



Open API Project

Aepona v2.0 Consent REST

Document Version 1.2

Document Revision History

Rev #	Date	Description
1.0	July 29 2013	Initial version, based on Aepona TWS 3.4 Consent v2.0 API Guide doc v1.7.
1.1	Nov 1 2013	Updated URI and code examples, to match production platform.
1.2	Nov 14 2013	Further updates to match updated production platform.

Copyright

2013

© Aepona Limited,

Beacon House,

Clarendon Dock,

Belfast BT1 3BG

All rights reserved. This document or any part thereof may not, without the written consent of Aepona Limited, be copied, reprinted or reproduced in any material form including but not limited to photocopying, transcribing, transmitting or storing it in any medium or translating it into any language, in any form or by any means, be it electronic, mechanical, xerographic, optical, magnetic or otherwise.

The information contained in this document is proprietary and confidential and all copyright, trademarks, trade names, patents and other intellectual property rights in the documentation are the exclusive property of Aepona Limited unless otherwise specified. The information (including but not limited to data, drawings, specification, documentation, software listings, source or object code) shall not at any time be disclosed directly or indirectly to any third party without Aepona Limited's prior written consent.

The information contained herein is believed to be accurate and reliable. Aepona Limited accepts no responsibility for its use by any means or in any way whatsoever. Aepona Limited shall not be liable for any expenses, costs by damage that may result from the use of the information contained within this document. The information contained herein is subject to change without notice.

Table of Contents

1	Consent REST Overview.....	6
2	Authentication.....	6
3	Methods and URI.....	6
4	Create Subscriber Consent.....	7
4.1	Request.....	7
4.1.1	Request Parameters.....	8
4.2	Response.....	9
5	Update Subscriber Consent.....	9
5.1	Request.....	9
5.1.1	Request Parameters.....	10
5.2	Response.....	10
6	Delete Subscriber Consent.....	11
6.1	Request.....	11
6.1.1	Request Parameters.....	12
6.2	Response.....	12
7	Request Subscriber Consent.....	13
7.1	Request.....	13
7.1.1	Request Parameters.....	13
7.2	Response.....	14
7.2.1	Response Parameters.....	14
8	Query Subscriber Consent.....	15
8.1	Request.....	15
8.1.1	Request Parameters.....	15
8.2	Response.....	16
8.2.1	Response Parameters.....	16
9	Consent Callback.....	16
9.1	Request.....	16
9.1.1	Request Parameters.....	17

9.2	Response.....	17
9.2.1	Response Parameters.....	18
10	Status Responses.....	18
11	Response Codes & Exceptions.....	18
11.1	Response Codes.....	18
11.2	Exceptions.....	19
11.2.1	Service Exceptions.....	19
11.2.2	Policy Exceptions.....	20
12	Sandbox Service.....	21
12.1	Create Consent.....	21
12.2	Update Consent.....	21
12.3	Delete Consent.....	21
12.4	Request Consent.....	21
12.5	Query Consent.....	21
13	HELP & INFO Message Support.....	22
13.1	Help & Info Policy.....	22

1 Consent REST Overview

The Consent interface allows an application to request the consent of a subscriber before it can use various enablers to perform actions against that subscriber, e.g. to obtain the subscribers location or send the subscriber an SMS or MMS. In addition the API allows direct management of subscriber consent via the 'Deposit API' methods: create, update and delete consent, that may be used when the application is trusted to manage consent, e.g. via its own subscriber dialogue mechanism. Access to these methods for applications will be dependent on operation set policies. Consent is managed at the application level.

Consent Callback is described in section 9.

A sandbox service is available. The scenarios are described in section 12.

! Please note that throughout this document, the examples may be shown WITHOUT URL encoding for readability purposes. For example, if the address tel: +12345678900 is in the example, this should be encoded as tel:%3A%2B12345678900, where the character ':' is denoted by '%3A' and the character '+' by '%2B'.

! Data types indicated as XSD in parameter tables refer to the standard XSD type. Relevant information may be obtained from <http://www.w3.org/2001/XMLSchema>.

2 Authentication

A server side certificate is required plus HTTP Basic Authentication.

For more information, refer to the 'Developer Access' section in the 'OneAPI v2.0 Common Information Guide'.

