

---

# CBIdentity Class Reference

Networking, Internet, & Web: Services & Discovery



2007-05-14



Apple Inc.  
© 2007 Apple Inc.  
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.  
1 Infinite Loop  
Cupertino, CA 95014  
408-996-1010

Apple, the Apple logo, Cocoa, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

**Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.**

**IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY**

**DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.**

**THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.**

**Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.**

# Contents

---

## **CBIdentity Class Reference 5**

---

Overview	5
Tasks	5
Finding Identities	5
Getting Identity Attributes	6
Storing Identities	6
Class Methods	6
identityWithCSIdentity:	6
identityWithName:authority:	7
identityWithPersistentReference:	7
identityWithUUIDString:authority:	8
Instance Methods	8
aliases	8
authority	9
CSIdentity	9
emailAddress	9
fullName	9
image	10
isHidden	10
isMemberOfGroup:	10
persistentReference	11
posixName	11
UUIDString	12

---

## **Document Revision History 13**

---



# CBIdentity Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSCoding NSCopying NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/Collaboration.framework
<b>Availability</b>	Available in Mac OS X v10.5 and later.
<b>Companion guide</b>	Identity Services Programming Guide
<b>Declared in</b>	CBIdentity.h

## Overview

A `CBIdentity` object is used for accessing the attributes of an identity stored in an identity authority. You can use an identity object for finding identities, and storing them in an access control list (ACL). If you need to edit these attributes, take advantage of the `CSIdentity` class in Core Services.

You can obtain a `CBIdentity` object from one of the following class factory methods:

`identityWithName:authority:`, `identityWithUUIDString:authority:`,  
`identityWithPersistentReference:`, or `identityWithCSIdentity:`.

A `CBIdentity` object has methods to support for interoperability with the Core Services Identity API. Send `CSIdentity` to your `CBIdentity` object to return an opaque object for use in the Core Services Identity API. Similarly, call `identityWithCSIdentity:` to use an Core Services Identity opaque object in the Collaboration framework.

There are two subclasses of `CBIdentity`: `CBGroupIdentity` and `CBUserIdentity`. If you are working specifically with a group identity, use `CBGroupIdentity`. Similarly, if you are working with a user identity, use `CBUserIdentity`.

## Tasks

### Finding Identities

+ [identityWithCSIdentity:](#) (page 6)

Returns an identity object created from the specified Core Services Identity opaque object.

- + [identityWithName:authority:](#) (page 7)  
Returns the identity object with the given name from the specified identity authority.
- + [identityWithPersistentReference:](#) (page 7)  
Returns the identity object matching the persistent reference data.
- + [identityWithUUIDString:authority:](#) (page 8)  
Returns the identity object with the given UUID from the specified identity authority.

## Getting Identity Attributes

- [aliases](#) (page 8)  
Returns an array of aliases (alternate names) for the identity.
- [authority](#) (page 9)  
Returns the identity authority where the identity is stored.
- [emailAddress](#) (page 9)  
Returns the email address of an identity.
- [fullName](#) (page 9)  
Returns the full name of the identity.
- [image](#) (page 10)  
Returns the image associated with an identity.
- [isHidden](#) (page 10)  
Returns a Boolean value indicating the state of the identity's hidden property.
- [isMemberOfGroup:](#) (page 10)  
Returns a Boolean value indicating whether the identity is a member of the specified group.
- [posixName](#) (page 11)  
Returns the POSIX name of the identity.
- [UUIDString](#) (page 12)  
Returns the UUID of the identity as a string.

## Storing Identities

- [CSIdentity](#) (page 9)  
Returns an opaque object for use with the Core Services Identity API.
- [persistentReference](#) (page 11)  
Returns a persistent reference to store a reference to an identity.

## Class Methods

### **identityWithCSIdentity:**

Returns an identity object created from the specified Core Services Identity opaque object.

```
+ (CBIdentity *)identityWithCSIdentity:(CSIdentityRef)csIdentity
```

**Parameters***csIdentity*

The Core Services Identity opaque object.

**Return Value**

The identity object for use with the Collaboration framework.

**Discussion**

This method is used for interoperability with the Core Services Identity API.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CBIdentity.h

**identityWithName:authority:**

Returns the identity object with the given name from the specified identity authority.

```
+ (CBIdentity *)identityWithName:(NSString *)name authority:(CBIdentityAuthority *)authority
```

**Parameters***name*

The name of the identity.

