
Unreal Engine 4 Steamworks Callback Plugin

Release 1.0

Jul 02, 2019

1	Initial Set Up	1
2	Getting Started	3
2.1	Enabling Steam Subsystem	3
2.2	Referencing Callback Object	3
2.3	Cleanup	4
2.4	Assigning Events	5
2.5	Useful References	8
3	Steam Friend Callbacks	9
3.1	GetFriendCallback	9
3.2	SteamGameOverlayActivatedCallback	10
3.3	SteamPersonaStateChangeCallback	10
3.4	ESteamPersonaChange	12
3.5	SteamSetPersonaNameCallback	12
4	Steam Matchmaking Callbacks	15
4.1	GetMatchmakingCallback	15
4.2	SteamLobbyUpdatedCallback	16
4.3	EChatMemberStateChange	17
4.4	SteamLobbyCreatedCallback	17
4.5	SteamLobbyEnteredCallback	18
4.6	ELobbyEnteredResponse	20
4.7	SteamLobbyInviteCallback	20
5	Steam User Callbacks	23
5.1	GetUserCallback	23
5.2	SteamGetAuthSessionTicketCallback	24
5.3	SteamMicroTxnAuthorizationCallback	24
5.4	LicensesUpdatedCallback	25
5.5	SteamServerConnectFailureCallback	26
5.6	SteamServersConnectedCallback	27
5.7	SteamServersDisconnectedCallback	27

CHAPTER 1

Initial Set Up

1) Unreal Engine 4 Steamworks Callback Plugin from the [UE4 Marketplace](#). The example project can be found on [GitHub](#) which requires the plugin to be installed before running.

The example project must also be run through steam to prevent crashes.

If installing from git hub create a folder in your project directory to place it in named 'Plugins'.

If installed from the UE4 Marketplace it will be located in:

C:\Program Files (x86)\UE_(VersionNumber)\Engine\Plugins\Marketplace

2.1 Enabling Steam Subsystem

Before this plugin can be used the Steam Subsystem must be enabled and active. If you are using blueprints I would highly recommend another free plugin to enable advanced steam functions [AdvancedSessions](#) along with [AdvancedSteamSessions](#). Most of the features of this plugin only work after the game is built and wrapped with Steam. I assume some features work with app id 480, but for all the features to be in full effect I advise using your own steam app id.

2.2 Referencing Callback Object

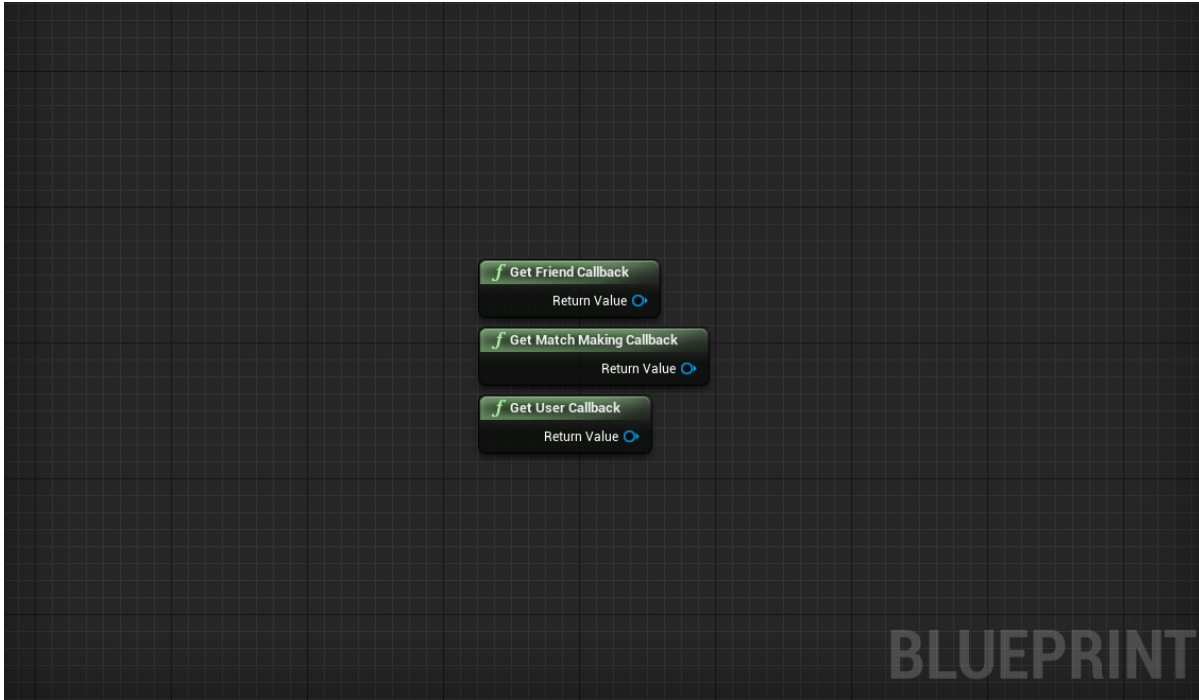
There are four callback objects which contain events that can be assigned

