

Privileges Interface

Access player privileges including online and crossplay as well as voice and text chat.



⚠ Learn to use this **Beta** feature, but use caution when shipping with it.

The **Online Services Privileges Interface** manages a player's online privileges. Privileges refer to a player's ability to:

- Play online.
- Participate in crossplay.
- Communicate with other players.
- Use community generated content.

API Overview

Functions

The following table provides a high-level overview of the function provided by the Privileges Interface:

Function	Description
<code>QueryUserPrivilege</code>	Query a given privilege for the provided user.

Enumerated Classes

The privileges interface communicates user privileges and user privilege status' through two enumerated classes., while privileges status is given by `EPrivilegeResults`.

EUserPrivileges

A user privilege is represented by a `EUserPrivileges` enumerated class:

Value	Description
<code>CanPlay</code>	Specifies whether the user can play the game — online or offline.
<code>CanPlayOnline</code>	Specifies whether the user can play in online modes.
<code>CanCommunicateViaTextOnline</code>	Specifies whether the user can use text chat.
<code>CanCommunicateViaVoiceOnline</code>	Specifies whether the user can use voice chat.
<code>CanUseUserGeneratedContent</code>	Specifies whether the user can use content generated by other users.
<code>CanCrossPlay</code>	Specifies whether the user can ever participate in crossplay.

EPrivilegeResults

User privileges status is given by one or more `EPrivilegeResults`:

Value	Description
<code>NoFailures</code>	User has the requested privilege.
<code>RequiredPatchAvailable</code>	Patch required before the user can use privilege.
<code>RequiredSystemUpdate</code>	System update required before the user can use privilege.
<code>AgeRestrictionFailure</code>	Privilege restricted due to parental control failure.
<code>AccountTypeFailure</code>	Privilege requires a premium account.
<code>UserNotFound</code>	Invalid user.
<code>UserNotLoggedIn</code>	User must be logged in to use privilege.
<code>ChatRestriction</code>	User restricted from chat.
<code>UGCRestriction</code>	User restricted from User Generated Content (UGC).
<code>GenericFailure</code>	Platform failed for an unknown reason and handles its own dialogues.
<code>OnlinePlayRestricted</code>	Online play is restricted.
<code>NetworkConnectionUnavailable</code>	Check failed because the network is unavailable.

User Privilege

Call `QueryUserPrivilege` to determine whether a user is permitted to use a particular online feature or privilege in your game. `QueryUserPrivilege` requires that you provide the local account ID of the user and the privilege in question as parameters. `QueryUserPrivilege` returns a result that indicates whether the user you specified is permitted the privilege you

specified. If the user is not permitted to use the queried privilege, the result contains the reasons why.

It is possible there are multiple reasons why a user is not permitted to use a privilege. Consequently, the structure of the result is a cumulative bitwise OR of `EPrivilegeResults` values that represent the reasons why this user is not permitted to use the provided privilege. If the user is permitted to use this privilege, then the query returns `EPrivilegeResults::NoFailures`, which indicates there is nothing preventing the user from using the desired privilege.




Certain privileges can change when the game is suspended. It is important to re-query user privileges each time the game returns to an active state as certain game options could no longer be available.

More Information

Header File

Consult the `Privileges.h` header file directly for more information as needed. The Privileges Interface header file `Privileges.h` is located in the directory:

```
Engine\Plugins\Online\OnlineServices\Source\OnlineServicesInterface\Public\Onlin
```

 Copy full snippet

For instructions on how to obtain the UE source code, see our documentation on [Downloading Unreal Engine Source Code](#).

Function Parameters and Return Types

See the [Functions](#) section of the [Online Services Overview](#) page for an explanation of function parameters and return types, including how to pass parameters and how to process the results when functions return.