

DRAFT



**RadioTAG Technical Specification**  
**RTAG01 V1.0.0 DRAFT 7 (2015-06)**

Bookmarking for hybrid radio devices

DRAFT



### **The RadioDNS Project**

<http://radiodns.org/>  
[feedback@radiodns.org](mailto:feedback@radiodns.org)

### **Important notice**

Individual copies of the present document can be downloaded from <http://radiodns.org/docs>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF).

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other RadioDNS documents is available at <http://radiodns.org/docs>

If you find errors in the present document, please send your comment to the document editors  
[feedback@radiodns.org](mailto:feedback@radiodns.org)

### **Copyright Notification**

No part may be reproduced except as authorised by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

Copyright © 2015 The RadioDNS Project.  
All rights reserved.

HD Radio™ is a trademark of iBiquity Digital Corporation.

# Contents

Intellectual Property Rights .....	4
Foreword .....	4
1. Scope .....	5
2. References .....	5
3. Definitions and abbreviations .....	6
3.1 Definitions .....	6
3.2 Abbreviations .....	6
4. How to read this document .....	7
5. Concepts .....	7
5.1 Application Discovery .....	7
5.2 HTTPS .....	7
5.3 Tag Requests .....	8
5.4 Tag responses .....	8
5.5 Clients, receivers, radios and devices .....	8
5.6 The tag service .....	8
5.7 Base path .....	8
5.8 Authentication .....	8
5.9 Authorisation modes .....	8
5.9.1 No identity .....	9
5.9.2 Client mode .....	9
5.9.3 User mode .....	10
5.10 Authorisation .....	10
6. API .....	11
6.1 POST /tag .....	11
6.1.1 Request .....	11
6.1.2 Response .....	11
6.1.3 Example 1 - POST /tag with no token .....	12
6.1.4 Example 2 - POST /tag with a valid client token .....	12
6.1.5 Example 3 - POST /tag with a valid user token .....	13
6.1.6 Example 4 - POST /tag with a service that does not provide client tagging .....	14
6.1.7 Example 5 - POST /tag with a service that provides only unidentified tagging .....	15
6.2 GET /tags .....	16
6.2.1 Request .....	16
6.2.2 Response .....	17
7. Data formats .....	19
7.1 Tag data .....	19
7.2 Tags data .....	20
8. Limits .....	22
8.1 Data elements .....	22
9. Appendix .....	23
9.1 Narratives .....	23
9.1.1 From client to user pairing .....	23
9.1.2 Unidentified to user identity .....	32
History .....	38

# Intellectual Property Rights

The contributors to the present document have declared that they have no IPRs in respect of their contributions. However no investigation, including IPR searches, has been carried out, so no guarantee can be given as to the existence of other IPRs which are, or may be, or may become, essential to the present document.

## Foreword

Radio is a complimentary media format in that it can be consumed whilst performing other tasks unlike visual-based mediums. It is content producers job to elicit response and encourage further engagement with their content, be it editorial or commercial, but due to the nature of radio consumption - such as in a vehicle - it is sometimes difficult to remember or record details for further investigation.

RadioTAG aims to deliver a simplified interaction mechanic for radio, particularly in challenging environment such as driving a vehicle, where a user can perform a single physical action to express interest in the content they are hearing on the radio and time shift their exploration of that content to a later point in time.

# 1. Scope

This document specifies version 1.0.0 of the RadioTAG protocol.

The RadioTAG protocol defines how a client (e.g. an IP-enabled radio) discovers whether a broadcaster supports RadioTAG and if so how it then communicates with a broadcaster-provided web service to record the time and radio service being listened to.

The protocol defines how the Cross Platform Authentication (CPA) protocol is implemented within this protocol, to allow the client to obtain authorisation to store data on the server and how it can become paired with a user account so that data can be accessed via the web.

The protocol also defines the format and content of the requests and responses that pass between the client and server.

# 2. References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] ETSI TS 103 270: "RadioDNS Hybrid Radio; Hybrid lookup for radio services"

[2] RFC 2818: "HTTP Over TLS"

[3] ETSI TS 102 818: "Hybrid Digital Radio (DAB, DRM, RadioDNS); XML Specification for Service and Programme Information (SPI)"

[4] RFC 4287: "The Atom Syndication Format"

[5] EBU TECH 3366: "The Cross Platform Authentication Protocol", Version 1.0

[6] [http://www.opensearch.org/Specifications/OpenSearch/1.1/Draft\\_5](http://www.opensearch.org/Specifications/OpenSearch/1.1/Draft_5), "OpenSearch 1.1 Draft 5"

[7] ETSI EN 300 401: "Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers"

[8] RFC 2616, "Hypertext Transfer Protocol - HTTP/1.1"

## 3. Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

**Client:** The device or user agent which interacts with the RadioTAG service

**Client identity:** An identity associated only with a specific client and not with a user account

**Cross Platform Authentication:** an open standard proposed by the EBU to provide a secure framework for authentication on devices such as smart radio and televisions.

**User identity:** An identity where a client has been associated with a user account, and which can then be accessed from any client which has been similarly associated

**Token:** An authorisation token which permits you to create a tag

**Unix Time:** The number of seconds elapsed since midnight Coordinated Universal Time (UTC) on January 1, 1970, not counting leap seconds

### 3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CPA	Cross Platform Authentication
DAB	Digital Audio Broadcasting
DNS	Domain Name Server
EPG	Electronic Programme Guide
HTTP	Hyper Text Transfer Protocol
RDS	Radio Data System
URI	Uniform Resource Identifier
URL	Uniform Resource Location
XML	eXtensible Markup Language

## 4. How to read this document

The document starts with an overview of the concepts underlying the RadioTAG protocol.

To get an idea of how the RadioTAG protocol works in practice, read the two narratives in the Appendix. These step through the two most common scenarios to show what is communicated between the client (radio) and the tag service at the HTTP level. This section is particularly useful for developers who want to understand how the various parts of the API work together.

