

Simplicity Loyalty Mobile API Developer Documentation

1. Service endpoints

The Simplicity Loyalty platform has separate development, test and production environments. Each environment is isolated, with the web hosting for each environment under a separate domain. The web site structure for each environment is identical. All app development is to be completed using the test environment. As such any examples or data in this document will be based on the test environment. The mobile app must accommodate the URL's for the services changing for the published app.

This API will be versioned by the URL with part of the URL branch being the version (original publication year and month). Future updates to add new non-breaking features or bug fixes will replace the current version. If a breaking change is required a new version branch will be introduced, which would also require a new version of the mobile app to be published. The previous version(s) of the API will continue to run unmodified wherever possible. In the event that there are underlying changes to the platform that make this unachievable then the previous version(s) of the service will be reconfigured to return an error code to indicate that they are no longer available. The mobile app should accommodate this which would in effect require the user to update the app to be able to continue using it. See the [Error codes](#) section later in this document for details.

All service endpoints are relative to the root URL. The current root URL's for the test and production environments are as follows:

Environment	Root URL
Test	https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/
Production	https://www.demoloyalty.simplicitycrm.com/services/mobileapi/201605/

All documentation for each service request will be treated as being relative to the root URL and will only specify the relative portion of the URL. All request examples will be from the test environment, and may not be from the current version of the service.

2. Supported request and response formats

The following MIME types are supported in both the Accept and Content-Type request headers:

Data type	MIME type
JSON	application/json
XML	application/xml

All samples in this document will utilise JSON for both request and response formats.

Wherever no format is specified in the response, or a format that isn't supported, request and response handling will default to JSON. It is best practice to always include the accept and content-type headers where appropriate.

All samples in this document have formatted JSON, including null values for readability and completeness. It should be expected that the published test and production services will be configured to return unformatted JSON excluding null values.

3. Error codes

3.1 ErrorCode format

All ErrorCode values returned by the API conform to the underlying Loyalty service error codes. The ErrorCode field is an integer where the value is comprised of two separate values to indicate the type of error, and the entity / area which cause the error. The format is:

XXXXXXXXYY

XXXXXX = type of error

YY = entity

The following table contains the XXXXXX codes:

XXXXXX code	Description
0	No error occurred
010000	An exception occurred in the underlying service layer
020100	A required entity record was not found in the database
020200	Inputs haven't met criteria or are missing
030000	Failed batch processing
030100	An exception occurred in a database stored procedure
040100	Invalid token or failed login
040300	Token is not linked to required access group
900001	An exception occurred in the API code
900002	API has been deprecated and is no longer available.
900003	API is offline temporarily for upgrade / maintenance

The following table contains the YY codes:

YY code	Description
00	N/A
01 - 26	Simplicity Loyalty System database entities
27 - 97	Not currently used
98	Mobile API configuration
99	Mobile API source code

As the error codes are represented as integers, any leading zeroes are removed.

Example:

The error code would be 3010002 if a stored procedure generated an exception for Simplicity database entity 02.

3.2 HTTP status codes

Wherever possible, a standardised error structure is returned from the API regardless of the HTTP status code. It is normal to get an error response structure with a non-zero error code regardless of the HTTP status code. Some errors may occur outside the API's control and will not conform to the standardised error structure. Below is an example of the standardised error structure:

```
{  
  "ErrorCode": 0,  
  "ErrorDescription": null  
}
```

3.3 Application exceptions

If an exception occurs in the application, a response with a 500 status code will be returned with the error code being set to 90000199. Below is a sample response:

```
HTTP/1.1 500 Internal Server Error  
Cache-Control: no-cache  
Pragma: no-cache  
Content-Type: application/json; charset=utf-8  
Expires: -1  
Server: Microsoft-IIS/10.0  
Date: Fri, 13 Nov 2015 03:41:35 GMT  
Content-Length: 81
```

```
{  
  "ErrorCode": 90000199,  
  "ErrorDescription": "Example exception message"  
}
```

Note that it is possible to receive a 500 status code with a different structure in the response data if the exception occurs outside the API's control, or in the exception handler itself. Under these circumstances, an exception could return a response similar to below:

```
HTTP/1.1 500 Internal Server Error  
Cache-Control: no-cache  
Pragma: no-cache  
Content-Type: application/json; charset=utf-8  
Expires: -1  
Server: Microsoft-IIS/10.0  
Date: Fri, 13 Nov 2015 03:43:51 GMT  
Content-Length: 1844
```

```
{  
  "Message": "An error has occurred.",  
  "ExceptionMessage": "Example exception message",  
  "ExceptionType": "System.Exception"  
}
```

3.4 Application offline

During upgrades and maintenance where it is a requirement that the API is taken offline, the API will return a service unavailable response (503) with a standardised structure with error codes populated. Since the API application is not running during this period, the response is generated from a static file, and as such the content type is reported as text/html rather than application/json. The ErrorCode value will always be 90000398. An example of the response is below:

```
HTTP/1.1 503 Service Unavailable
Content-Type: text/html
Retry-After: 3600
Server: Microsoft-IIS/8.5
Date: Fri, 13 Nov 2015 03:14:21 GMT
Content-Length: 79
```

```
{
  "ErrorCode": 90000398,
  "ErrorDescription": "Application unavailable"
}
```

3.5 API version deprecation

The API supports versioning through the URL branch. At some point in the future it may become necessary to deprecate one or more of the versions due to it being impossible or impractical to continue to maintain the older versions. In these cases it's possible to configure the application to return a service unavailable (503) response with standard data structure and error codes populated. The ErrorCode value will always be 90000298. Additional, configurable data will be returned that could be used by the application to display a prompt to the user or redirect them to a website or the app store page to force them to update the app. The message and URL are configurable for each mobile platform. An example of this response from an Android device is below:

```
HTTP/1.1 503 Service Unavailable
Cache-Control: no-cache
Pragma: no-cache
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.5
Date: Fri, 13 Nov 2015 04:11:23 GMT
Content-Length: 223
```

```
{
  "Data": {
    "Message": "Android specific user message",
    "URL": "https://play.google.com/store/apps/details?id=com.simplicitycrm.loyalty"
  },
  "ErrorCode": 90000298,
  "ErrorDescription": "API Deprecated"
}
```

The platform specific message and URL are determined by inspecting the User-Agent HTTP header. If the platform can't be determined from the User-Agent, a default message and URL can be configured.

Deprecation configuration currently supports Android, iOS and Windows devices, however Windows has been disabled since no Windows app is being developed. Additional platforms would require a code change to be supported.

4. Authenticating using OAuth

4.1 Authorization header

The Authorization HTTP header is used for access to all protected resources, using the OAuth authorization scheme.

All Authorization header parameters are case sensitive, including parameter names.

The following Authorization header parameters must be specified in order to authenticate successfully:

- `oauth_consumer_key`
- `oauth_timestamp`
- `oauth_signature_method`

The following Authorization header parameter is used for transmitting encrypted data. For example, when authenticating a user's credentials:

- `data`

The following Authorization header parameter is used for accessing protected resources:

- `oauth_token`

Optionally, the following Authorization header parameter can be specified in order to restrict the use of the token to the device:

- `device`

All Authorization header parameters must be specified in key / value pairs as specified in RFC 5849

<http://tools.ietf.org/html/rfc5849#section-3.5.1>

An example Authorization header is below (line breaks between parameters have been included for clarity):

```
Authorization: OAuth oauth_signature_method="HMAC-SHA256",  
  oauth_timestamp="1528842504",  
  oauth_consumer_key="pjdrOnv4F06D3IP0mnP4AA",  
  device="TEST_DEVICE_ID",  
  data="kymaM4GCEzwZTW9bub1Hn93L9p0L4bytgr3GLn1Yx6yNMIwn1etwf6pFSb810Dec2ZjibngETS  
Q/ip1g32gLF3dkYBMs6lMx9qGw2JKDJYnMY6u0dcrohE22AcrKQxrmU8Tg+BNgKw0q8LxroU2eQ==",  
  oauth_signature="3Glu8QHEdritUQsReAdCOWSVGqn/KaluyCwwW8Xe8Y0="
```

4.1.1 `oauth_consumer_key`

This parameter is a cryptographic key specific to the mobile app. This key will be provided by Simplicity, along with an application secret. The key must be included in the Authorization header as provided by Simplicity. All requests

must specify this value or will receive a Bad Request (400) response. The following example key will be used for all examples in this document:

pjdrOnv4F06D3IP0mnP4AA

4.1.2 oauth_timestamp

This parameter is the time the request was initiated from the application expressed in Unix time. That is, the number of seconds that have elapsed since midnight 1 Jan 1970 UTC. Traditionally implementations of Unix time have used a signed 32 bit integer to express the date, which has a maximum value of 2038-01-19T03:14:07+00:00. More modern implementations use a 64 bit signed integer which has a range of approx. 239 billion years. This API implements a signed 64 bit integer. The following table illustrates some examples for clarity:

Date/Time	Unix Timestamp
1970-01-01T00:00:00+00:00	0
1970-01-01T13:00:00+13:00	0
2015-12-02T09:29:22+00:00	1449048562
2015-12-02T22:29:22+13:00	1449048562
2015-11-28T23:54:23+00:00	1448754863
2015-11-28T23:54:23+13:00	1448708063
2038-01-19T03:14:07+00:00	2147483647
2038-01-19T16:14:07+13:00	2147483647
2106-02-07T06:28:14+00:00	4294967294
2106-02-07T19:28:14+13:00	4294967294
2242-03-16T12:56:28+00:00	8589934588
2242-03-17T01:56:28+13:00	8589934588

4.1.3 oauth_signature_method

This parameter specifies the hash function used to generate the signature. This must be set to HMAC-SHA256. No other values are currently supported.