*authority*

The identity authority to search.

**Return Value**The identity object, or `nil` if no identity is found with the specified name.**Discussion**

The name is compared against all valid identity names, including full names, short names, email addresses, and aliases.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CBIdentity.h

**identityWithPersistentReference:**

Returns the identity object matching the persistent reference data.

```
+ (CBIdentity *)identityWithPersistentReference:(NSData *)data
```

**Parameters***data*

The persistent data object that refers to an identity.

**Return Value**The identity object matching the persistent data object, or `nil` if the identity is not found.

**Discussion**

A persistent reference is an opaque data object suitable for persistent storage.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

`CBIdentity.h`

**identityWithUUIDString:authority:**

Returns the identity object with the given UUID from the specified identity authority.

```
+ (CBIdentity *)identityWithUUIDString:(NSString *)uuid
  authority:(CBIdentityAuthority *)authority
```

**Parameters**

*uuid*

The UUID of the identity you are searching for.

*authority*

The identity authority to search.

**Return Value**

The identity object, or `nil` if no identity is found with the matching criteria.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

`CBIdentity.h`

## Instance Methods

**aliases**

Returns an array of aliases (alternate names) for the identity.

```
- (NSArray *)aliases
```

**Return Value**

An array of `NSString` objects containing the alternate names for the identity.

**Discussion**

An identity can have zero or more aliases. Like the full and short names, two identities cannot share an alias.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

`CBIdentity.h`



## authority

Returns the identity authority where the identity is stored.

```
- (CBIdentityAuthority *)authority
```

### Return Value

The identity authority where the identity is stored.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## CSIdentity

Returns an opaque object for use with the Core Services Identity API.

```
- (CSIdentityRef)CSIdentity
```

### Return Value

The opaque object for use with the Core Services Identity API.

### Discussion

This method, along with `identityWithCSIdentity:`, is used for interoperability with the Core Services Identity API.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## emailAddress

Returns the email address of an identity.

```
- (NSString *)emailAddress
```

### Return Value

The email address of an identity or `nil` if none exists.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## fullName

Returns the full name of the identity.

- (NSString \*)fullName

**Return Value**

The full name for the identity.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CBIdentity.h

## image

Returns the image associated with an identity.

- (UIImage \*)image

**Return Value**

The image associated with an identity, or `nil` if none exists.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CBIdentity.h

## isHidden

Returns a Boolean value indicating the state of the identity's hidden property.

- (BOOL)isHidden

**Return Value**

YES if the identity is hidden; NO if it is not.

**Discussion**

A hidden identity does not show up in the Identity Picker. A hidden identity refers to system identities such as `root`, `www`, and `wheel`.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

CBIdentity.h

## isMemberOfGroup:

Returns a Boolean value indicating whether the identity is a member of the specified group.

- (BOOL)isMemberOfGroup:(CBGroupIdentity \*)group

### Parameters

*group*

The group to check for membership.

### Return Value

YES if the identity is a member of the group; NO if it is not.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## persistentReference

Returns a persistent reference to store a reference to an identity.

- (NSData \*)persistentReference

### Return Value

A data object that uniquely references an identity.

### Discussion

A persistent reference data object is an object generated from an identity. Persistent data objects can be written to and read from a file, making them extremely useful for storing identities in an ACL.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## posixName

Returns the POSIX name of the identity.

- (NSString \*)posixName

### Return Value

The POSIX name of the identity.

### Discussion

The POSIX name is also referred to as the “short name” for an identity. It can only contain the characters A-Z, a-z, 0-9, -, \_, ., and @.

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

CBIdentity.h

## UUIDString

Returns the UUID of the identity as a string.

```
- (NSString *)UUIDString
```

### Return Value

The UUID string of the identity.

### Discussion

The UUID string is generated so it is unique across all identity authorities. When storing ACLs, one method is to store the UUID of each identity. However, it is recommended that you use a persistent data object instead (see `persistentReference`).

### Availability

Available in Mac OS X v10.5 and later.

### Declared In

`CBIdentity.h`

# Document Revision History

---

This table describes the changes to *CBIdentity Class Reference*.

Date	Notes
2007-05-14	New document that describes the class used to create and operate on identities.

## REVISION HISTORY

### Document Revision History