ETSI TS 103 407 V1.1.1 (2016-04)

Cross Platform Authentication for limited input hybrid consumer equipment

Reference
DTS/JTC-033

Keywords
authentication, authorization, protocol

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

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 ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

The Cross Platform Authentication (CPA) protocol defines how to securely associate an IP-connected media device (such as a hybrid radio, set top box or Smart TV) with the online account of a user of a set of web services delivered to that device.

Introduction

The present document specifies version 1.0 of the Cross Platform Authentication (CPA) protocol.

The CPA protocol is specifically designed for devices with limited input and display capabilities that are not addressed by existing standards and to cater for companies that share a back-end authorization provider for managing identities but implement services separately.

The CPA protocol defines a clear separation of responsibilities between the service provider and the authorization provider. This enables the protocol to be applied in a variety of business configurations typical of the broadcast industry. For example, the CPA protocol can be used in the following scenarios:

- the service provider and the authorization provider are both managed by the same company;
- an umbrella company manages the authorization provider (so centralizing identity management) but keeps its service providers separate from one another;
- service providers from different companies use the same central authorization provider managed by a separate company (e.g. a national federated identity service).

1 Scope

1.1 General

The protocol described in the present document is intended for devices with limited input capabilities, such as hybrid radios, IP-connected set top boxes and Smart TVs, that can communicate with web services over HTTPS.

The protocol specifies two APIs:

- the API between a client device and an authorization provider by which a client obtains a bearer token;
- the API between a service provider and an authorization provider by which a service provider verifies an access token.

The present document gives an overview of the protocol (clause 4), covering the core concepts (clause 5) and roles (clause 6) used in CPA and how the device flow works (clause 7).

The CPA APIs are specified in the present document in clauses 8, Client/Authorization Provider API and clause 9, Service Provider/Authorization Provider API.

An informative annex A describes how service providers can tell clients that the option to authenticate using the CPA protocol is available, and how the bearer token obtained via CPA should be used to access protected resources. Although this clause is not normative, it is strongly recommended these conventions are followed where possible to maximize interoperability.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document:

[1] IETF RFC 2818: "HTTP Over TLS".

[2] IETF RFC 7159: "The JavaScript Object Notation (JSON) Data Interchange Format".

[3] IETF RFC 4122: "A Universally Unique Identifier (UUID) URN Namespace".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

[i.2] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

3 Definitions and abbreviations

3.1 Definitions

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

access token: bearer token used by a client as authority to access a service provider

authenticated association: association in which the client identity is associated with a user identity

authorization: granting of permission based on authenticated identification

authorization provider: service that manages client identities and the association between client identities and authenticated user identities

bearer token: security token the possession of which entitles the bearer to all rights associated with it

client: application managed entity uniquely identified by an authorization provider

client mode: authorization mode where the client relation is not associated with a user account

device: IP-connected hardware intended for user interaction

device flow: series of exchanges between the client and the authorization provider by which the client obtains a bearer token

identity provider: service that manages identity information on behalf of users and services

limited display device: device with the ability to display at least two rows of sixteen alphanumeric characters from the ISO-646 Invariant Code Set (loosely, 7-bit ASCII)

limited input device: device that has at least the ability to accept or cancel a proposed action

registration: process by which a client obtains a client identity from an authorization provider

service provider: service requiring authorization to access its protected resources

token: unique opaque string used to represent a specific granting of authorization to use a service

NOTE: See also access token and bearer token.

unauthenticated association: association in which the client identity is not associated with a user identity

user: person or agent using an application

user mode: authorization mode where the client relation is associated with a user account

3.2 Abbreviations

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

API	Application Programming Interface
CPA	Cross Platform Authentication

NOTE The subject of the present document.

HTTP	Hypertext Transfer Protocol
HTTPS	HTTP Secure
OAuth	Open Authorization
URI	Uniform Resource Identifier

4 Protocol overview

4.1 Introduction

The present document defines the protocol between a client device and an authorization provider by which a client device acquires an authorization token it may use to access protected resources and by which the service provider can identify the client device and any associated user account.

Cross Platform Association (CPA) is based on OAuth 2.0 IETF RFC 6749 [i.1] and the bearer tokens used in CPA are compatible with OAuth 2.0 bearer tokens IETF RFC 6750 [i.2]. While CPA can be used more generally, the focus of the present document is on IP-connected media devices with limited input and display. When such a device cannot run a fully capable HTTP User Agent such as a browser, the user needs access to another device which can run such software to complete the authorization.

In the typical CPA device flow described below, the application-specific steps are prefixed with (App) and the CPA-specific steps with (CPA):

- (App): An application on the device requests a protected resource from the service provider.
- (App): The service provider checks with the authorization provider to see if the device is authorized.
- (App): The authorization provider indicates that the device is not authorized.
- (App): The service provider response to the device includes the endpoint where the device may initiate the device flow and which modes of association the service provider supports.
- (CPA): The device registers itself with the authorization provider and receives a `client_id` and `client_secret` in return.
- (CPA): The device makes a request to the authorization provider to associate the device.
- (CPA): The authorization provider returns a verification URI and one-time `user_code` which the device displays to the user. It also returns a corresponding `device_code` which is not displayed.
- (CPA): The device polls the authorization provider using the `device_code` as identification.
- (App): In the meantime, the user uses a browser on another computer to open the supplied verification URI to enter the code. As part of this process, the user will be required to authenticate their user identity, typically by logging in via their identity provider.
- (CPA): Once the user has verified the association by entering their code, the authorization provider responds to the polling device by returning a bearer token.
- (App): The application running on the device can then reissue the request for the protected resource, this time including the supplied bearer token.