Friend, Matchmaking, User

They all contain unique events which can be seen later in this documentation.

To obtain a reference to one of these objects use the static function:

Get (Name of Object) Callback



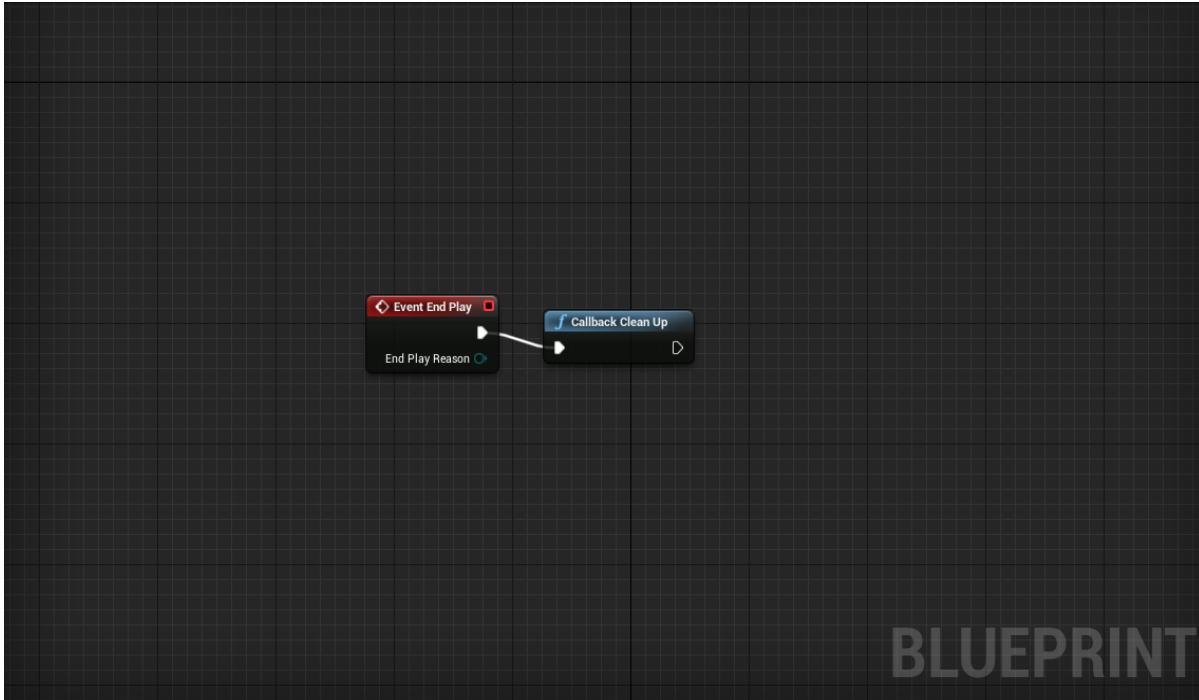
This will return a reference to the object type specified in the function.

On the first call of Get (Name of Object) Callback an Actor will be created in the scene named PSCallbackCleanUpActor.

This actor can be ignored it is just created for storage you will never need to reference it.