For full details about each endpoint provided by the tag service, read the API section.

Finally, to see how the Atom format is used in tag responses, what each element contains and what limits apply, see data formats.

## 5. Concepts

A client sends a tag request to a tag service, specifying a time and radio service. The tag service responds by sending a tag entry containing relevant metadata. The tag data may be stored on the server and may be viewed on the client or on the web or be used for another application.

### 5.1 Application Discovery

A client must be capable of resolving the authoritative FQDN for a service via the methodology defined in the RadioDNS specification[1].

Application lookup may then be performed against this FQDN by means of a DNS SRV Record request using the service name `radiotag`.

If at least one SRV record is successfully resolved, this service supports the RadioTAG application, accessed on the host and port indicated in the relevant SRV record. For example, for a query made to:

```
_radiotag._tcp.bbc.co.uk.
```

Using the `nslookup` tool, this would yield the following SRV record:

```
service = 0 100 80 radiotag.bbc.co.uk.
```

This indicates that the RadioTAG application can be accessed on the FQDN `radiotag.bbc.co.uk`, port 80.

Note that more than one SRV record may be returned with differing values. This can be used for load balancing purposes by providing different FQDNs/Ports with different priorities/weightings. See the SRV record specification for a more detailed explanation on handling SRV resolution results.

### 5.2 HTTPS

Any implementation of RadioTAG MUST require the use of Transport Layer Security (TLS)[2].

The appropriate version (or versions) of TLS will vary over time, based on the widespread deployment and known security vulnerabilities.

Implementations MAY also support additional transport-layer security mechanisms that meet their security requirements.

## 5.3 Tag Requests

A tag request specifies a time and bearer. The time is specified using seconds since Jan 1 1970, i.e. the Unix epoch. The bearer is specified using the bearerURI format defined in ETSI TS 103 270[3].

An optional attribute for the time's source is defined, which allows the broadcaster to make a more informed decision on the accuracy of the time value supplied and therefore the event that this request is in relation to.

How that information is interpreted is up to the broadcaster.

## 5.4 Tag responses

The content of the response to a tag request is up to the broadcaster but must follow the Atom Syndication Format[4] as specified below. A tag response could contain programme, now playing metadata, an advertising message or the response to a call for action.

## 5.5 Clients, receivers, radios and devices

In this specification, the **client** is any device or software program that implements the client side of the RadioTAG specification. While this would most commonly be an IP-enabled radio, it could also be, say, an application running on a mobile phone or in a web browser.

## 5.6 The tag service

The **tag service** is the web service provided by the broadcaster to respond to client requests. It must implement a number of endpoints depending on the authorisation modes it provides.

## 5.7 Base path

All end points defined in this document are intended to be located on the tag service host within the base path:

```
/radiodns/tag/1/
```

## 5.8 Authentication

This specification does not directly provide an authentication solution. The tag application should be secured using the processes detailed in the CPA specification[5]. This specification simply aims to highlight the important components and the direct impact they have on the implementation of tagging.

## 5.9 Authorisation modes

There are three modes a tag service can provide:

- unidentified
- client
- user

*User* in this context refers to an authenticated user account.



## DRAFT

The authorisation modes are distinguished by whether or not tags are retrievable on the device or on the web and by whether you need a user account on the broadcaster's web service. The table below summarises the differences:

Authorisation mode	Tag list on device	Tag list on web	Account needed
Unidentified	No	No	No
Client	Yes	No	No
User	Yes	Yes	Yes

These authorisation modes can be provided in a number of combinations. For example, a service may offer unidentified tagging by default which can be upgraded to user mode tagging or it may support tagging out of the box (client mode) with no option to pair the device to an online user account. These are the possible combinations:

- Unidentified only
- Unidentified upgradeable to user
- Client only
- Client upgradeable to user

The client and user modes map directly to the equivalent authorisation modes offered within the CPA specification[5].

There is no defined combination to upgrade from unidentified to client mode. This is an automated operation, no end user input is proposed in this process. Any client capable of upgrading to client mode will do so automatically and therefore never originate in an unidentified state.

### 5.9.1 No identity

Unidentified tagging is the minimal mode of service. The broadcaster must provide the following endpoint:

- POST /tag

A POST to this endpoint should return metadata relevant to the radio service and time specified in the request. Tags are *not* stored on the server so it is not possible to retrieve a list of tags on the client.

### 5.9.2 Client mode

Client identity is designed to provide an "out-of-the-box" experience without the user having to create an account and pair the client. The protocol enables the client to become authorised to store tags on the server without being associated with an authenticated user account.

To indicate that it supports client identity, the server must issue a `WWW-Authenticate` response-header with a client mode to an unauthorised request to POST /tag. It must provide the following endpoints:

- POST /tag
- GET /tags

In addition it must adhere fully to Client mode within the CPA specification[5].

Tags are stored on the server. The server must be able to store at least 10 tags per client. There is no upper limit. A typical implementation would store the tags as a FIFO list. When the list is full, a new tag would replace the oldest in the list.

## DRAFT

A client should implement an interface to display the current list of tags associated with it as returned by the `GET /tags` method.

Note that with client identification, the client stores a token which uniquely identifies it to the tag service. If that identity is reset by the client deleting the token, any tags which have been submitted against it are effectively orphaned.

### 5.9.3 User mode

User identity is where the client has been paired to an authenticated user's account on a tag service. The same limits apply as for client identification, though a typical implementation will not put any limit on how many tags a user can create.

A tag service that enables tagging with a user identity must provide the following endpoints:

- `POST /tag`
- `GET /tags`

In addition it must adhere fully to the User mode within the CPA specification[5].

### 5.10 Authorisation

To store or retrieve anything at the tag service, a client needs a token. A valid token authorises the client to perform a specific set of actions. In the case of RadioTAG, those actions are to create a tag or get a list of tags for either a client identity or user account identity.

To obtain a token, the client must follow the process detailed within the CPA specification[5].