3 Methods and URI

Consent may only be accessed via the REST API (described in this document). The following methods are available, with their respective resource URI:

- Create ('deposit') consent of a subscriber - HTTP POST command - section 4

https://developerportal.uscellular.com/Privacy

- Update consent previously deposited via the create operation - HTTP PUT command - section 5

https://developerportal.uscellular.com/Privacy/?**address={address}&status={status}&expiryTime={expiryTime}**

- Delete consent previously deposited via the create operation - HTTP DELETE command - section 6

https://developerportal.uscellular.com/Privacy/?address={address}

- Request the consent of a subscriber - HTTP POST command - section 7

https://developerportal.uscellular.com/Privacy

- Query the status of the subscriber consent - HTTP GET command - section 8

https://developerportal.uscellular.com/Privacy/?address={address}

! The Consent API supports application/x-www-form-urlencoded for POST operations. The response content type is application/XML.

4 Create Subscriber Consent

This method allows the application to 'deposit' consent that is managed by a mechanism outside of TWS.

4.1 Request

```
POST https://developerportal.uscellular.com/services/Privacy/ HTTP/1.1
```

```
Content-Type: application/x-www-form-urlencoded
```

```
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
```

```
Content-Length: 80
```

```
Host: developerportal.uscellular.com
```

```
Connection: Keep-Alive
```

```
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
```

```
expiryTime=100&address=tel%3A
%2B12345600001&operation=createConsent&status=ALLOWED
```

! Query strings should be URL encoded.

4.1.1 Request Parameters

Table 1: Create Subscriber Consent - Request Parameters

Parameter	Data Type	Description	Optional
address	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, whose consent request is being deposited, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be must be URL-escaped where %3A represents ':' and %2B represents '+'.</p> <p>Example: %3A%2B15087300001</p>	No
operation	xsd:string	Must be 'createConsent' to distinguish from the requestConsent POST method.	No
status	status	The consent status. (See table 9 for details).	No
expiryTime	xsd:int	The number of hours until the consent expires.	No

4.2 Response

```
HTTP/1.1 204 No Content
Date: Thu, 31 Oct 2013 16:17:25 GMT
Server: Jetty(6.1.x)
Content-Type: application/xml
Host: developerportal.uscellular.com
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
expiryTime: 100
operation: createConsent
breadcrumbId: ID-ase3-43542-1383117279982-14-10
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
address: tel:+12345600001
status: ALLOWED
Content-Length: 0
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
```

5 Update Subscriber Consent

Allows the application to update consent previously deposited via the create operation.

5.1 Request

```
PUT https://developerportal.uscellular.com/services/Privacy/?
expiryTime=100&address=tel%3A%2B12345600001&status=DENIED HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
Content-Length: 0
Host: developerportal.uscellular.com
Connection: Keep-Alive
```

User-Agent: Apache-HttpClient/4.1.1 (java 1.5)

! Query strings should be URL encoded.

5.1.1 Request Parameters

Table 2: Update Subscriber Consent - Request Parameters

Parameter	Data Type	Description	Optional
address	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, whose consent is being updated, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be must be URL-escaped where %3A represents ':' and %2B represents '+'. Example: %3A%2B15087300001</p>	No
status	status	The consent status. (See table 9 for details).	No
expiryTime	xsd:int	The number of hours until the consent expires.	No

5.2 Response

HTTP/1.1 204 No Content

Date: Thu, 31 Oct 2013 16:22:25 GMT

Server: Jetty(6.1.x)

```
Content-Type: application/xml
Host: developerportal.uscellular.com
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
breadcrumbId: ID-ase2-51909-1383117280141-22-14
status: DENIED
expiryTime: 100
X-Forwarded-Server: developerportal.uscellular.com
address: tel:+12345600001
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
Content-Length: 0
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
```

6 Delete Subscriber Consent

Allows the application to delete consent previously deposited via the create operation.

6.1 Request

```
DELETE https://developerportal.uscellular.com/services/Privacy/?address=tel%3A
%2B12345600001 HTTP/1.1
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
Host: developerportal.uscellular.com
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
```

! Query strings should be URL encoded.

6.1.1 Request Parameters

Table 3: Delete Subscriber Consent Request Parameters

Parameter	Data Type	Description	Optional
address	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, the consent deposit for whom is being deleted, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be must be URL-escaped where %3A represents ':' and %2B represents '+'. Example: %3A%2B15087300001</p>	No

6.2 Response

```

HTTP/1.1 204 No Content
Date: Thu, 31 Oct 2013 16:25:08 GMT
Server: Jetty(6.1.x)
Content-Type: application/xml
Host: developerportal.uscellular.com
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
address: tel:+12345600001
breadcrumbId: ID-ase3-43542-1383117279982-14-13
Content-Length: 0
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive

```

7 Request Subscriber Consent

Allows the application to seek consent from a subscriber for various actions e.g. to access their location or to be able to send an SMS to the subscriber.