2.3 Cleanup

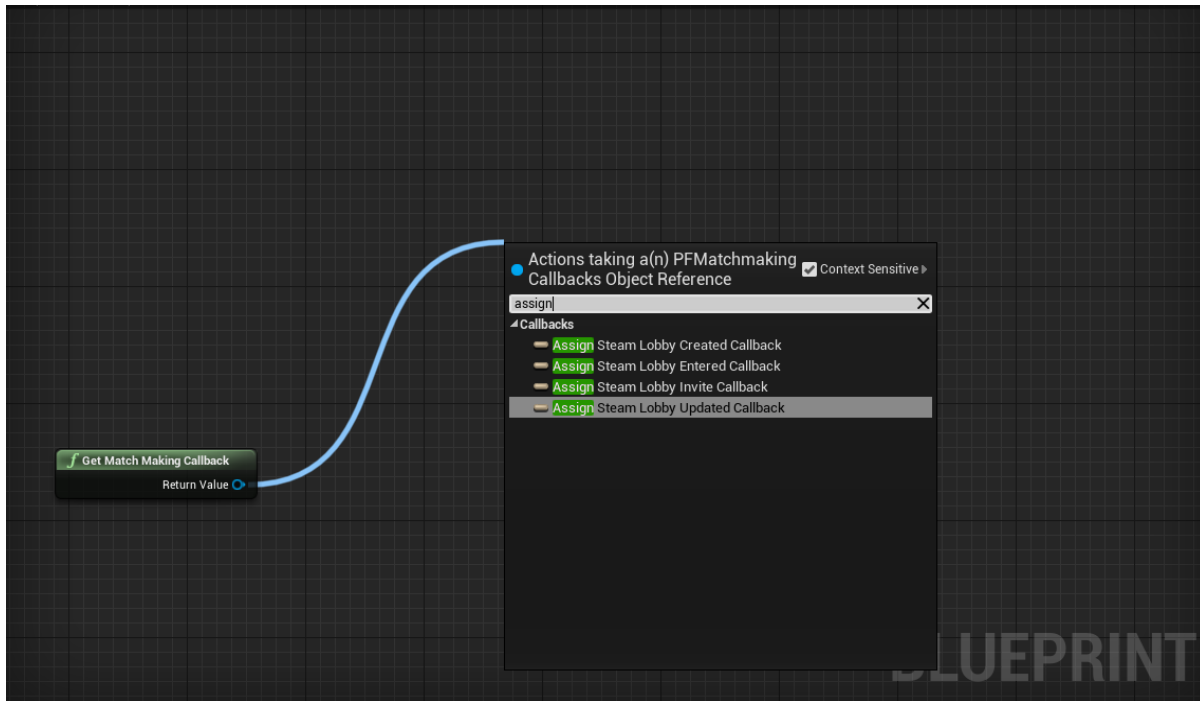
In order to prevent unwanted memory leaks, the function CallbackCleanUp should be called before the game ends