## 6. API

Requests pass information in a combination of HTTP headers and form encoded POST parameters.

Responses pass data back in a combination of HTTP headers and XML.

While headers are shown here in a canonical form, due to the fact that proxies and other intermediaries may adjust HTTP headers, both client and server implementations should be prepared to accept header keys in any mixture of upper and lower case. One common way to handle this is to lowercase all header keys on reading. Header values on the other hand should not be changed by intermediaries and should be read as is.

UTF-8 is the only supported character set.

### 6.1 POST /tag

#### 6.1.1 Request

##### Headers

Name	Value
Authorization	Not set OR client OR user token

##### Parameters

Name	Value
bearer	RadioDNS bearerURI as defined in ETSI TS 103 270, e.g. "dab:ce1.ce15.c224.0"
time	Whole number of seconds since 00:00a.m Jan 1 1970 UTC (Unix Epoch)
time_source	(Optional) where the time is sourced from, value of either: <i>user</i> - the user has set the client clock manually, <i>broadcast</i> - derived from the bearer's time source (e.g. RDS for FM, the FIG in DAB), <i>ntp</i> - derived from IP source. When not set it is assumed the time is derived from the bearer's time source.

#### 6.1.2 Response

##### Status

HTTP Status Code	HTTP Status	Explanation
200	OK	The service does not store tags but has returned metadata in Atom format
201	Created	The service has stored the requested tag
401	Unauthorized	Unidentified tagging is not supported and the token is missing or does not match either a client or user identify.

## Parameters

Name	Value
Authorization	Not set OR client OR user token

### 6.1.3 Example 1 - POST /tag with no token

#### Request

```
POST /radiodns/tag/1/tag HTTP/1.1↵  
Host: radiotag.bbc.co.uk↵  
Content-Type: application/x-www-form-urlencoded↵  
↵  
bearer=dab%3Ace1.ce15.c224.0&time=1312301004&time_source=broadcast
```

#### Response

```
HTTP/1.1 401 Unauthorized↵  
Date: Tue, 02 Aug 2011 16:03:24 GMT↵  
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://  
ap.bbc.co.uk" modes="client"↵  
Content-Type: text/plain; charset=utf-8↵  
↵  
Must request client token
```

### 6.1.4 Example 2 - POST /tag with a valid client token

#### Request

```
POST /radiodns/tag/1/tag HTTP/1.1↵  
Host: radiotag.bbc.co.uk↵  
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵  
Content-Type: application/x-www-form-urlencoded↵  
↵  
bearer=dab%3Ace1.ce15.c224.0&time=1312301004&time_source=broadcast
```

## DRAFT

### Response

```
HTTP/1.1 201 Created␣
Date: Tue, 02 Aug 2011 16:03:25 GMT␣
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"␣
Content-Type: application/atom+xml; charset=utf-8␣
␣
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>PM</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-08-02T17:03:24+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:fb669d2c-63b3-420b-9dd6-131f5d58e68a</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>PM</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b012wjd3_150_84.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=204"/>
    <id>urn:uuid:fb669d2c-63b3-420b-9dd6-131f5d58e68a</id>
    <updated>2011-08-02T17:03:24+01:00</updated>
    <published>2011-08-02T17:03:24+01:00</published>
    <summary>Eddie Mair presents the day's top stories.</summary>
  </entry>
</feed>
```

Note that the response header contains the WWW-Authenticate header with a user modes value. This will be present only if the service supports user tagging.

### 6.1.5 Example 3 - POST /tag with a valid user token

#### Request

```
POST /radiodns/tag/1/tag HTTP/1.1␣
Host: radiotag.bbc.co.uk␣
Authorization: Bearer kldhvkjxhoiqwyeh3khkj3␣
Content-Type: application/x-www-form-urlencoded␣
␣
bearer=dab%3Ace1.ce15.c224.0&time=1312302129&time_source=broadcast
```

## DRAFT

### Response

```
HTTP/1.1 201 Created↵
Date: Tue, 02 Aug 2011 16:22:09 GMT↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
      xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>PM</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-08-02T17:22:09+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:8facc0c0-ce13-4349-8664-dc71d55c6c97</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>PM</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b012wjd3_150_84.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=1329"/>
    <id>urn:uuid:8facc0c0-ce13-4349-8664-dc71d55c6c97</id>
    <updated>2011-08-02T17:22:09+01:00</updated>
    <published>2011-08-02T17:22:09+01:00</published>
    <summary>Eddie Mair presents the day's top stories.</summary>
  </entry>
</feed>
```

Note that the response header does not contain any WWW-Authenticate headers but does contain the paired user account name.

### 6.1.6 Example 4 - POST /tag with a service that does not provide client tagging

#### Request

```
POST /radiodns/tag/1/tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Content-Type: application/x-www-form-urlencoded↵
↵
bearer=dab%3Ace1.ce15.c224.0&time=1312195118&time_source=broadcast
```

## DRAFT

### Response

```
HTTP/1.1 200 OK␣
Date: Mon, 01 Aug 2011 10:38:38 GMT␣
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"␣
Content-Type: application/atom+xml; charset=utf-8␣
␣
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
      xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Meet David Sedaris</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-08-01T11:38:38+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>Meet David Sedaris</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b01211y4_150_84.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b01211y4?
t=518"/>
    <id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
    <updated>2011-08-01T11:38:38+01:00</updated>
    <published>2011-08-01T11:38:38+01:00</published>
    <summary>'Me Talk Pretty One Day' and 'It's Catching'.</summary>
  </entry>
</feed>
```

### 6.1.7 Example 5 - POST /tag with a service that provides only unidentified tagging

#### Request

```
POST /radiodns/tag/1/tag HTTP/1.1␣
Host: radiotag.bbc.co.uk␣
Content-Type: application/x-www-form-urlencoded␣
␣
bearer=dab%3Ace1.ce15.c224.0&time=1312195118&time_source=broadcast
```

#### Response

```
HTTP/1.1 200 OK␣
Date: Mon, 01 Aug 2011 10:38:38 GMT␣
Content-Type: application/atom+xml; charset=utf-8␣
```

## DRAFT

↵

```
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
      xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Meet David Sedaris</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-08-01T11:38:38+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>Meet David Sedaris</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b01211y4_150_84.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b01211y4?
t=518"/>
    <id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
    <updated>2011-08-01T11:38:38+01:00</updated>
    <published>2011-08-01T11:38:38+01:00</published>
    <summary>'Me Talk Pretty One Day' and 'It's Catching'.</summary>
  </entry>
</feed>
```

Note that no WWW-Authenticate headers are issued.