7.1 Request

```
POST https://developerportal.uscellular.com/services/Privacy/?callbackUrl=http%3A%2F%2Fexample.com%2FMockRestService%2Frest&address=tel%3A%2B12345600001 HTTP/1.1
```

```
Content-Type: application/x-www-form-urlencoded
```

```
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
```

! Query strings should be URL encoded. The address parameter is converted to tel%3A%2B12345600001 and the callback URL parameter is converted to http%3A%2F%2Fwww.partnersite.com%2FprivacyReceiver.

7.1.1 Request Parameters

Table 4: Request Subscriber Consent Request Parameters

Parameter	Data Type	Description	Optional
address	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, whose consent is requested, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be must be URL-escaped where %3A represents ':' and %2B</p>	No

		represents '+'. Example: %3A%2B15087300001	
operation	xsd:string	This parameter is used to distinguish this operation from the createConsent operation. It may be absent, or anything other than 'createConsent'	Yes
callbackUrl	xsd:anyURI	The URL to be invoked when the subscriber has given/withheld their consent.	No

7.2 Response

```

HTTP/1.1 200 OK
Date: Thu, 31 Oct 2013 16:25:08 GMT
Server: Jetty(6.1.x)
Content-Type: application/xml
Host: developerportal.uscellular.com
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Consent status="PENDING"/>

```

7.2.1 Response Parameters

Table 5: Request Subscriber Consent Response Parameters

Parameter	Data Type	Description	Optional
status	status	The consent status. (See table 9 for details).	No

8 Query Subscriber Consent

Allows the application to query the response of a previous consent request.

8.1 Request

```
GET https://developerportal.uscellular.com/services/Privacy/?address=tel%3A%2B12345600001
HTTP/1.1
```

```
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
```

! Query strings should be URL encoded.

8.1.1 Request Parameters

Table 6: Query Subscriber Consent - Request Parameters

Parameter	Data Type	Description	Optional
address	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be must be URL-escaped where %3A represents ':' and %2B represents '+'.</p> <p>Example: %3A%2B15087300001</p>	No

8.2 Response

```

HTTP/1.1 200 OK
Date: Thu, 31 Oct 2013 16:25:08 GMT
Server: Jetty(6.1.x)
Content-Type: application/xml
Host: developerportal.uscellular.com
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic QWVwb25hVfdghfdghdfghdfgWF6eHN3MjNI
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Consent status="PENDING"/>

```

8.2.1 Response Parameters

Table 7: Query Subscriber Consent - Response Parameters

Parameter	Data Type	Description	Optional
status	status	The consent status. (See table 9 for details).	No

9 Consent Callback

When the subscriber gives their consent or withholds it, your service at the callback URL (which you specified in your original consent request) is invoked. Your service must respond with "204 No Content".

9.1 Request

```

POST https://<privacyReceiver>
HTTP/1.1

Accept: application/xml
Content-type: application/xml

```



```
<?xml version="1.0" encoding="UTF-8"?>
<privacyReceipt>
  <subscriber>tel:+15087300001</subscriber>
  <status>ALLOWED</status>
</privacyReceipt>
```

9.1.1 Request Parameters

Table 8: Consent Callback - Request Parameters (consentEventType)

Parameter	Data Type	Description	Optional
subscriber	xsd:anyURI	<p>The address (CTN) of U.S. Cellular subscriber, in the format comprising:</p> <ol style="list-style-type: none"> 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001 <p>The address must be URL-escaped where %3A represents ':' and %2B represents '+'. Example: %3A%2B15087300001</p>	No
status	status	Specifies if the subscriber has granted or denied consent. (See table 9 for details).	No

9.2 Response

204 No Content

9.2.1 Response Parameters

N/A

10 Status Responses

The following Consent Response statuses may be returned:

Table 9: Status Types

Status	Description
PENDING	The subscriber has been notified by text message that the application wishes to use their number. They have not yet replied and the request has not yet expired.
ALLOWED	The subscriber has replied to the notification text message and has chosen to allow the application to use their number.
DENIED	The subscriber has replied to the notification text message and has chosen not to allow the application to use their number.
EXPIRED	The subscriber has not replied to the notification text message, and the time allowed for replying has passed or the Consent session has expired

11 Response Codes & Exceptions

11.1 Response Codes

HTTP response codes are used to indicate:

- **200** – Success!
- **400** – Bad request; check the error message for details
- **401** – Authentication failure, check your authentication details
- **403** – Forbidden; please provide authentication credentials
- **404** – Not found: mistake in the host or path of the service URI
- **405** – Method not supported: for example you mistakenly used a HTTP GET to create an SMS instead of a POST

- **500** – The server encountered an unexpected condition. It could be incorrect authentication details or limited user permission
- **503** – Server busy and service unavailable. Please retry the request.