4.1.4 oauth_signature

This parameter contains the Base64 encoded signature. See the [Request signature](#) section for details on how to create the signature.

4.1.5 oauth_token

This parameter contains the token received from the Service resource: [Token](#).

4.1.6 data

This parameter is used to transmit encrypted data. The unencrypted data uses a JSON structure and UTF8 encoding. When authenticating a user's credentials the unencrypted data would look similar to the below example:

```
{
  "UserName": "test.user@simplicitycrm.com",
  "Password": "P@ssw0rd",
  "Timestamp": 1528781748
}
```

The Timestamp included in the above JSON structure must match the timestamp specified in the `oauth_timestamp` parameter.

As an example, the above JSON structure with the formatting removed can be represented as below:

```
{"UserName":"test.user@simplicitycrm.com","Password":"P@ssW0rd","Timestamp":1528781748}
```

The above JSON string is encrypted using the Rijndael (AES-256) algorithm. The required initialisation vector (IV) and key to perform this encryption will be supplied by Simplicity. The following example IV and key (base64 encoded) will be used for all examples in this document:

```
IV: kP0t3H1FRh0MbKgS4ap3KA==  
Key: F/Tdp34tohqrqg7E3IPCdu+4rx64x2yyIzTv160Gr0zM=
```

Make sure your encryption algorithm has:

- a) Key size = 256 bytes
- b) Block size = 128 bytes
- c) Padding = PKCS7
- d) Mode = CBC

UTF8 encoded string:

```
{"UserName":"test.user@simplicitycrm.com","Password":"P@ssW0rd","Timestamp":1528781748}
```

Encrypt the above value to produce the following hexadecimal value:

```
0x3BFFFCB0731516EC1C8ECC79F6A9C1515F4D16FBA865E8C4F8284B084B860BD1F2BCF261E8B226862114C  
C70D6F518078BE674CD810C48975B61F634CB702D71759E0A60772CE80F33E684EE5125ABDE236BF3331BA  
AE03C65FC28DE9E572FA77
```

Base64 encode the above value to produce the following result:

```
O//LBzFRbsHI7MefapwVFfTRb7qGXoxPgoSwhLhgvR8rzyYeiyJoYhFMxw1vUYB4vmdM2BDEiXW2H2NMtwLXF1  
ngpgdyzoDzPmh05RJaveI2vzMxuq4Dx1/Cjen1cvp3
```

4.1.7 device

This is an optional parameter for the token request process. When supplied, this should be a value that uniquely identifies the device, or an application on a particular device. If this parameter is included in the initial token request, it must be specified for all requests using that token. Likewise, if the device parameter isn't included in the initial token request, it must be excluded for all requests using that token.

4.2 Request signature

To generate the signature, a normalised signature base string needs to be constructed.

The following steps must be used to build the signature:

1. Take the HTTP verb of the request.
2. Normalise the request URL by converting to lowercase, removing any query string parameters and trailing slashes.
3. Take all OAuth Authorization parameters required for the request as described in the [Authorization header](#) section excluding oauth_signature.
4. Format all parameters as {parameter name}={parameter value}
5. Sort OAuth Authorization parameters alphabetically.
6. URL encode all values from steps 1, 2 and 5. Uppercase hexadecimal digits must be used for all percent-encodings as specified in RFC 3986 URI normalization <http://tools.ietf.org/html/rfc3986#section-2.1>
7. Join all values from step 6 using the & character as a separator
8. Hash the string from step 7 using the HMAC-SHA256 algorithm. Use the application secret as the hash key
9. Base64 encode the value from step 8. This result is the signature.

For the purposes of this document the following application secret is used:

Ad8L9XKmvx51DLtJKbiWQmmhocgFaUEP3lfWsrklTmg

4.2.1 Example: Token

Parameter	Value
HTTP verb	POST
URL	https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/token
data	kymaM4GCEzwZTW9bub1Hn93L9p0L4bytgr3GLnlYx6yNMiWn1etwf6pFSb8l0Dec2ZjibngETSQ/ip1g32gLF3dkYBMs6lMx9qGw2JKDJYn9eb00TmUFLTJurE1thDa7/Lbq37/Tb5tBqq0mf/WhDg==
device	TEST_DEVICE_ID
oauth_consumer_key	pjdrOnv4F06D3IP0mnP4AA
oauth_signature_method	HMAC-SHA256
oauth_timestamp	1528781915

The values from the above table create the following normalised signature base string (step 7):

POST&https%3A%2F%2Ftest.demoloyalty.simplicitycrm.com%2Fservices%2Fmobileapi%2F201605%2Foauth%2Ftoken&data%3DkymaM4GCEzwZTW9bub1Hn93L9p0L4bytgr3GLnlYx6yNMiWn1etwf6pFSb8l0Dec2ZjibngETSQ%2Fip1g32gLF3dkYBMs6lMx9qGw2JKDJYn9eb00TmUFLTJurE1thDa7%2FLbq37%2FTb5tBqq0mf%2FWhDg%3D%3D&device%3DTEST_DEVICE_ID&oauth_consumer_key%3DpjdrOnv4F06D3IP0mnP4AA&oauth_signature_method%3DHMAC-SHA256&oauth_timestamp%3D1528781915

Hashing the above normalised signature base string using the specified application secret, would result in the following Base64 encoded signature:

VD7nXPYEzGao0EGw4Ba6JkRwfGDCxuA9jtS2DZm0src=

4.2.2 Example: Member profile / dashboard

Parameter	Value
HTTP verb	GET

URL	https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/7222c1c109704357b5d6ff76856a66a7
device	TEST_DEVICE_ID
oauth_consumer_key	pjdrOnv4F06D3IPOmnP4AA
oauth_signature_method	HMAC-SHA256
oauth_timestamp	1528782023
oauth_token	ZiRGZVacz2oG8htSDypuDkC/RpgJn2V2a1Kaurh7j6pxl1w/OMbN0j3r1EDYKFdZTWFIzbLTZffqbnocQgvFa9f++15h3n93cjfAEmB0Gki0k9FV4Qo7Rq95aXQbKlqrL4K+qEMyv84m3haD17wd9yY34Br4e7QWIhIAihmv9urqwRtktnoTtRlu5hpl1BRTJoESNerkZntiQCvzpy8EkZ3e2RtCdgpJBY4hJWJ5dXa+1Py16dxLMTS0rd88mncy2y3vPGpvOgvFiCZ82wLSxVTrA+ITFJi41V3mDLRmmQsrBawgp+74Y3maSccFotSsp7mcF6XgkBUQGy5z8vZpibiQ2CAfJ4UAcMiV4fTjTCiNayCKIM2I06eLxPGYmitcbymvZG6p1dP1QkouNIOrjHuWfD33y8I5LHenXrW3W87isyT1GPPID3zfzXcOMJvec/5YQ9pwry3yKkldbsnjYx+GQzpTdvwtMnr1gLFGBfIU+hR3+PVPw4V/abFKpjmjgNgrNcSQOnpKZRxRariCG3pzWV6JQ+TPLugFVj673mIGE1D7Bzi1LlmHhhL7pYuTmyGXTnCM1c/tCvUTBcHOoYR/okw/lsKIky5FPhlaHE=