## 6.2 GET /tags

### 6.2.1 Request

#### Headers

Name	Value
Authorization	client OR user token

#### Parameters

Name	Value
startIndex	(Optional) the 1-based index of the first result
itemsPerPage	(Optional) maximum number of entries to return

If the caller does not specify `startIndex`, it defaults to 1.

If the caller does not specify `itemsPerPage`, the number of entries returned is determined by the server.



## DRAFT

The server specifies the total number of entries that can be returned in the result set using the `totalResults` element (see below).

Note: the `startIndex`, `itemsPerPage` and `totalResults` parameters are based on the OpenSearch specification[6].

### Example

```
GET /radiodns/tag/1/tags HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵
↵
```

## 6.2.2 Response

### Status

HTTP Status Code	HTTP Status	Explanation
200	OK	The request was successful
401	Unauthorized	The token is invalid or the service does not allow storing of tags

### Body

#### Example

```
HTTP/1.1 200 OK↵
Date: Tue, 02 Aug 2011 16:22:08 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Tag List</title>
  <link href="http://radiotag.bbc.co.uk/tags"/>
  <link href="http://radiotag.bbc.co.uk/tags" rel="self"/>
  <updated>2011-08-02T17:22:08+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:6a041e97-65bb-4b12-82da-c1b373731905</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>PM</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
```

**DRAFT**

```
<link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b012wjd3_150_84.jpg"/>
<link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=1328"/>
<id>urn:uuid:9f61f2c1-f3f7-4ff7-bc61-32f0e468054d</id>
<updated>2011-08-02T17:22:08+01:00</updated>
<published>2011-08-02T17:22:08+01:00</published>
<summary>Eddie Mair presents the day's top stories.</summary>
</entry>
</feed>
```

## 7. Data formats

### 7.1 Tag data

All server responses containing tags use the Atom Syndication Format[4] to represent tags, with some extensions under a RadioTAG namespace.

In the following, the element prefix "radiotag:" indicates the RadioTag namespace. All other elements are assumed to be from the Atom namespace.

Atom defines these elements as required:

Element	Description	Max length
id	unique identifier for this tag ref	48
title	broadcaster generated title ref	128
updated	the datetime the tag was modified ref	20

All dates are UTC in ISO format (ISO 8601 or RFC 3339), e.g. 2011-08-08T09:00:00Z.

The RadioTAG specification also requires the following:

Element	Description	Max length
author	name of tag service provider (e.g. BBC, Global)	16
published	the datetime of creation (=tag time)	20
summary	text only - i.e. must not include HTML tags	180
link rel="image"	link to image representing the tag	255
link rel="self"	a user accessible url for the tag	255
radiotag:service	the human-readable name of the service tagged	16
radiotag:service uri	the radio service's bearer URI	32

Note the difference here between id and link rel="self". id is a globally unique identifier. link rel="self" gives the url as visible to the client/user (i.e. scoped by the auth mode).

Also note that we are interpreting the published entry as equivalent to the tag time. The updated element can be used to indicate that the tag data has been updated, e.g. the description has changed.

The 255 character limit on urls is based on a strict reading of the note in paragraph 3 of RFC 2616 Section 3.2.1.

The radiotag:service limit matches the mediumNameType in the EPG specifications[3] (and also the DAB label length[7]).

The example below shows these elements in context:

```
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
```

## DRAFT

```
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
<title>Meet David Sedaris</title>
<link href="http://radiotag.bbc.co.uk"/>
<link href="http://radiotag.bbc.co.uk" rel="self"/>
<updated>2011-08-01T11:38:38+01:00</updated>
<author>
  <name>BBC</name>
</author>
<id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
<os:totalResults>1</os:totalResults>
<os:startIndex>1</os:startIndex>
<os:itemsPerPage>1</os:itemsPerPage>
<entry>
  <title>Meet David Sedaris</title>
  <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b01211y4_150_84.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b01211y4?
t=518"/>
  <id>urn:uuid:5001c814-7a28-42a4-b35a-eef17abc7249</id>
  <updated>2011-08-01T11:38:38+01:00</updated>
  <published>2011-08-01T11:38:38+01:00</published>
  <summary>'Me Talk Pretty One Day' and 'It's Catching'.</summary>
</entry>
</feed>
```

## 7.2 Tags data

```
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
<title>Tag List</title>
<link href="http://radiotag.bbc.co.uk/tags"/>
<link href="http://radiotag.bbc.co.uk/tags" rel="self"/>
<updated>2011-08-02T17:22:09+01:00</updated>
<author>
  <name>BBC</name>
</author>
<id>urn:uuid:bf3e7d30-ccb8-4b45-b438-c790fb2ec5f7</id>
<os:totalResults>3</os:totalResults>
<os:startIndex>1</os:startIndex>
<os:itemsPerPage>3</os:itemsPerPage>
<entry>
  <title>PM</title>
  <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b012wjd3_150_84.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=1329"/>
  <id>urn:uuid:8facc0c0-ce13-4349-8664-dc71d55c6c97</id>
  <updated>2011-08-02T17:22:09+01:00</updated>
  <published>2011-08-02T17:22:09+01:00</published>
  <summary>Eddie Mair presents the day's top stories.</summary>
</entry>
```

