

Developer  
/ Documentation  
/ Unreal Engine ▾  
/ Unreal Engine 5.4 Documentation  
/ Programming and Scripting  
/ Online Subsystems and Services  
/ Online Subsystem  
/ Online Subsystem Presence Interface

# Online Subsystem Presence Interface

Overview of the presence interface.



When logged into an online service, users can often see information about their friends and other users they've met online, such as whether these users are online, what they're doing, if they're available to join matches, and so on. The **Presence Interface** provides the Online Subsystem with access to these features.

## Presence Status

Most online services recognize several basic presence states for each user, such as "online", "offline", and "away", as well as game-specific states like "In the lobby" or "Playing a match on (Map)". However, these settings are not always visible, or may be visible to some users but not others, due to service-specific privacy policies and account settings. The Online Subsystem does not interact with these policies or settings, but the information that it retrieves will be affected by them.

## Defining Presence

The `FOnlineUserPresence` class contains all information related to a user's presence. In addition to basic information like whether or not the user is currently online, and whether or not the user is playing a game, the user's presence (using the `FOnlineUserPresenceStatus` class) stores more in-depth information. This generally includes a localized string for display, an enumerated value (of type `EOnlinePresenceState`) to describe the user's basic state, and a set of key-value pairs to hold any game-specific custom data, which can be used when building the exact presence display message.

## Retrieving Information About Other Users

The basic flow for collecting presence information about a specific user begins with making a request to the online service through `QueryPresence`, specifying that user by `FUniqueNetId`. Once that operation finishes, the provided `FOnPresenceTaskCompleteDelegate` will be called, indicating the user and whether the request succeeded or failed. If successful, the local presence information cache will contain up-to-date presence information, which is available through the `GetCachedPresence` function.



Some online services proactively notify the Online Subsystem about user presence. On these services, although you don't actually have to call `QueryPresence` or wait for its delegate, the usual code flow should remain unchanged so that it will work across multiple services.

## Setting a User's Presence

The `SetPresence` function sets the presence status of a local user through the online service. This is an asynchronous call due to the need to verify the new status with the online service, and a delegate of type `FOnPresenceTaskCompleteDelegate` will be called upon completion. Presence status itself is described by the `FOnlineUserPresenceStatus` class.