The values from the above table create the following normalised signature base string:

```
GET&https%3A%2F%2Ftest.demoloyalty.simplicitycrm.com%2Fservices%2Fmobileapi%2F201605%2Fmembers%2F7222c1c109704357b5d6ff76856a66a7&device%3DTEST_DEVICE_ID&oauth_consumer_key%3DpjdrOnv4F06D3IPOmnP4AA&oauth_signature_method%3DHMAC-SHA256&oauth_timestamp%3D1528782023&oauth_token%3DZiRGZVacz2oG8htSDypuDkC%2FRpgJn2V2a1Kaurh7j6pxl1w%2FOMbN0j3r1EDYKFdZTWFIzbLTZffqbnocQgvFa9f%2B%2B15h3n93cjfAEmB0Gki0k9FV4Qo7Rq95aXQbKlqrL4K%2BqEMyv84m3haD17wd9yY34Br4e7QWIhIAihmv9urqwRtktnoTtRlu5hpl1BRTJoESNerkZntiQCvzpy8EkZ3e2RtCdgpJBY4hJWJ5dXa%2B1Py16dxLMTS0rd88mncy2y3vPGpvOgvFiCZ82wLSxVTrA%2B1ITFJi41V3mDLRmmQsrBawgp%2B74Y3maSccFotSsp7mcF6XgkBUQGy5z8vZpibiQ2CAfJ4UAcMiV4fTjTCiNayCKIM2I06eLxPGYmitcbymvZG6p1dP1QkouNIOrjHuWfD33y8I5LHenXrW3W87isyT1GPPID3zfzXcOMJvec%2F5YQ9pwry3yKkldbsnjYx%2BGQzpTdvwtMnr1gLFGBfIU%2BhR3%2BPVPw4V%2FabFKpjmjgNgrNcSQOnpKZRxRariCG3pzWV6JQ%2BTPUGFVj673mIGE1D7Bzi1LlmHhhL7pYuTmyGXTnCM1c%2FtCvUTBcHOoYR%2Fokw%2FlsKIky5FPhlaHE%3D
```

Hashing the above normalised signature base string using the specified application secret, would result in the following Base64 encoded signature:

```
xGwaulAx92FcZdszh3lr3xTdZBXMyu6x1XpbsqbET58=
```

4.2.3 Example: Member change password

The values used in the “data” parameter are:

```
{
  "OldPassword": "P@ssW0rd_0ld",
  "NewPassword": "P@ssW0rd_new",
  "Timestamp": 1528783296
}
```

For instructions on properly encrypting and encoding this data see section [data](#).

Parameter	Value
-----------	-------

HTTP verb	POST
URL	https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/082b14a457f844aebb046fc013bfad27/change-password
data	w9BBBy8n9HZnuIR+Jfk4+AMExWEPmoREh4kO+sU865blj8wwRlg1NQE1jk+rkg8x0e0uWZ5r05xkD497N2aJr0wKeEA1DlCupo5QK/OLtavZEeu/OwmDT26jiXgbbuQ3
device	TEST_DEVICE_ID
oauth_consumer_key	pjdrOnv4F06D3IPOmnP4AA
oauth_signature_method	HMAC-SHA256
oauth_timestamp	1528783296
oauth_token	ZiRGZVacz2oG8htSDypuDpLq1uUzFcirQXyZYqABUT3h4U7SMua9CVD0qBkxw5u4XiTJC57a9RHv2Pp/YVkdri/17i8giW8IRwHUQ41A7ukwnPyWRllOIUQgaFrXLv54ZRCEiWt96w2uADyS6CFnbn/NpO8Gn1ERUR2luOM1APyQ5RAPBdkCL0ur+aAYOMWu69LmyCETFEN0yBorjK1sVdtj3mveWfkGpbgl8Lbi0zhI3YHanKfu1+N2nUZaj/6J6/YdTjf9XcrBQhF6Bkr34/7EHuJWg3r4MJZmxrVSskqrDLJpo39wv58wEtA/4t28ofV/UyZALpjU+uje4f2Yu0FuKweqhtn1e8xNbjdL3hjkvCpBSsdy+9h2j4khdhJceILLJwrIW35Jfq4sqNPB9SvoCO6sfhcgCzqWL1n6f1XpbXtSVuBYpbKC0Jl61KoVBId7VBNiNzQ+qqSj0FV6eByBhujq1UnlOllaEmUumGs8cnwN2XZLUauArr/n9rKVQxkNL1xqTBf61VnEAFUTp6sgnmh9B9HzkckLif5ZvDrvXojOY/Vnyf+GZ4ToabcNPN950iQYk5ajYyqm3V4lAwT2U86Zq0/wYjdsuuK5hg=

The values from the above table create the following normalised signature base string:

```
POST&https%3A%2F%2Ftest.demoloyalty.simplicitycrm.com%2Fservices%2Fmobileapi%2F201605%2Fmembers%2F082b14a457f844aebb046fc013bfad27%2Fchange-password&data%3Dw9BBBy8n9HZnuIR%2BJfk4%2BAMExWEPmoREh4kO%2BsU865blj8wwRlg1NQE1jk%2Brkg8x0e0uWZ5r05xkD497N2aJr0wKeEA1DlCupo5QK%2FOLtavZEeu%2FOwmDT26jiXgbbuQ3&device%3DTEST_DEVICE_ID&oauth_consumer_key%3DpjdrOnv4F06D3IPOmnP4AA&oauth_signature_method%3DHMAC-SHA256&oauth_timestamp%3D1528783296&oauth_token%3DZiRGZVacz2oG8htSDypuDpLq1uUzFcirQXyZYqABUT3h4U7SMua9CVD0qBkxw5u4XiTJC57a9RHv2Pp%2FYVkdri%2F17i8giW8IRwHUQ41A7ukwnPyWRllOIUQgaFrXLv54ZRCEiWt96w2uADyS6CFnbn%2FNpO8Gn1ERUR2luOM1APyQ5RAPBdkCL0ur%2BaAYOMWu69LmyCETFEN0yBorjK1sVdtj3mveWfkGpbgl8Lbi0zhI3YHanKfu1%2BN2nUZaj%2F6J6%2FYdTjf9XcrBQhF6Bkr34%2F7EHuJWg3r4MJZmxrVSskqrDLJpo39wv58wEtA%2F4t28ofV%2FUyZALpjU%2BUje4f2Yu0FuKweqhtn1e8xNbjdL3hjkvCpBSsdy%2B9h2j4khdhJceILLJwrIW35Jfq4sqNPB9SvoCO6sfhcgCzqWL1n6f1XpbXtSVuBYpbKC0Jl61KoVBId7VBNiNzQ%2BqqSj0FV6eByBhujq1UnlOllaEmUumGs8cnwN2XZLUauArr%2Fn9rKVQxkNL1xqTBf61VnEAFUTp6sgnmh9B9HzkckLif5ZvDrvXojOY%2FVnyf%2BGZ4ToabcNPN950iQYk5ajYyqm3V4lAwT2U86Zq0%2FWYjdsuuK5hg%3D
```

Hashing the above normalised signature base string using the specified application secret, would result in the following Base64 encoded signature:

```
zDIrqjksB0Q2r8r0voGiLk20Qc/fjYn+VF7xaH5JNyA=
```

4.3 Obtaining an OAuth token

To obtain an OAuth token a POST request is made to either `oauth/token` or `account/login` resources. Both resources require the user's credentials to be included in the Authorization header, as described below.

The following Authorization header parameters are supported (**red** parameters are mandatory):

- **data**

- device
- **oauth_consumer_key**
- **oauth_signature**
- **oauth_signature_method**
- **oauth_timestamp**

Using the information from the [Authorization header](#) section populate the values for each parameter and create the signature.

Sample Request:

```
POST https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/token
HTTP/1.1
Authorization: OAuth oauth_signature_method="HMAC-SHA256",oauth_timestamp="1528783414",oauth_consumer_key="pjdrOnv4F06D3IP0mnP4AA",device="TEST_DEVICE_ID",data="kymaM4GCEzwZTW9bub1Hn93L9p0L4bytgr3GLn1Yx6yNmiWn1etwf6pFSb810Dec2ZjibngETSQ/ip1g32gLF3dkYBMs6lMx9qGw2JKDJY1fxSs98/l0aBzg/haj/L1T/oFaDAakbxDZqT8D9aZaYQ=",oauth_signature="OPAE1pA+qrk7UWx5AFKSI2CTY40KZAgNnRWaAS/eLEc="
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
Connection: Keep-Alive
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 805
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Tue, 12 Jun 2018 06:03:36 GMT
```

```
{
  "Data": {
    "Token":
    "ZiRGZVacz2oG8htSDypuDuPDCK3c2mthUyabVxTl60rmq232Kq3FCVq2J0B+Lg2Qm4x5PYZXJZ7U1UUC8ByGRAoN32fd3g/KZhnWzLYxW0U1S5cNnU2MCb5WwdXnqyl2xGnM11h0p8b/hvYCG6uUQJDEj8AGzf/L7tokaJnJlgPfiubejRjv33DYdsS0wxcfSi7NXQAp20/gvvSI3Ivq4sXW0rR112g8PMUlwYV3JGTz/EHpPUwxMed2asg/BBA6bB5yrJghME7YEiyQocPR97oHeCjRwldFmY4FrSnsILkYHG1ZPxyaCwR8Ih82gcBFZLHmWnbGbI8V2b024F18GQhARvi4EZ50CRgqj0j+Ni8ucWGrQ8NjTS+Y8fjI65cVrvxUM00Fq5ZNGzit1eKjFOulFg4DB7cDn3L1bWxsxa56HEx0zWbXdi91PjfwDY/ySi/XzXOC2KYh3EPuMH7Yzj8DJZ9UnLslwSLgFzME/lS7ve9nPJEJ1m1HCGKKrEqHhwQ70TLbxexq2C2M+N/kpoe7K3PjhZXPnMq6k859kjV3JScwLwsRgJbA6SjgmEpYOS69W9MdyRkKAYd16fsHNLv3vAuppUrDp14VbUf9bnnng=",
    "TokenExpiry": "2018-06-13T05:43:36",
    "ResourceID": "c51579c158ca427bb6da6afb85bc5090"
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

}

4.3.1 Sample C# source code

The following C# console application code is supplied to assist developers integrating with this API. This is not intended to be production ready code and is to only be used as a guideline.