2.4 Assigning Events

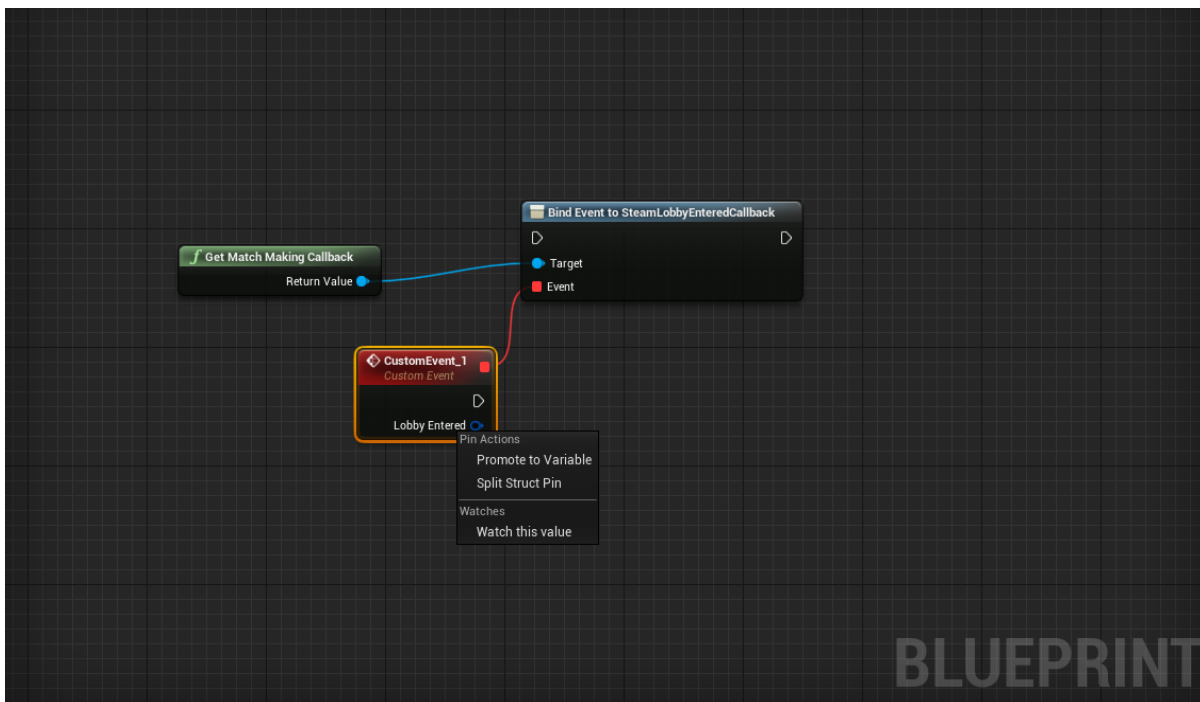
Finding Callbacks

To assign an event first drag a node from a referenced callback object out. Type “Assign” for unreal to display all available callbacks then select the callback you wish to choose. This will create a Bind Event to node.



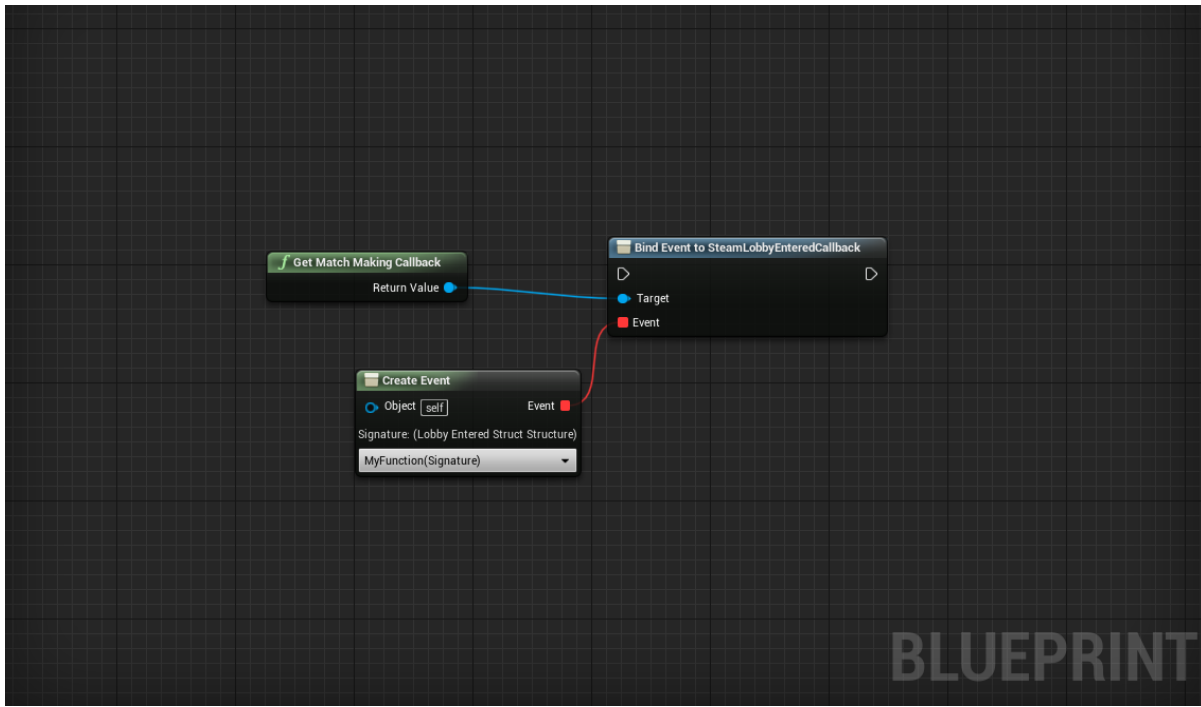
Assigning Callbacks to Custom Events

Drag the red node out from the Bind Event node and select Add Event -> Add Custom Event to assign the callback to an event. If the callback returns a struct (dark blue node) you can right click it to 'Split Struct Pin'