For more details on these, refer to <http://www.ietf.org/rfc/rfc2616.txt>.

11.2 Exceptions

```
HTTP/1.1 403 Forbidden
```

```
Content-Type: application/xml
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<error> A policy error occurred. Error code is POL-014: Destination White List is enforced and address is not in Destination White List.</error>
```

This section lists the available error codes, the possible reasons why the exception may have occurred, and possible solutions.

11.2.1 Service Exceptions

The following exceptions may be thrown when an operation fails:

Table 10: Service Error Codes

Error	Explanation
SVC0001 – Service error occurred	A service-related error has occurred as a result of a client invocation on the service. This category can be used for implementation-specific errors. Contact the support team.
SVC0002 – Invalid input value	An input parameter value is not of the expected type. Check the parameter types and re-submit your request.
SVC0004 – No valid address(es)	The requested terminal device address does not exist. Use an address that exists.

11.2.2 Policy Exceptions

A policy exception means that the request syntax is valid, however an operator policy has been broken.

POL0001 - Policy error occurred

The above exception may be thrown to indicate a fault relating to a policy associated with the service. This category can be used for implementation-specific errors such as:

Table 11: Policy Error Codes

Error	Explanation
POL-006: TPA exceeded its maximum allowed rate of transactions	The maximum rate of transactions is exceeded. Ensure that the rate of your requests is within the limits set up in your SLA, e.g. 10 TPS (Transactions Per Second).
POL-008: TPA is invalid	The Third Party Application authentication details are incorrect. Check your basic authentication username and password are correct and re-submit your request.
POL-014: White List is enforced, and address is not in White List	A white list is enforced and the number is not in the white list. Check your SLA details.
POL-015: Black List is enforced, and address is in Black List	A black list is enforced and the number is in the black list. Check you SLA details.
POL-016: Max Requests is enforced, and max requests has been exceeded	The maximum number of requests for this service is exceeded. Contact the support team.
POL-017: Operation is not allowed	The method/operation is not supported in your current SLA. Check your SLA and use a method that is supported.

12 Sandbox Service

The Consent service contains a sandbox for use in testing applications using consent, available at <https://developerportal.uscellular.com/services/PrivacySandbox/>. This currently uses built in responses and does not make use of the sandbox data service.

The behaviour of the sandbox is covered in the following subsections.

12.1 Create Consent

Creates consent using the specified request parameters.

All statuses are expired and removed after 5 minutes.

12.2 Update Consent

Updates (existing) consent using the specified request parameters.

12.3 Delete Consent

Deletes (existing) consent using the specified request parameters.

12.4 Request Consent

All requests will return a PENDING status.

In production, this status would change to ALLOWED or DENIED depending on the response from the subscriber. Please use the Create Consent and Update Consent methods to view these response in Sandbox.

All statuses are expired and removed after 5 minutes.

12.5 Query Consent

This returns the current status according to the rules above for requesting consent.

All statuses are expired and removed after 5 minutes.

If the requested subscriber does not have a consent state associated, a 'Consent Not Found' error will be returned.

13 HELP & INFO Message Support

13.1 Help & Info Policy

An application can choose to support HELP & INFO messages from the subscriber or rely on more generic HELP & INFO messages configured through the Consent service. To support HELP & INFO messages an application must request the administrator to enable the "Help Info Messages" Policy. If an application chooses to support HELP & INFO messages and a HELP or INFO message is received from a subscriber an event will be published to the original callbackUrl appended with a "/keyword".

If the original callbackUrl was <http://callbackUrl/appName>. The HELP or INFO event will be published to <http://callbackUrl/appName/keyword>.

```
Host: www.partnersite.com
Accept: text/plain
Content-type: application/xml
<?xml version="1.0" encoding="UTF-8"?>
<privacyReceipt>
  <subscriber>tel:+447990123456</subscriber>
  <messageType>messageTypeHelp</messageType>
  <message>HELP blah, blah, blah</message>
</privacyReceipt>
```