```

using System;
using Newtonsoft.Json;
using System.Security.Cryptography;
using System.IO;
using System.Collections.Generic;
using System.Net;
using System.Text;
using System.Net.Http;
using System.Net.Http.Headers;

namespace MobileAPIOAuthExample
{
    class Program
    {
        static void Main(string[] args)
        {
            //Note: These values are from the documentation and must be replaced with supplied urls, keys
            and credentials in order to work with the API
            string url =
"https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/token";

            string consumerKey = "pjdrOnv4F06D3IP0mnP4AA";
            string consumerSecret = "Ad8L9XKmvx51DLtJKbiWQmmhocgFaUEP31fWsrk1Tmg";

            string userName = "angus.simplicitycrm.com+3@gmail.com";
            string password = "Example#Pass12";

            byte[] encryptionIV = Convert.FromBase64String("kP0t3H1FRh0MbKgS4ap3KA==");
            byte[] encryptionKey =
Convert.FromBase64String("F/Tdp34tohqrq7E3IPCdu+4rx64x2yyIzTv160Gr0zM=");

            long timestamp = (long)((DateTime.UtcNow - new DateTime(1970, 1, 1, 0, 0, 0)).TotalSeconds);

            //STEP 1: Create and encrypt the data packet to authenticate the user

            //create data object
            dynamic dataObject = new { UserName = userName, Password = password, Timestamp = timestamp };
            string dataJson = JsonConvert.SerializeObject(dataObject);

            Console.WriteLine("Data structure: {0}", dataJson);
            Console.WriteLine();

            //encrypt using AES256
            string dataEncrypted = string.Empty;
            using (var encryptionProvider = Rijndael.Create())
            {
                encryptionProvider.IV = encryptionIV;
                encryptionProvider.Key = encryptionKey;

                encryptionProvider.Padding = PaddingMode.PKCS7;
                encryptionProvider.Mode = CipherMode.CBC;
            }
        }
    }
}

```

```
        using (var cryptoTransform = encryptionProvider.CreateEncryptor(encryptionKey,
encryptionIV))
        using (var encryptedStream = new MemoryStream())
        using (var cryptStream = new CryptoStream(encryptedStream, cryptoTransform,
CryptoStreamMode.Write))
        {
            using (var writer = new StreamWriter(cryptStream))
            {
                writer.Write(dataJson);
            }

            dataEncrypted = Convert.ToBase64String(encryptedStream.ToArray());
        }
    }

    Console.WriteLine("Data encrypted: {0}", dataEncrypted);
    Console.WriteLine();

    //STEP 2: Create OAuth Signature:

    //create oauth parameters
    var oauthParameters = new SortedDictionary<string, string>();
    var values = new List<string>();

    oauthParameters.Add("oauth_signature_method", "HMAC-SHA256");
    oauthParameters.Add("oauth_consumer_key", consumerKey);
    oauthParameters.Add("oauth_timestamp", timestamp.ToString());
    oauthParameters.Add("data", dataEncrypted);

    //first item is the request verb, uppercase
    values.Add("POST");

    //second item is the url, lower case, with any querystring parameters removed
    string urlLower = url.ToLower();
    if (urlLower.IndexOf("?") > 0)
    {
        urlLower = urlLower.Substring(0, urlLower.IndexOf("?"));
    }
    if (urlLower.EndsWith("/"))
    {
        urlLower = urlLower.TrimEnd('/');
    }

    // WebUtility.UrlEncode must be used to normalise the hex digits for escaped chars as
uppercase. HttpUtility.UrlEncode uses lower case hex digits.
    values.Add(WebUtility.UrlEncode(urlLower));

    //OAuth parameters are then added in alphabetical order
    foreach (string key in oauthParameters.Keys)
    {
        if (!string.IsNullOrEmpty(oauthParameters[key]))
        {
            // WebUtility.UrlEncode must be used to normalise the hex digits for escaped chars as
uppercase. HttpUtility.UrlEncode uses lower case hex digits.
            values.Add(WebUtility.UrlEncode(string.Format("{0}={1}", key, oauthParameters[key])));
        }
    }

    string signatureValue = string.Join("&", values);
    Console.WriteLine("Signature structure: {0}", signatureValue);
    Console.WriteLine();
```

```

string signatureHash = string.Empty;
using (var hmac = new HMACSHA256(Encoding.UTF8.GetBytes(consumerSecret)))
{
    signatureHash =
Convert.ToBase64String(hmac.ComputeHash(Encoding.UTF8.GetBytes(signatureValue)));
}
Console.WriteLine("Signature hashed: {0}", signatureHash);
Console.WriteLine();

//STEP 3: Make OAuth Request

var oauthHeader = new StringBuilder();
foreach (string key in oauthParameters.Keys)
{
    if (!string.IsNullOrEmpty(oauthParameters[key]))
    {
        oauthHeader.AppendFormat("{0}=\"{1}\"", " ", key, oauthParameters[key]);
    }
}
oauthHeader.AppendFormat("oauth_signature=\"{0}\"", signatureHash);
Console.WriteLine("OAuth Header: {0}", oauthHeader);
Console.WriteLine();

using (var client = new HttpClient())
{
    client.BaseAddress = new Uri(url);
    client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderValue("OAuth",
oauthHeader.ToString());
    var response = client.PostAsync("", new StringContent("", Encoding.UTF8,
"application/json")).Result;
    Console.WriteLine("OAuth Response :{0} {1}", response.StatusCode,
response.Content.ReadAsStringAsync().Result);
    Console.WriteLine();
}

Console.WriteLine("Press any key to exit");
Console.ReadLine();
}
}
}

```

4.4 Accessing a protected resource

In order to access a protected resource, a request is made to the resource URL with the Authorization header populated as described below.

The following Authorization header parameters are supported (red parameters are mandatory):

- data
- device
- **oauth_consumer_key**
- **oauth_signature**
- **oauth_signature_method**
- **oauth_timestamp**
- **oauth_token**

Note that the header parameter “data” should only be provided if the protected resource requires it. Check the relevant resource description in [Service Resources](#) to see if the parameter “data” is required. Do not provide an empty parameter.

Using the information from the [Authorization header](#) section populate the values for each parameter and create the signature.

The signature for accessing a protected resource MUST include the parameter “**oauth_token**”. The parameter “data” should only be included in the signature if the protected resource requires it. For details see [Request signature](#).

The example below illustrates using the protected refresh-token resource.

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/refresh-token HTTP/1.1
Authorization: OAuth oauth_signature_method="HMAC-SHA256",oauth_timestamp="1528783491",oauth_consumer_key="pjdrOnv4F06D3IP0mnP4AA",oauth_token="ZiRGZVac2oG8htSDypuDurTnIGA6IJydVcSww4Zq0GT8VsVnH7w0KrRmnKCdD2gPqLiibUxJkZKq/AwfWHj4xnkEmFiLLuvfe15f7rhbdg3UapuXVsSayVwkaDU5GPk0xrzMMhyomVrBCZEdRR3w4aWdYWZ0s4zW+0r0tIsfxpg89kZEQfuh7dz9nToPNTjCIuL6iEVrQ6j1WfXSIEz9Zrszhjiz0x7TGNLCc40uhvW9maJ9GgXlsgn3e1R/OrdzKTUXiLXW9+iRNM9J/if5WNYjusDLywu/RZ2SSec1lmMmvdfk5WDYbLVxWearN+AuNW4gXuj4XJ0rpRLNlkqMZhnNdiWbpj9Q4zyFOnYEgyc2YNZ5TKFd3bj03qIJ0NS17hZdCsgige/mC4sokWtmaRbbAcEej95ULfh7175uoJ/i0FTJ4MUjXYOgym1d3zdVm45CriiyFJRvgc+u6iPTvU23884yXpigoTWMi0e0xZqVB0MSpd5C8BX9u55DnnZ2m8lzS/M80Top2BpQSwb720ibv9PmeYJUzaD/tX0Bxu20g1lR+Fq4PgVT0PBCHm6WAxb6hZxLXe03rhG5TEy18eJIhpN6oY1v+c2s8iEaLT4=",device="TEST_DEVICE_ID",oauth_signature="GUbMn7+e7PNtzjtEU3xunggrWiWPNzfhnKH40fUsfbE="
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 805
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Tue, 12 Jun 2018 06:04:53 GMT
```

```
{
  "Data": {
    "Token":
    "ZiRGZVac2oG8htSDypuDurTnIGA6IJydVcSww4Zq0GT8VsVnH7w0KrRmnKCdD2gPqLiibUxJkZKq/AwfWHj4xnkEmFiLLuvfe15f7rhbdg3UapuXVsSayVwkaDU5GPk0xrzMMhyomVrBCZEdRR3w4aWdYWZ0s4zW+0r0tIsfxqiHPJZ6T4WY/B9uT6eLKy58ax7pPcShhhB4WcLVcLn6GkvY5r1anwZKYn3U+6hQ6cu2UVqk/uFrjkduny+ImwpH
```



```
AofbLITa0l7so/ZBEbSDF960SNeIyMFUmsBdtgi0XPggopEzGImJh5LV4z1JBZ/Afk0OpXtEVOFh/kOUhBIxbl  
QtHettBAEQpVTgalFRZ4i9NlpOGVX1BuRcyImZRvhFAb50XxfWQGh8W8JZQAQl0GoPBVkfLzqufWbnv0B0knE3  
PAjWsYWAIs8faEwmPp806+BE0ix9fDzyvKHWPJb0JDg8U0l7PPqw8Dpho3Y6u9hbbE82pu0KNqJoqRXS4B+df  
iVBT8aC3m4245T/gicAlM0YyTBmyiYm4039of0z6kosKecCfGq1k0ob9u3hQfFFRdevbd1V0JUE9AKg4lSTrXu  
GaIDMuWkqQSkk6vt5U=",  
  "TokenExpiry": "2018-06-13T05:44:53",  
  "ResourceID": "00770131886a4b5596c3a075ba4b3bc1"  
},  
"ErrorCode": 0,  
"ErrorDescription": null  
}
```