**DRAFT**

```
<entry>
  <title>PM</title>
  <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b012wjd3_150_84.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=1328"/>
  <id>urn:uuid:9f61f2c1-f3f7-4ff7-bc61-32f0e468054d</id>
  <updated>2011-08-02T17:22:08+01:00</updated>
  <published>2011-08-02T17:22:08+01:00</published>
  <summary>Eddie Mair presents the day's top stories.</summary>
</entry>
<entry>
  <title>PM</title>
  <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://node1.bbcimg.co.uk/iplayer/images/
episode/b012wjd3_150_84.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b012wjd3?
t=1328"/>
  <id>urn:uuid:8e67aef6-4e8c-47ac-bc10-f89d4d5bac17</id>
  <updated>2011-08-02T17:22:08+01:00</updated>
  <published>2011-08-02T17:22:08+01:00</published>
  <summary>Eddie Mair presents the day's top stories.</summary>
</entry>
</feed>
```

## 8. Limits

### 8.1 Data elements

Data element	Max. size in bytes	Notes
author	16	Atom entry
id	48	Atom entry
bearerURI	32	As specified by RadioDNS
summary	180	Atom entry
title	128	Atom entry (compatible with DAB DLS/RDS RT)
url	255	See HTTP/1.1 specification[8]

## 9. Appendix

### 9.1 Narratives

#### 9.1.1 From client to user pairing

This section describes the requests and responses made between a client and a RadioTAG server when the server supports both client and user tagging.

##### *Tune radio to BBC Radio 4*

After tuning to BBC Radio 4, a RadioDNS look-up is performed to resolve the broadcast parameters into a hostname for the RadioTAG service.

Having ascertained that the service supports RadioTAG, the client makes available a Tag button.

##### *Press tag*

The user presses the Tag button.

##### *Request*

The client makes a POST request to the tag service with the bearer identifier, and a time. Unix Time is used for the time parameter.

As this client has no Auth Token, the Authorization header is blank. It could also simply not be there at all. The following sequence of events is also triggered when the request contains an invalid authentication token.

```
POST /radiodns/tag/1/tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Authorization:↵
Content-Type: application/x-www-form-urlencoded↵
↵
bearer=dab%3Ace1.ce15.c224.0&time=1319201989&time_source=broadcast
```

##### *Response*

When a tag service supports client tagging, it responds to an unauthenticated /tag request by returning a 401 Unauthorized response containing a WWW-Authenticate header that advertises the ability to upgrade authorisation mode.

This header consists of three parts: name, a plain text string suitable for display in a client UI to indicate who or what the client is attempting to authorise with or for, a URI which provides the base or prefix to all CPA endpoint URLs, and mode which is used to inform the client which upgradable modes of the CPA specification[5] are supported by this auth provider.

```
HTTP/1.1 401 Unauthorized↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="client"↵
Content-Type: text/plain; charset=utf-8↵
↵
```

## DRAFT

Must request client token

### **Request**

As per the CPA specification[5], the client begins a client mode registration.

Because it has no client identifiers, it must first request these.

```
POST /register HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-Type: application/json↵
↵
{
  "client_name": "Revo Axis",
  "software_id": "ir-svn",
  "software_version": "1.0.0#100443"
}
```

### **Response**

The client credentials are returned in a JSON response.

```
HTTP/1.1 201 Created↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-Type: application/json↵
↵
{
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs"
}
```

### **Request**

Now that an identifier and secret have been obtained for the client, these can be exchanged for a client token.

```
POST /token HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-type: application/x-www-form-urlencoded↵
↵
{
  "grant_type": "http://tech.ebu.ch/cpa/1.0/client_credentials",
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs",
  "domain": "radiotag.bbc.co.uk"
}
```

### **Response**

The token is returned in a JSON response object.

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-type: application/json↵
Cache-control: no-store↵
Pragma: no-cache↵
```



## DRAFT

```
←  
{  
  "access_token": "28b8caec68ae4a8c89dffaa37d131295",  
  "token_type": "bearer",  
  "domain_name": "BBC"  
}
```

### Request

Now the client has successfully exchanged its credentials for a token, the tag request can be made again, this time passing the token in a header of a POST request to /tag.

```
POST /radiodns/tag/1/tag HTTP/1.1←  
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295←  
Content-Type: application/x-www-form-urlencoded←  
Host: radiotag.bbc.co.uk←  
←  
bearer=dab%3Ace1.ce15.c224.0&time=1319201989&time_source=broadcast
```

### Response

The server verifies the request by checking the token against its nominated authorisation provider, and if valid creates a tag. The metadata corresponding to this tag is returned in the body of a 201 Created response, in the form of an Atom document. See Data formats for more details.

An example entry for a tag created during an episode of a BBC Radio 4 programme is shown below:

```
HTTP/1.1 201 Created←  
Date: Fri, 21 Oct 2011 12:59:49 GMT←  
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://  
ap.bbc.co.uk" modes="user"←  
Content-Type: application/atom+xml; charset=utf-8←  
←  
<?xml version="1.0"?>  
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://  
radiodns.org/2011/radiotag"  
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">  
  <title>Feedback</title>  
  <link href="http://radiotag.bbc.co.uk"/>  
  <link href="http://radiotag.bbc.co.uk" rel="self"/>  
  <updated>2011-10-21T13:59:49+01:00</updated>  
  <author>  
    <name>BBC</name>  
  </author>  
  <id>urn:uuid:661417da-cb8d-4fd0-a8fd-9b55ed2086d7</id>  
  <os:totalResults>1</os:totalResults>  
  <os:startIndex>1</os:startIndex>  
  <os:itemsPerPage>1</os:itemsPerPage>  
  <entry>  
    <title>Feedback</title>  
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</  
radiotag:service>  
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/  
b015zsz2.jpg"/>  
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zsz2?  
t=1789"/>
```

## DRAFT

