31:10",
        "market": {
          "name": "market",
          "title": "Market"
        },
        "campaignstartdate": "2020-05-08T00:00:00",
        "campaignenddate": null,
```

```
              "websiteurl": "www.tvsquared.com",
              "advertisercontact": "example@tvsquared.com",
              "spotsconfirmedstamp": "2020-05-06T00:00:00",
              "spotsconfirmedstatus": true,
              "advertiserid": "EXAMPLE-002",
              "actions": [
                "action1",
                "action2",
                "action3"
              ],
              "tag1": "value",
              "tag2": "value",
              "tag3": "value"
            },
            "status": {
              "id": 5,
              "title": "Live"
            }
          }
        }
      ]
    }
  ]
}
```

Uplift Metric API

This section of the API allows retrieval of uplift data. The base URL is appended with `uplift` so that the base URL for the uplift endpoint becomes:

```
https://<partnerdomain>/api/v2/uplift/
```

All URLs in this section start with this base URL.

Setup

The TVSquared Uplift Metric API is a standard RESTful HTTP API based on GET, POST and DELETE requests with JSON formatted responses.

The API exclusively uses the HTTPS protocol for encryption and data security - plain HTTP requests are blocked and will not be successful.

All API endpoints share the same base URL, accessed at the partner level:

```
https://<partnerdomain>.tvsquared.com/api/v2/
```

where `<partnerdomain>` is a unique identifier for the API in the format `api-xx`. Use your `<partnerdomain>` for the base URL for all API endpoints.

Inventory

Performing a GET request at the following endpoint, returns uplift metric information for the specified client, action, medium, date from and date to.

URL

```
https://<partnerdomain>/api/v2/uplift/<clientslug>/<brand>/<uaction>/<media>/
        <y1>/<m1>/<d1>/<y2>/<m2>/<d2>
```

The <y1>, <y2> parameters represent the start and end dates of the date range you want to view uplift metrics for.

Parameters

All date and time parameters are formatted in the ISO standard as follows, unless otherwise stated:

- Date: `YYYY-MM-DD`
- Datetime: `YYYY-MM-DDThh:mm:ss`

Client id and brand id are used in several API endpoints, identified by `<clientid>` and `<brandid>` respectively.

These are unique identifiers for the required client and brand and can be derived from the TVSquared ADvantage portal.

Parameters

Parameter	Description
MEDIA	A list of the media types, for example, **web**, **app**, **phone**, **sms**.
UACTION	A list of available response actions, for example, **all response**, **signup**, **subscription**.

Example response format

```
{
  "selectedperiod" :{

  "upliftpercentage":<value>
},

  "previousperiod" :{

  "upliftpercentage":<value>
},

  "difference" :{

  "percentage": <value>
},
}
```

Additional information

The API will return these uplift values:

Key Name	Description
selectedperiod	The date range requested in the API call.
previousperiod	The date range prior to the selectedperiod, calculated from the dateto minus the number of days in the date range from selectedperiod.
difference	The percentage difference in uplift between the selectedperiod and the previousperiod.
upliftpercentage	The value as a percentage.