In the device flow outlined above, CPA specifies the protocols between a) the client device and the authorization provider and b) the service provider and the authorization provider.

The client/authorization provider API (see clause 8) consists of the following endpoints supplied by the authorization provider:

- /register to obtain a client identity consisting of a `client_id` and `client_secret;`
- /associate to initiate the association between a client and a user account;
- /token to acquire a token;
- the verification endpoint to authenticate the user's identity and verify the association.

The service provider/authorization provider API (see clause 9) consists of one endpoint supplied by the authorization provider:

- /authorized to verify a token

All other interactions between the participants in the CPA protocol are outside the scope of the present document. However, the present document includes illustrative examples to show how CPA may be integrated into an application-specific flow. These integration points include:

- how the client discovers the authorization endpoint (see clause A.2);
- how the service provider and authorization provider establish a trusted relationship (see clause 9.2);
- how a user verifies the association between their client and their user account (see clause 8.5).

4.2 Modes of association

4.2.1 Overview

CPA defines two modes of association: authenticated association ('user mode') which links a device to a user account, and unauthenticated association ('client mode'), which provides a client identity to a device without linking it to a user account.

In both cases, the client is issued a persistent client ID so that it can be identified across requests to service providers that use the same authorization provider.

Authorization providers may offer the following forms of association:

- client mode only;
- user mode only;
- client mode and user mode.

4.2.2 Authenticated association ('user mode')

If the authorization provider offers user mode, then the client ID may be linked with a user account so that requests from the device can be linked to that user. This linking of device and user is called 'authenticated association', or user mode (see clause 7.3).

Once an authenticated association has been set up, a service provider can provide a unified personalized service across a range of devices.

4.2.3 Unauthenticated association ('client mode')

The present document also defines a way to create an unauthenticated association between a device and an authorization provider, known as *client mode* (see clause 7.4), and a means of migrating from this unauthenticated association to an authenticated association.

The purpose of unauthenticated association is to support applications that require service-side persistence but do not require association with an online user account.

By using the `client_id` as an identifier, a service provider can identify a client from session to session and hence provide personalization limited to a specific device.

5 Core concepts

5.1 Bearer token

The CPA device flow is built around the concept of a *bearer token*, possession of which grants access to protected services IETF RFC 6750 [i.2].

The core function of the CPA protocol is to securely get this bearer token onto the device.

A *token* is a unique string that is intended to be unforgeable. The token is a shared secret between the client, the service provider and the authorization provider.

A *bearer* token is one for which simple possession confers the corresponding authority to its bearer.

In CPA, the token also serves as the key for looking up the association between a client and a user account. It is not a permanent identifier as it may expire and be replaced with a new token.

5.2 Domain

Tokens are valid only within the scope of the hostname of the service provider for which they are granted. This is to prevent service providers enabling the issuing of tokens for domains for which they are not responsible.

Tokens are valid for all service provider endpoints in that domain. For example, a token granted for the domain ***api.example.com*** is valid for the endpoints ***https://api.example.com/tag*** and ***https://api.example.com/epg***.

The domain is specified by the client when it requests the token. The authorization provider decides whether the client is issued a token for the requested domain.

6 Roles

6.1 Client

The client represents the user of a service on a device.

When a client registers with an authorization provider, it is given a unique `client_id` and a `client_secret`.

The `client_id` is known to the client, the service provider and the authorization provider. This client identity is shared between all service providers that use the same authorization provider.

A `client_id` shall be unique within an authorization provider and shall be valid only for that authorization provider. Different authorization providers shall not share client identities.

The `client_secret` shall be known only to the client and the authorization provider. All requests from the client to the authorization provider shall include the `client_id` and `client_secret.` The `client_secret` shall be used only in requests from the client to the authorization provider.

6.2 Service provider

The service provider provides one or more web services under the same domain.

Each bearer token is valid only for the service provider domain for which it was issued.

The exact details of how a service provider indicates to a client which authorization provider to use and which protected resources are accessible with a CPA bearer token are application-specific and not defined by the present document.

One way is for the service provider to return a `WWW-Authenticate` header in response to a request for a protected resource. This header tells the client the URI of the authorization provider and which association modes are available. For a fuller description of this approach, see clause A.2.

6.3 Authorization provider

An authorization provider provides client identities to one or more clients and maintains the link between the client identity and the user identity.

An authorization provider may answer authorization requests for one or more service providers.

Service providers may be grouped within an authorization provider to share an authorization domain. This enables the automatic provisioning of tokens for service providers within the same group once a client has been verified for one service provider within that group.