```
<id>urn:uuid:661417da-cb8d-4fd0-a8fd-9b55ed2086d7</id>
<updated>2011-10-21T13:59:49+01:00</updated>
<published>2011-10-21T13:59:49+01:00</published>
<summary>Listener views on local radio cuts. Roger hears how to secure
inclusion on R4's Last Word.</summary>
</entry>
</feed>
```

### Press OK

In the previous, successful `/tag` request, the server's response contained an invitation to upgrade to user mode. The presence of this header indicates to the client that the server supports the pairing a client with a user account. At this stage the client can present to the user the option to register with the server, or to accept the information in the current tag and return to the default state for the radio service.

In this case, we chose the latter by pressing OK.

### Press Tags

As the server supports client tagging the tags created so far have been stored on the server against the token. The client can request a list of tags by making a GET request to `/tags` with the token in the header:

### Request

```
GET /radiodns/tag/1/tags HTTP/1.1↵
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵
Host: radiotag.bbc.co.uk↵
↵
```

### Response

The server responds with an Atom feed containing a list of tags created for this device.

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Tag List</title>
  <link href="http://radiotag.bbc.co.uk/tags"/>
  <link href="http://radiotag.bbc.co.uk/tags" rel="self"/>
  <updated>2011-10-21T13:59:49+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:8eca1859-bb85-4c23-ba06-d078f6bfc9f5</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>Feedback</title>
```

## DRAFT

```
<radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</radiotag:service>
<link rel="image" href="http://radiotag.bbc.co.uk/images/episode/b015zxsx2.jpg"/>
<link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zxsx2?t=1789"/>
<id>urn:uuid:661417da-cb8d-4fd0-a8fd-9b55ed2086d7</id>
<updated>2011-10-21T13:59:49+01:00</updated>
<published>2011-10-21T13:59:49+01:00</published>
<summary>Listener views on local radio cuts. Roger hears how to secure inclusion on R4's Last Word.</summary>
</entry>
</feed>
```

### Press Tag

We now show the flow of interactions when a user decides to register their client with the service. The process begins with the user pressing the Tag button as before.

### Request

```
POST /radiodns/tag/1/tag HTTP/1.1
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295
Content-Type: application/x-www-form-urlencoded
Host: radiotag.bbc.co.uk
bearer=dab%3Ace1.ce15.c224.0&time=1319201989&time_source=broadcast
Response
```

The response in this case is a 201 Created, since the service supports client tagging and the client has passed in the token with the request to /tag. Again the response contains a WWW-Authenticate advertisement. The client uses the presence of this header to decide to display the option to register.

```
HTTP/1.1 201 Created
Date: Fri, 21 Oct 2011 12:59:49 GMT
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://ap.bbc.co.uk" modes="user"
Content-Type: application/atom+xml; charset=utf-8
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Feedback</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-10-21T13:59:49+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:4b8a1b72-f76b-4dc2-9db8-cb15042454ea</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>Feedback</title>
```

## DRAFT

```
<radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</radiotag:service>
<link rel="image" href="http://radiotag.bbc.co.uk/images/episode/b015zxsx2.jpg"/>
<link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zxsx2?t=1789"/>
<id>urn:uuid:4b8a1b72-f76b-4dc2-9db8-cb15042454ea</id>
<updated>2011-10-21T13:59:49+01:00</updated>
<published>2011-10-21T13:59:49+01:00</published>
<summary>Listener views on local radio cuts. Roger hears how to secure inclusion on R4's Last Word.</summary>
</entry>
</feed>
```

### **Press Register**

This time the user selects the option to register on the client. The client needs to follow the steps for obtaining a user access token detailed in the CPA specification[5], namely repeating the token step but this time with a request for a device code.

### **Request**

```
POST /associate HTTP/1.1↵
Content-Type: application/json↵
Host: ap.bbc.co.uk↵
↵
{
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs",
  "domain": "radiotag.bbc.co.uk"
}
```

### **Response**

The service responds with a user code and the location of a web site where the user can complete the registration process.

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-Type: application/json↵
Cache-Control: no-store↵
Pragma: no-cache↵
↵
{
  "device_code": "197bf88c-749a-42e2-93f0-e206bac2252f",
  "user_code": "AbfZDgJr",
  "verification_uri": "https://ap.example.com/verify",
  "interval": 5,
  "expires_in": 1800
}
```

### **Register with a web front end**

Registering with a web front end and obtaining the authenticating PIN number is outside the scope of both RadioTAG and the CPA specification[5], so is not specified here.

## DRAFT

The following is a sketch of how this part of the system might be implemented:

A typical scenario would be that the user visits the AP's web front end, authenticates by some means with the provider (using their user name and password, for example), and submits the user code obtained in the previous step using a form.

This causes a request to be made to the service which has previously stored the user code that was issued to the client in the previous step. The service then checks the authenticity and, if valid, links the associated client id with user.

### ***Polling to validate registration***

Whilst the user is completing the web front end steps, the client can begin to poll the AP to see if they have been paired. The request should be repeated, at most, as frequently as the interval duration advises.

Polling should be attempted for no longer than the `expires_in` duration.

### ***Request***

```
POST /token HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-type: application/json↵
↵
{
  "grant_type": "http://tech.ebu.ch/cpa/1.0/device_code",
  "device_code": "197bf88c-749a-42e2-93f0-e206bac2252f",
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs",
  "domain": "radiotag.bbc.co.uk"
}
```

### ***Intermediate Response***

Whilst the user is still in the process of registering with the web front end, the response to this request will indicate a pending state.

The same request is repeated each time.

```
HTTP/1.1 202 Accepted↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-type: application/json↵
Cache-control: no-store↵
Pragma: no-cache↵
↵
{
  "reason": "authorization_pending"
}
```

### ***Final Response***

When the user has completed the necessary steps on the web front end and the client is now paired with a user, the AP can return an appropriate response to the client, which includes a new token that should replace the previously stored token.