5. Service Resources

The following structure is employed for all data resource responses:

```
{  
  "Data": {  
    ...  
  },  
  "ErrorCode": 0,  
  "ErrorDescription": null  
}
```

Some responses which have no data associated with them will return a simple HTTP status code response.

Example:

```
HTTP/1.1 403 Forbidden  
Cache-Control: no-cache  
Pragma: no-cache  
Expires: -1  
Server: Microsoft-IIS/10.0  
Date: Wed, 11 Nov 2016 09:56:13 GMT  
Content-Length: 0
```

5.1 Token

Authentication is achieved using a modified OAuth model.

In order to receive a token, an empty POST request is made to the following resource with the OAuth authentication scheme Authorization header:

[oauth/token](#)

See [Authenticating using OAuth](#) section for more information on creating an authenticated request.

The response includes the following information:

- Token
- TokenExpiry
- ResourceID

The Token and ResourceID must be supplied for all future requests for that user. Note that the ResourceID is unique for that token and therefore is only valid when supplied in conjunction with the supplied Token.

Sample Request:

```
POST https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/token
HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
Connection: Keep-Alive
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 793
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 04:45:59 GMT
```

```
{
  "Data": {
    "Token": {TOKEN},
    "TokenExpiry": "2017-06-30T04:25:59",
    "ResourceID": "586e1a884a124c59bd5b2245cf38de58"
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.2 Token refresh

When a token is issued, it has a short expiry time. Once this expiry time has passed the token can no longer be used, i.e. the user will effectively be logged out. In order to keep a user logged in, the token must be refreshed prior to the expiry. A new token is issued for the same resource and will have a new expiry time.

A new token can be retrieved by making an authenticated POST request to the following resource:

[oauth/refresh-token](#)

The response is identical to the [Token](#) response. See [Authenticating using OAuth](#) section for more information on creating an authenticated request.

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/oauth/refresh-token HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 805
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 04:47:30 GMT
```

```
{
  "Data": {
    "Token": {TOKEN},
    "TokenExpiry": "2017-06-30T04:27:30",
    "ResourceID": "28780e29fb9e4b918f8f6d410ba373fa"
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.3 Member forgotten password

Password reset is achieved using a modified OAuth model.

In order to reset a password, an empty POST request is made to the following resource with the OAuth authentication scheme Authorization header:

[account/forgotten-password](#)

The following values must be provided in the header parameter "data":

- Username
- Timestamp

Example:

```
{
  "UserName": "test.user@simplicitycrm.com",
  "Timestamp": 1449050412
}
```

The user will receive an email with instructions on how to reset their password.

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/account/forgotten
-password HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
Connection: Keep-Alive
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 51
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 04:48:48 GMT
```

```
{
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.4 Member profile / dashboard

Member profile / dashboard information can be retrieved by making an authenticated GET request to the following resource:

members/{ResourceID}

where the ResourceID is the value retrieved from the authentication request.

If the member is already at the maximum tier (e.g. Platinum), the following values will be returned for the next tier specific data (subset of JSON response shown below for clarity):

```
{
  "Data": {
```

```
"ProgramMembership": {  
  "NextTier": "Platinum",  
  "PointsRequiredToReachNextTier": 0.00000000,  
  "PercentageProgressTowardsNextTier": 100.0  
}  
}  
}
```

Sample Request:

```
GET  
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/aa22a1c61  
8e844e58aff9b227c3d4181 HTTP/1.1  
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}  
Accept: application/json  
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 200 OK  
Cache-Control: no-cache  
Pragma: no-cache  
Content-Length: 1371  
Content-Type: application/json; charset=utf-8  
Expires: -1  
Server:  
Strict-Transport-Security: max-age=31536000  
Date: Thu, 29 Jun 2017 04:50:17 GMT
```

```
{  
  "Data": {  
    "PersonalDetails": {  
      "Title": "Dr",  
      "GivenName": "Testing",  
      "FamilyName": "Member",  
      "EmailAddress": "test.user@simplicitycrm.com",  
      "DateOfBirth": "1982-06-01T00:00:00",  
      "Address": {  
        "Line1": "14 Triton Drive",  
        "Line2": null,  
        "Suburb": "Rosedale",  
        "City": "Auckland",  
        "PostCode": "0632",  
        "Country": "New Zealand"  
      },  
      "Phone": {  
        "IDD": null,  
        "STD": "09",  
        "Local": "1234567"  
      }  
    },  
    "OrganisationDetails": null,  
  }  
}
```

```
"ProgramMembership": {
  "JoinDate": "2017-06-29T12:37:20",
  "MembershipNumber": "21585616",
  "CurrentTier": "Bronze",
  "NextTier": "Diamond Tier for Dealers",
  "PointsBalance": 0.0,
  "PointsRequiredToReachNextTier": 2000.00000000,
  "PercentageProgressTowardsNextTier": -100.0
},
"ErrorCode": 0,
"ErrorDescription": null
}
```

5.5 Member profile / dashboard combined with token

Members can be logged in with the profile / dashboard result included in a single request. This can be achieved by sending an empty POST request with the OAuth authentication scheme Authorization header to the following resource:

[account/login](#)

The Authorization header parameters required as described for [Token](#). See [Authenticating using OAuth](#) section for more information on creating a valid authenticated request.

The response includes the same token information as the [oauth/token](#) resource. This data will be found in the Data.Authorization node.

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/account/login
HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
Connection: Keep-Alive
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 2144
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:08:55 GMT
```

```
{
  "Data": {
    "Authorization": {
      "Token": {TOKEN},
      "TokenExpiry": "2017-06-30T21:48:55",
      "ResourceID": "3c8f751df4ff4e6387a91364f0be846a"
    },
    "PersonalDetails": {
      "Title": "Dr",
      "GivenName": "Testing",
      "FamilyName": "Member",
      "EmailAddress": "test.user@simplicitycrm.com",
      "DateOfBirth": "1982-06-01T00:00:00",
      "Address": {
        "Line1": "14 Triton Drive",
        "Line2": null,
        "Suburb": "Rosedale",
        "City": "Auckland",
        "PostCode": "0632",
        "Country": "New Zealand"
      },
      "Phone": {
        "IDD": null,
        "STD": "09",
        "Local": "1234567"
      }
    },
    "OrganisationDetails": null,
    "ProgramMembership": {
      "JoinDate": "2017-06-29T12:37:20",
      "MembershipNumber": "21585616",
      "CurrentTier": "Bronze",
      "NextTier": "Diamond Tier for Dealers",
      "PointsBalance": 0.0,
      "PointsRequiredToReachNextTier": 2000.00000000,
      "PercentageProgressTowardsNextTier": -100.0
    }
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.6 Member join

Member join is achieved using a modified OAuth model.

In order to for a member to join, a POST request is made to the following resource with the OAuth authentication scheme Authorization header:

[account/join](#)

The following values must be provided in the header parameter "data":

- UserName
- Password
- Timestamp

Example:

```
{  
  "UserName": "test.user.joining@simplicitycrm.com",  
  "Password": "P@ssw0rd_N3wM3mb3r",  
  "Timestamp": 1449050412  
}
```

The following values must be provided in the POST body:

- ProgramRSN
- CardType
- CardStatus
- PersonalDetails
 - Title
 - GivenName
 - FamilyName
 - EmailAddress
 - DateOfBirth
 - Address
 - Line1
 - Line2
 - Suburb
 - City
 - Postcode
 - Country
 - Phone
 - IDD
 - STD
 - Local
- OrganisationDetails
 - CompanyName
 - Contact_Title
 - Contact_GivenName
 - Contact_FamilyName
 - Contact_EmailAddress
 - Contact_DateOfBirth
 - Contact_Address
 - Line1
 - Line2
 - Suburb
 - City
 - Postcode
 - Country
 - Contact_Phone
 - IDD

- STD
- Local

Provide either PersonalDetails or OrganisationDetails, do NOT provide both.

Sample Request:

POST https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/account/join
HTTP/1.1

Authorization: OAuth {OAUTH PARAMETERS}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 398
Expect: 100-continue
Connection: Keep-Alive

```
{
  "PersonalDetails": {
    "Title": "Mr",
    "GivenName": "Steven",
    "FamilyName": "Randall",
    "EmailAddress": "test.user.2@simplicitycrm.com",
    "DateOfBirth": "1968-01-01T00:00:00+13:00",
    "Address": {
      "Line1": "14 Triton Drive",
      "Line2": "",
      "Suburb": "Rosedale",
      "City": "Auckland",
      "PostCode": "0632",
      "Country": "New Zealand"
    },
    "Phone": {
      "IDD": "+64",
      "STD": "9",
      "Local": "9265400"
    }
  },
  "ProgramRSN": "a0cfba9b-237e-4d0a-b8d4-112fb193ecf2"
}
```

Sample Response:

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 51
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:14:25 GMT


```
{  
  "ErrorCode": 0,  
  "ErrorDescription": null  
}
```

5.7 Member profile update

Member profile data can be updated by making an authenticated POST request to the following resource:

members/{ResourceID}

where the ResourceID is the value retrieved from the authentication request.

Only member profile data in the PersonalDetails section may be updated, all other profile data will be ignored.

Sample Request:

POST

```
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/0d150df0c  
dcc49abab0fb4922aaec4dd HTTP/1.1  
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}  
Accept: application/json  
Content-Type: application/json  
Host: test.demoloyalty.simplicitycrm.com  
Content-Length: 322  
Expect: 100-continue
```

```
{  
  "PersonalDetails": {  
    "Title": "Dr",  
    "GivenName": "Testing",  
    "FamilyName": "Member",  
    "EmailAddress": "test.user@simplicitycrm.com",  
    "DateOfBirth": "1982-06-01T00:00:00+12:00",  
    "Address": {  
      "Line1": "14 Triton Drive",  
      "Suburb": "Rosedale",  
      "City": "Auckland",  
      "PostCode": "0632",  
      "Country": "New Zealand"  
    },  
    "Phone": {  
      "STD": "09",  
      "Local": "1234567"  
    }  
  }  
}
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 51
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:17:03 GMT
```

```
{
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.8 Member change password

Member password can be updated by making an authenticated POST request to the following resource:

[members/{ResourceID}/change-password](#)

where the ResourceID is the value retrieved from the authentication request.

The following values must be provided in the header parameter “data”:

- OldPassword
- NewPassword
- Timestamp

Example:

```
{
  "OldPassword": "P@ssW0rd_old",
  "NewPassword": "P@ssW0rd_new",
  "Timestamp": 1449050412
}
```

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/a2b01eccb
b084cd3b68421bc0ca6cbb0/change-password HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Content-Type: application/json
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 51
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:35:46 GMT
```

```
{
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.9 Member transactions

Transactions for a specified member can be retrieved by making an authenticated GET request to the following resource:

members/{ResourceID}/transactions

This is a wrapper around the standard SelectTransactionDetailList loyalty service. This service supports paging and date range searching using query string parameters. If not specified, default values are used. Query string parameters must be URL encoded.

Example:

members/{ResourceID}/transactions?page=3&pageSize=10&fromDate=2016-02-24T00%3A00.000%2B12%3A00&toDate=2016-06-01T00%3A00.000%2B12%3A00

The following table contains the available query string parameters and their default values:

Parameter name	Default value
page	1
pageSize	100
fromDate	1753-01-01T00:00:00.000
toDate	9999-12-31T23:59:59.997

Note: dates are ISO-8601 compliant. See [Date and time formats](#) for a list of accepted formats.

For additional documentation on the standard service see <http://www.simplicitycrm.com/developers/web-service-api/loyalty-web-service-api/loyalty-service/loyalty.aspx#method:LoyaltyClient.SelectTransactionDetailList>

Sample Request:

```
GET
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/5f97ea59925e44b0b03d73a322f41a8d/transactions?page=1&pageSize=1&fromDate=1914-01-01&toDate=9999-01-01 HTTP/1.1
```

Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Host: test.demoloyalty.simplicitycrm.com

Sample Response:

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 699
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:43:59 GMT

```
{
  "Data": {
    "PageNumber": 1,
    "PageStart": 1,
    "PageEnd": 1,
    "PageRowCount": 1,
    "TotalPages": 1,
    "TotalRows": 1,
    "Items": [
      {
        "RSN": "da236364-0b94-4fa1-b0be-8c5a5ee5fb5d",
        "DisplayValue": null,
        "SaleValue": 0.0,
        "PartnerTransactionDate": "2017-06-30T10:43:05",
        "Organisation_DisplayValue": "Richard.co",
        "Card_Number": "21585616",
        "Transaction_Type": "ADJUSTMENT",
        "Points": 10.00000000,
        "Transaction_ItemDescription": "Adjustment - Goodwill",
        "TransactionProcessedDate": "2017-06-30T10:43:06"
      }
    ]
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.10 Member vouchers

Vouchers for a specified member can be retrieved by making an authenticated GET request to the following resource:

[*members/{ResourceID}/vouchers*](#)

This is a wrapper around the standard SelectVouchers loyalty service. This service supports paging using the page and pageSize query string parameters. If not specified, the default values are page=1, pageSize=100.

Example:

members/{ResourceID}/vouchers?page=3&pageSize=10

For additional documentation on the standard service see <http://www.simplicitycrm.com/developers/web-service-api/loyalty-web-service-api/loyalty-service/loyalty.aspx#method:LoyaltyClient.SelectVouchers>

Sample Request:

```
GET
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/f5d602608
0d4474c89779bdd329241e0/vouchers?page=1&pageSize=10 HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 816
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 22:59:16 GMT
```

```
{
  "Data": {
    "PageNumber": 1,
    "PageStart": 1,
    "PageEnd": 2,
    "PageRowCount": 2,
    "TotalPages": 1,
    "TotalRows": 2,
    "Items": [
      {
        "ProgramMembership_RSN": "94005062-4a68-4e1d-95da-ac5f47cc1bc2",
        "Status": "VALID",
        "VoucherExpiry": "2018-02-01T00:00:00",
        "VoucherNumber": "DF7JYV3J3",
        "VoucherValidFrom": "2017-06-30T10:57:49",
        "VoucherValue": 5.00000000
      },
      {
        "ProgramMembership_RSN": "94005062-4a68-4e1d-95da-ac5f47cc1bc2",
        "Status": "VALID",
        "VoucherExpiry": "2018-01-01T00:00:00",
        "VoucherNumber": "DF7JYV2PF",
        "VoucherValidFrom": "2017-06-30T10:57:31",

```

```
        "VoucherValue": 15.00000000
      }
    ]
  },
  "ErrorCode": 0,
  "ErrorDescription": null
}
```

5.11 Member offers

Vouchers for a specified member can be retrieved by making an authenticated GET request to the following resource:

members/{ResourceID}/offers

This is a wrapper around the standard SelectCampaignOfferForMemberList loyalty service. This service supports paging using the page and pageSize query string parameters. If not specified, the default values are page=1, pageSize=100.

Example:

members/{ResourceID}/offers?page=3&pageSize=10

For additional documentation on the standard service see <http://www.simplicitycrm.com/developers/web-service-api/loyalty-web-service-api/loyalty-service/loyalty-campaign-offer.aspx#method:LoyaltyCampaignOfferClient.SelectCampaignOfferForMemberList>

Sample Request:

```
GET
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/7915a6f2bd3845c394bb10b891bb3cc0/offers?page=1&pageSize=1 HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 1746
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 23:02:10 GMT
```

```
{
  "Data": {
    "PageNumber": 1,
```

```
"PageStart": 1,
"PageEnd": 1,
"PageRowCount": 1,
"TotalPages": 1,
"TotalRows": 1,
"Items": [
  {
    "RSN": "685cee0d-6e65-4266-a33f-a74de8fa8f92",
    "DisplayValue": "Test $100 offer",
    "OfferDetailHeading": "Test $100 offer",
    "OfferDetailText": "Test $100 offer\nTest $100 offer\nTest $100 offer\nTest
$100 offer",
    "OfferName": "Test $100 offer",
    "Active": true,
    "ActiveFrom": "2017-05-29T00:00:00",
    "ActiveTo": "9999-12-31T23:59:59",
    "Visible": true,
    "VisibleFrom": "2017-05-29T00:00:00",
    "DisplayOrder": 50,
    "ProductCode": "TESTING",
    "GalleryImages": [
      {
        "RSN": "5b106d07-9765-4400-a10c-3cb312133245",
        "DisplayValue": "download.jpg",
        "DisplayOrder": 1,
        "FeaturedImage": true,
        "ImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/gallery/724f7d8c96d4465a946
2943cbbccdb3c/image.jpg"
      },
      {
        "RSN": "2e0c88c8-8a01-41eb-9b04-644ec817e9a1",
        "DisplayValue": null,
        "DisplayOrder": 4,
        "FeaturedImage": false,
        "ImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/gallery/202f65d1f5fe40bbbe5
9980d5699d009/image.jpg"
      }
    ],
    "FeaturedImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/offer/685cee0d6e654266a33fa
74de8fa8f92/image.jpg",
    "OfferWebUrl": "https://test.demoloyalty.simplicitycrm.com//offer-
details.aspx?r=685cee0d6e654266a33fa74de8fa8f92"
  }
]
},
"ErrorCode": 0,
"ErrorDescription": null
}
```

5.12 Member featured offers

Offers for a specified member can be retrieved by making an authenticated GET request to the following resource:

members/{ResourceID}/featured-offers

This service supports paging using the page and pageSize query string parameters. If not specified, the default values are page=1, pageSize=100.

Example:

members/{ResourceID}/featured-offers?page=3&pageSize=10

Sample Request:

```
GET
http://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/f1a7884632
5241758c23b42090d14223/featured-offers?page=1&pageSize=1 HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 2322
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Date: Tue, 12 Jun 2018 06:08:15 GMT
```

```
{
  "Data": {
    "PageNumber": 1,
    "PageStart": 1,
    "PageEnd": 1,
    "PageRowCount": 1,
    "TotalPages": 2,
    "TotalRows": 2,
    "Items": [
      {
        "RSN": "50dc1857-ce00-4dd2-a47f-22e30667a149",
        "DisplayValue": "IPHONE 6S 64GB",
        "OfferDetailHeading": "",
        "OfferDetailText": "Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur.",
        "OfferName": "IPHONE 6S 64GB",
```



```

"Active": true,
"ActiveFrom": "2017-06-21T00:00:00",
"ActiveTo": "9999-12-31T23:59:59",
"Visible": true,
"VisibleFrom": "2017-06-21T00:00:00",
"DisplayOrder": 10,
"ProductCode": "414669-6442",
"GalleryImages": [
  {
    "RSN": "664c53e1-2b54-4452-bc37-48aab12411c9",
    "DisplayValue": null,
    "DisplayOrder": 1,
    "FeaturedImage": false,
    "ImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/gallery/1740b8f3cf33429eae6
86cbba54a431a/image.jpg"
  },
  {
    "RSN": "ea559566-64a1-4c82-b8ba-07044cc59c0f",
    "DisplayValue": null,
    "DisplayOrder": 3,
    "FeaturedImage": false,
    "ImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/gallery/b700eebcd3ac4ac4b4b
a92a7bac726bd/image.jpg"
  },
  {
    "RSN": "b7c30d0e-b5c9-4b79-a29f-84ee26ded2fe",
    "DisplayValue": null,
    "DisplayOrder": 2,
    "FeaturedImage": true,
    "ImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/gallery/b127cd7462bc446db01
3133023033018/image.jpg"
  }
],
"FeaturedImageUrl":
"https://test.demoloyalty.simplicitycrm.com//Media/Content/offer/50dc1857ce004dd2a47f2
2e30667a149/image.jpg",
"OfferWebUrl": "https://test.demoloyalty.simplicitycrm.com//offer-
details.aspx?r=50dc1857ce004dd2a47f22e30667a149"
}
]
},
"ErrorCode": 0,
"ErrorDescription": null
}

```

5.13 Member orders

Offers for a specified member can be retrieved by making an authenticated GET request to the following resource:

members/{ResourceID}/orders

This service supports paging and date range searching using query string parameters. If not specified, the default values are page=1, pageSize=100, fromDate=1753-01-01T00:00:00.000, toDate=9999-12-31T23:59:59.997.

Example:

members/{ResourceID}/orders?page=3&pageSize=10&fromDate=2016-02-24T00%3A00.000%2B12%3A00&toDate=2016-06-01T00%3A00.000%2B12%3A00

Note: dates are ISO-8601 compliant. See [Date and time formats](#) for a list of accepted formats.

Sample Request:

```
GET
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/f2e6f9ef397843c28e4d050d88ac5e88/orders?page=1&pageSize=2&fromDate=1914-01-01&toDate=2018-08-25
HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Accept: application/json
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 1496
Content-Type: application/json; charset=utf-8
Expires: -1
Server:
Strict-Transport-Security: max-age=31536000
Date: Tue, 12 Jun 2018 22:44:11 GMT
```

```
{
  "Data": {
    "PageNumber": 1,
    "PageStart": 1,
    "PageEnd": 2,
    "PageRowCount": 2,
    "TotalPages": 4,
    "TotalRows": 7,
    "Items": [
      {
        "RSN": "31f74319-0b74-4d33-91bb-086955bf5645",
        "Status": "INPROGRESS",
        "StatusDisplayValue": "In Progress",
        "OrderDate": "2018-06-07T11:00:32",
        "OrderNumber": 943,
        "Items": [
          {
            "RSN": "96fa5d3c-3656-409c-b439-60e05e485c8a",
```

```

        "DisplayValue": "Iphone 1",
        "InvoiceStatus": null,
        "InvoiceStatusDisplayValue": null,
        "OfferName": "Iphone 1",
        "Quantity": 1.00000000,
        "ItemUnitValue": 0.0,
        "ItemUnitValuePoints": 500.00000000
    }
}
},
{
    "RSN": "c6204fd3-ab4d-4bbd-a6b8-4a91ccb1cde3",
    "Status": "INPROGRESS",
    "StatusDisplayValue": "In Progress",
    "OrderDate": "2018-05-31T12:09:30",
    "OrderNumber": 940,
    "Items": [
        {
            "RSN": "5aec46c9-f2b2-4138-aadc-1fb4d4ec3a81",
            "DisplayValue": "IPHONE 6S 64GB",
            "InvoiceStatus": null,
            "InvoiceStatusDisplayValue": null,
            "OfferName": "IPHONE 6S 64GB",
            "Quantity": 1.00000000,
            "ItemUnitValue": 0.0,
            "ItemUnitValuePoints": 1000.00000000
        }
    ]
}
]
},
"ErrorCode": 0,
"ErrorDescription": null
}

```

5.14 Member redirection

The member redirection resource provides the ability for authenticated users to access both secured and public web pages via a consistent interface. Resource URL's and redirection behaviour can be configured, and may be returned as the URL to be used in other API calls. The following resource types are supported:

- Loyalty web site pages requiring member authentication
- Third party integrated web pages that could accept the same encrypted parameter as described in the [Third party member integration](#) section

The member redirection resource URL format is below:

members/{ResourceID}/rd/{RedirectionCode}/{RedirectionResourceID}

The *RedirectionResourceID* parameter is only applicable to particular redirection codes (currently only offer). The following RedirectionCode values are currently configured:

- [contact](#)
Logs user into web site and redirects request to the contact us page
- [my-details](#)
Logs user into web site and redirects request to the account details page
- [offer/{RSN}](#)
Logs user into web site and redirects request to the specific offer page. The RSN is returned in the result from the [Member offers](#) resource

Redirection for a specified member can be initiated by making an authenticated POST request to one of the resources described above:

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/8fb782104e1e4028a9ed8a871e5eb2d0/rd/my-details/ HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
```

Sample Response:

```
HTTP/1.1 302 Found
Cache-Control: no-cache
Pragma: no-cache
Expires: -1
Location: https://test.demoloyalty.simplicitycrm.com//my-account/my-details
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 23:07:46 GMT
Content-Length: 0
```

Note that it is possible for an error to occur during the authentication or handoff process. This may result in an error response as described in the [Error codes](#) section. The error could be included either in the redirect (302) response or as an internal error (500) response.