```
HTTP/1.1 200 OK↵
```

## DRAFT

```
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-type: application/json↵
Cache-control: no-store↵
Pragma: no-cache↵
↵
{
  "user_name": "Alice",
  "access_token": "28b8caec68ae4a8c89dffaa37d131295",
  "token_type": "bearer",
  "domain_name": "BBC"
}
```

### Press Tag

The client now has a token which identifies the client with the user account on the server. Subsequent tag requests are made as POSTs to /tag with this token sent in the request headers, so that they can be stored against the user's account.

### Request

```
POST /radiodns/tag/1/tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵
Content-Type: application/x-www-form-urlencoded↵
↵
bearer=dab%3Ace1.ce15.c224.0&time=1319201990&time_source=broadcast
```

### Response

```
HTTP/1.1 201 Created↵
Date: Fri, 21 Oct 2011 12:59:50 GMT↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Feedback</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-10-21T13:59:50+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:5967db0e-dc63-428d-a075-90cf316ded5d</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>Feedback</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zsx2.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zsx2?
t=1790"/>
```

## DRAFT

```
<id>urn:uuid:5967db0e-dc63-428d-a075-90cf316ded5d</id>
<updated>2011-10-21T13:59:50+01:00</updated>
<published>2011-10-21T13:59:50+01:00</published>
<summary>Listener views on local radio cuts. Roger hears how to secure
inclusion on R4's Last Word.</summary>
</entry>
</feed>
```

### Press Tags

The client can again request a list of tags, this time using the new authentication token. The server has migrated the tags created while the client was unpaired to the user's account, so all three tags created above are returned in the Atom feed.

### Request

```
GET /radiodns/tag/1/tags HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵
↵
```

### Response

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 12:59:50 GMT↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>Tag List</title>
  <link href="http://radiotag.bbc.co.uk/tags"/>
  <link href="http://radiotag.bbc.co.uk/tags" rel="self"/>
  <updated>2011-10-21T13:59:50+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:93beb9c2-0b8d-49ad-a813-c1e6120f63b9</id>
  <os:totalResults>3</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>3</os:itemsPerPage>
  <entry>
    <title>Feedback</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zxsx2.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zxsx2?
t=1790"/>
    <id>urn:uuid:5967db0e-dc63-428d-a075-90cf316ded5d</id>
    <updated>2011-10-21T13:59:50+01:00</updated>
    <published>2011-10-21T13:59:50+01:00</published>
    <summary>Listener views on local radio cuts. Roger hears how to secure
inclusion on R4's Last Word.</summary>
  </entry>
</entry>
```

## DRAFT

```
<title>Feedback</title>
<radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zxsx2.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zxsx2?
t=1789"/>
  <id>urn:uuid:4b8a1b72-f76b-4dc2-9db8-cb15042454ea</id>
  <updated>2011-10-21T13:59:49+01:00</updated>
  <published>2011-10-21T13:59:49+01:00</published>
  <summary>Listener views on local radio cuts. Roger hears how to secure
inclusion on R4's Last Word.</summary>
</entry>
<entry>
  <title>Feedback</title>
  <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
  <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zxsx2.jpg"/>
  <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zxsx2?
t=1789"/>
  <id>urn:uuid:661417da-cb8d-4fd0-a8fd-9b55ed2086d7</id>
  <updated>2011-10-21T13:59:49+01:00</updated>
  <published>2011-10-21T13:59:49+01:00</published>
  <summary>Listener views on local radio cuts. Roger hears how to secure
inclusion on R4's Last Word.</summary>
</entry>
</feed>
```

### 9.1.2 Unidentified to user identity

This section shows the HTTP traces of transactions between a RadioTAG client (e.g. a radio) and a RadioTAG service. It covers the scenario where the RadioTAG service permits unidentified tagging upgradeable to user account tagging, i.e. it provides a response to an unauthorized client but does not store tags until the client has been paired with a user account.

Here we deal only with the differences between this scenario and the client-to-user scenario above. Please refer to that document for more information.

#### **Press Tag**

The user presses the Tag button. Note that the request is exactly the same as in the client case above.

#### **Request**

```
POST /radiodns/tag/1/tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Content-Type: application/x-www-form-urlencoded↵
↵
bearer=dab%3Ace1.ce15.c224.0&time=1319201989&time_source=broadcast
```

#### **Response**

The response is a 200 OK rather than a 201 Created. The client should remember this result for later as it indicates that the client should resubmit the tag request after registration.



## DRAFT

Note that just like the client case, the response contains a WWW-Authenticate header. The client can use this to provide the choice to accept the result or upgrade from client to user mode.

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 13:00:59 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"↵
Content-Type: application/atom+xml; charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>The Archers</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-10-21T14:00:59+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:3bfaa9dd-11ed-45f9-8f3c-6587db086b04</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>The Archers</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zs13.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zs13?
t=59"/>
    <id>urn:uuid:3bfaa9dd-11ed-45f9-8f3c-6587db086b04</id>
    <updated>2011-10-21T14:00:59+01:00</updated>
    <published>2011-10-21T14:00:59+01:00</published>
    <summary>David brings shocking news.</summary>
  </entry>
</feed>
```

### **Press OK**

At this point, the client can forget the stored 200 OK result code.

### **Press Tags**

#### **Request**

```
GET /radiodns/tag/1/tags HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
↵
```

#### **Response**

As this service does not provide client tagging, there are no tags stored on the server.

## DRAFT

```
HTTP/1.1 401 Unauthorized↵
Date: Fri, 21 Oct 2011 13:00:59 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"↵
Content-Type: text/plain;charset=utf-8↵
↵
Must request user token
```

### **Press Tag**

#### **Request**

```
POST /radiodns/tag/1/tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Content-Type: application/x-www-form-urlencoded↵
↵
bearer=dab%3Ace1.ce15.c224.0&time=1319202059&time_source=broadcast
```

#### **Response**

Again, the client should remember that the return code for this /tag request is 200.

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 13:01:00 GMT↵
WWW-Authenticate: CPA version="1.0" name="BBC AP" uri="https://
ap.bbc.co.uk" modes="user"↵
Content-Type: application/atom+xml;charset=utf-8↵
↵
<?xml version="1.0"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://
radiodns.org/2011/radiotag"
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
  <title>The Archers</title>
  <link href="http://radiotag.bbc.co.uk"/>
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
  <updated>2011-10-21T14:00:59+01:00</updated>
  <author>
    <name>BBC</name>
  </author>
  <id>urn:uuid:8ea43558-70c2-4a4a-aeb9-8ffeee700898</id>
  <os:totalResults>1</os:totalResults>
  <os:startIndex>1</os:startIndex>
  <os:itemsPerPage>1</os:itemsPerPage>
  <entry>
    <title>The Archers</title>
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
radiotag:service>
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
b015zs13.jpg"/>
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zs13?
t=59"/>
    <id>urn:uuid:8ea43558-70c2-4a4a-aeb9-8ffeee700898</id>
    <updated>2011-10-21T14:00:59+01:00</updated>
    <published>2011-10-21T14:00:59+01:00</published>
    <summary>David brings shocking news.</summary>
  </entry>
```

</feed>

### **Press Register**

#### **Request**

As per the CPA specification[5], the client begins a client mode registration.

Because it has no client identifiers, it must first request these.

```
POST /register HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-Type: application/json↵
↵
{
  "client_name": "Revo Axis",
  "software_id": "ir-svn",
  "software_version": "1.0.0#100443"
}
```

#### **Response**

The client credentials are returned in a JSON response.

```
HTTP/1.1 201 Created↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-Type: application/json↵
↵
{
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs"
}
```

#### **Request**

Now that an identifier and secret have been obtained for the client, the client can begin user mode association.

```
POST /associate HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-type: application/json↵
↵
{
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs",
  "scope": "radiotag.bbc.co.uk"
}
```

#### **Response**

```
HTTP/1.1 200 OK↵
Content-type: application/json↵
Cache-Control: no-store↵
Pragma: no-cache↵
↵
{
```

## DRAFT

```
"user_code": "Abf13",
"verification_uri": "https://www.bbc.co.uk/id/verify"
"expires_in": 1800
"interval": 5
}
```

### **Register with the web front end**

Registering with a web front end is outside the scope of the RadioTAG specification. See the previous note on registering with a web front end above for one possible implementation.

### **Poll for token**

Whilst the user is dealing with the web front end, the client should poll for a token at the rate indicated in the `interval` which will return in the event of the user completing the necessary steps on the web front end.

### **Request**

```
POST /token HTTP/1.1↵
Host: ap.bbc.co.uk↵
Content-type: application/json↵
↵
{
  "grant_type": "http://tech.ebu.ch/cpa/1.0/device_code",
  "device_code": "197bf88c-749a-42e2-93f0-e206bac2252f",
  "client_id": "1234",
  "client_secret": "sdalfqealskdfnk13984r2n23klndvs",
  "domain": "radiotag.bbc.co.uk"
}
```

### **Response**

```
HTTP/1.1 200 OK↵
Date: Fri, 21 Oct 2011 12:59:49 GMT↵
Content-type: application/json↵
Cache-control: no-store↵
Pragma: no-cache↵
↵
{
  "user_name": "Alice",
  "token": "28b8caec68ae4a8c89dffaa37d131295",
  "token_type": "bearer",
  "domain_name": "BBC"
}
```

### **Request**

The client should have stored the result of the previous request to `/tag`. As it was a 200 OK rather than 201 Created, the client knows it should resubmit the tag request, this time including the newly acquired Authorization header token value:

```
POST /tag HTTP/1.1↵
Host: radiotag.bbc.co.uk↵
Authorization: Bearer 28b8caec68ae4a8c89dffaa37d131295↵
Content-Type: application/x-www-form-urlencoded↵
```

## DRAFT

↵

bearer=dab%3Ace1.ce15.c224.0&time=1319202060&time\_source=broadcast

### Response

This time the response status is 201 Created to indicate that the tag data has been stored on the server and can be retrieved both on the device and via the web.

HTTP/1.1 201 Created↵

Date: Fri, 21 Oct 2011 13:01:00 GMT↵

Content-Type: application/atom+xml; charset=utf-8↵

↵

```
<?xml version="1.0"?>
```

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:radiotag="http://radiodns.org/2011/radiotag"
```

```
  xmlns:os="http://a9.com/-/spec/opensearch/1.1/">
```

```
  <title>The Archers</title>
```

```
  <link href="http://radiotag.bbc.co.uk"/>
```

```
  <link href="http://radiotag.bbc.co.uk" rel="self"/>
```

```
  <updated>2011-10-21T14:01:00+01:00</updated>
```

```
  <author>
```

```
    <name>BBC</name>
```

```
  </author>
```

```
  <id>urn:uuid:fcbb6008-aa54-45b4-91c9-78ec0655f9d7</id>
```

```
  <os:totalResults>1</os:totalResults>
```

```
  <os:startIndex>1</os:startIndex>
```

```
  <os:itemsPerPage>1</os:itemsPerPage>
```

```
  <entry>
```

```
    <title>The Archers</title>
```

```
    <radiotag:service uri="dab:ce1.ce15.c224.0">BBC Radio 4</
```

```
radiotag:service>
```

```
    <link rel="image" href="http://radiotag.bbc.co.uk/images/episode/
```

```
b015zs13.jpg"/>
```

```
    <link rel="canonical" href="http://www.bbc.co.uk/programmes/b015zs13?
```

```
t=60"/>
```

```
    <id>urn:uuid:fcbb6008-aa54-45b4-91c9-78ec0655f9d7</id>
```

```
    <updated>2011-10-21T14:01:00+01:00</updated>
```

```
    <published>2011-10-21T14:01:00+01:00</published>
```

```
    <summary>David brings shocking news.</summary>
```

```
  </entry>
```

```
</feed>
```

# History

Document history		
First Release	June, 2015	Initial release