5.15 Public redirection

The public redirection resource provides the ability for the mobile app to access public web pages via a consistent interface. It also allows the location of web pages to change over time without the need for the mobile app to be republished.

The public redirection resource URL format is below:

[rd/{RedirectionCode}](#)

The following RedirectionCode values are currently configured:

- [about](#)
Redirects user to the publically available about page

- [contact](#)
Redirects user to the publically available contact us page
- [faq](#)
Redirects user to the publically available FAQ page
- [forgotten-password](#)
Redirects user to the publically available forgotten password page
- [privacy-policy](#)
Redirects user to the publically available privacy policy page
- [terms](#)
Redirects user to the publically available terms and conditions page

Redirection can be initiated by making a GET request to one of the resources described above:

Sample Request:

```
GET https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/rd/contact
HTTP/1.1
Host: test.demoloyalty.simplicitycrm.com
```

Sample Response:

```
HTTP/1.1 302 Found
Cache-Control: no-cache
Pragma: no-cache
Expires: -1
Location: https://test.demoloyalty.simplicitycrm.com//contact
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 23:12:13 GMT
Content-Length: 0
```

Note that it is possible for an error to occur during the authentication or handoff process. This may result in an error response as described in the [Error codes](#) section. The error could be included either in the redirect (302) response or as an internal error (500) response.

5.16 Third party member integration

The member redirection resource has the ability to redirect to a URL with an encrypted parameter included in the URL. The encrypted parameter includes the following JSON structure:

```
{
  "MembershipNumber": "0123456789",
  "TokenIssued": "2015-11-26T11:13:01",
  "TokenExpiry": "2015-11-26T12:13:01"
}
```

The MembershipNumber value is used to identify which member is associated with the third party resource (e.g. customer survey).

The TokenExpiry value is used to time bomb the generated URL. Third parties should reject the request if the expiry period has passed. The expiry duration is configurable and is currently set to 1 hour.

The above JSON structure is encrypted using the AES-256 encryption algorithm and Base64 encoded. URL encoding is performed on the resulting value in order to safely include it in a URL. Third parties will need to be provided with the initialisation vector and key to enable decryption. The following pseudocode shows how to decode the parameter:

```
key = Base64Decode("Supplied Base64 encoded key")
iv = Base64Decode("Supplied Base64 encoded IV")
json = Decrypt(key, iv, Base64Decode(UrlDecode(url.querystring["memberKey"])))
```

The following example utilising the redirection resource illustrates how this appears in the resulting URL.

Sample Request:

```
POST
https://test.demoloyalty.simplicitycrm.com/services/mobileapi/201605/members/b5bb1ac05f8e4792bbaff74fdfea64ae/rd/wellness-assessment/ HTTP/1.1
Authorization: OAuth {OAUTH PARAMETERS WITH TOKEN}
Host: test.demoloyalty.simplicitycrm.com
Content-Length: 0
```

Sample Response:

```
HTTP/1.1 302 Found
Cache-Control: no-cache
Pragma: no-cache
Expires: -1
Location: http://3rdpartysurvey.com/test-survey/my-test-survey?v=EuAkk7UZGyWFwsSgrzfmIH2lY7fAEEJ3uqjfreY%2BzWBwjCPUtVx9etHUMv38cm87T7HSIi4pq9j5jBPxTcoUYj1ni%2BQ0aeZog5l%2FbqmurM%2B3pAHyCGWHMIXuRXi9J%2FuELBAFDEMTbDQdV69gmV%2F6pIcq8YcugQ1NxveIL10tH6lrFqwEwo02%2B5VNKtaPOUL6
Server:
Strict-Transport-Security: max-age=31536000
Date: Thu, 29 Jun 2017 23:28:22 GMT
Content-Length: 0
```

5.16.1 Decryption example

The following step by step decrypting example is provided to assist in determining any compatibility issues. The following values will be used for this example:

Name	Value
Key (Base64 encoded)	1bCJKX8uq85zSthCUcnVA+oQt+5xOiV9XjAmr3yqExA=
IV (Base64 encoded)	Rt2/ZNy/zuZ/zZ5dqkzO2A==
memberKey (URL encoded)	UZYzHHmrBc0%2F2shmCxT8qhYukpLAY0aW5OOQp3%2BpK4IFIWf7m6oZsdfAjDzEXhiOb67X18RGLZYdPDwQs8Srto6qpdizBuT50WiDNvoDXdsusULYnB7cbG%2B23LFO2raQgxocp7Uj9kKBIATqEX%2FHMnYCdSLCaytj%2FZL45mOZjIw%3D

1. Perform a Base64 decode on both the Key and IV. Decoding the values will result in the following data:
 Key: 0xDC6AEC67422553C07BFB0FE1D6FF752EF16C53006268DD0F6913E8262AAF168F
 IV: 0x064EDC845CF8E6A811FCBB3085A541EE

2. Perform a URL decode on the memberKey. Decoding the memberKey will result in the following string:

 UZyZHHmrBc0/2shmCxT8qhYukpLAY0aW5OOQp3+pK4IFIWf7m6oZsdfAjDzEXhiOb67X18RGLZYdPDwQs8Srto6qpdizBuT50WiDNvoDXdsusULYnB7cbG+23LFO2raQgxocp7Uj9kKBIATqEX/HMnYCdSLCaytj/ZL45mOZjIw=

3. Perform a Base64 decode on the value from step 2. Decoding the above value will result in the following data:

 0x5196331C79AB05CD3FDAC8660B14FCAA162E9292C0634696E4E390A77FA92B82059567FB9BAA19B1D7C08C3CC45E188E6FAED7D7C4462D961D3C3C10B3C4ABB68EAAA5D8B306E4F9D1688336FA035DDB2EB142D89C1EDC6C6FB6DCB14EDAB690831A1CA7B523F642812004EA117FC73276027522C26B2B63FD92F8E663998C8C

4. Decrypt the data from step 3 using Rijndael cipher with the following parameters (AES-256):
 - a. Key size = 256
 - b. Block size = 128
 - c. Padding = PKCS7
 - d. Mode = CBC

Decrypting the above value will result in the following value:

```
{"MembershipNumber": "0123456789", "TokenIssued": "2015-11-26T11:13:01+13:00", "TokenExpiry": "2015-11-26T12:13:01+13:00"}
```

6. Appendices

6.1 Appendix A: Date and time

6.1.1 Date and time range

The following table shows the accepted date range:

	Value (yyyy-MM-ddTHH:mm:ss.fff)
Minimum date	1753-01-01T00:00:00.000
Maximum date	9999-12-31T23:59:59.997

Values are in the hosting server's local time zone. Values passed to the server with a specified time zone must account for conversion to the server's local time zone.

Example: If the server's local time zone is GMT+02:00, then the value "9999-12-31T23:59:59.997+11:00" will exceed the maximum date. See [Date and time formats](#) for further examples.

Do **NOT** assume your hosting server is in GMT+02:00, contact Simplicity CRM for details of your implementation.

6.1.2 Date and time formats

All date formats must be ISO-8601 compliant. The maximum level of granularity is to the nearest 3 milliseconds. The following table shows accepted date formats:

Format	Example	Computed (assuming server in GMT)
yyyy	2015	2015-01-01T00:00:00.000
yyyy-MM	2015-11	2015-11-01T00:00:00.000
yyyy-MMzzz	2015-11+06:00	2015-11-01T06:00:00.000
yyyy-MM-dd	2015-11-19	2015-11-19T00:00:00.000
yyyy-MM-ddzzz	2015-11-19+12:00	2015-11-19T12:00:00.000
yyyy-MM-ddTHH:mm	2016-11-24T08:13	2016-11-24T08:13:00.000
yyyy-MM-ddTHH:mmzzz	2016-11-24T08:13-11:00	2016-11-23T21:13:00.000
yyyy-MM-ddTHH:mm:ss	2016-11-24T10:25:57	2016-11-24T10:25:57.000
yyyy-MM-ddTHH:mm:sszzz	2016-11-24T10:25:57+03:00	2016-11-24T13:25:57.000
yyyy-MM-ddTHH:mm:ss.fff	2016-11-24T08:13:42.997	2016-11-24T08:13:42.997
yyyy-MM-ddTHH:mm:ss.fffzzz	2016-11-24T08:13:42.997+12:00	2016-11-24T20:13:42.997

All values passed without a time zone specified are computed in the hosting server's local time zone.

Do **NOT** assume your hosting server is in GMT, contact Simplicity CRM for details of your implementation.