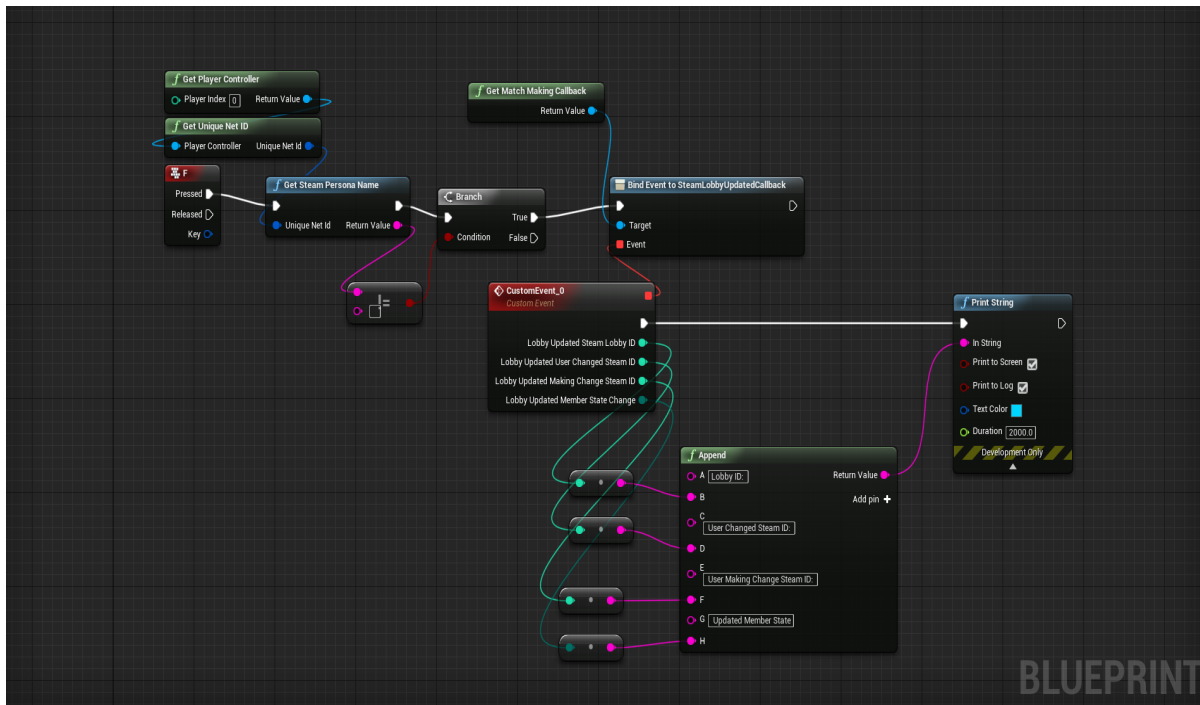
Assigning Callbacks to Functions

Drag the red node out from the Bind Event node and select Event Dispatchers->Create Event. This will create a Create Event node which you can use to find the event to a function. To do this you must create a function matching the “Signature:” on the Create Event node. Then use the drop down to select your function and bind the event.



Preventing Crashes

Before events can be assigned the **Steam Subsystem must be enabled and active or the game WILL crash**. A way to determine if Steam is active in blueprints is to use the AdvancedSessionsSteam plugin function `GetSteamPersona` if the function returns empty then do not bind events:



In C++ determining if Steam is active is as simple as checking that the Steam Subsystem is not null:

```
if (IOnlineSubsystem::Get("Steam")) {}
```

2.5 Useful References

[Steamworks Documentation](#)

[Github Source](#)

Steam Friend Callbacks

3.1 GetFriendCallback

Get reference to Friend Callback Object to assign callback events.

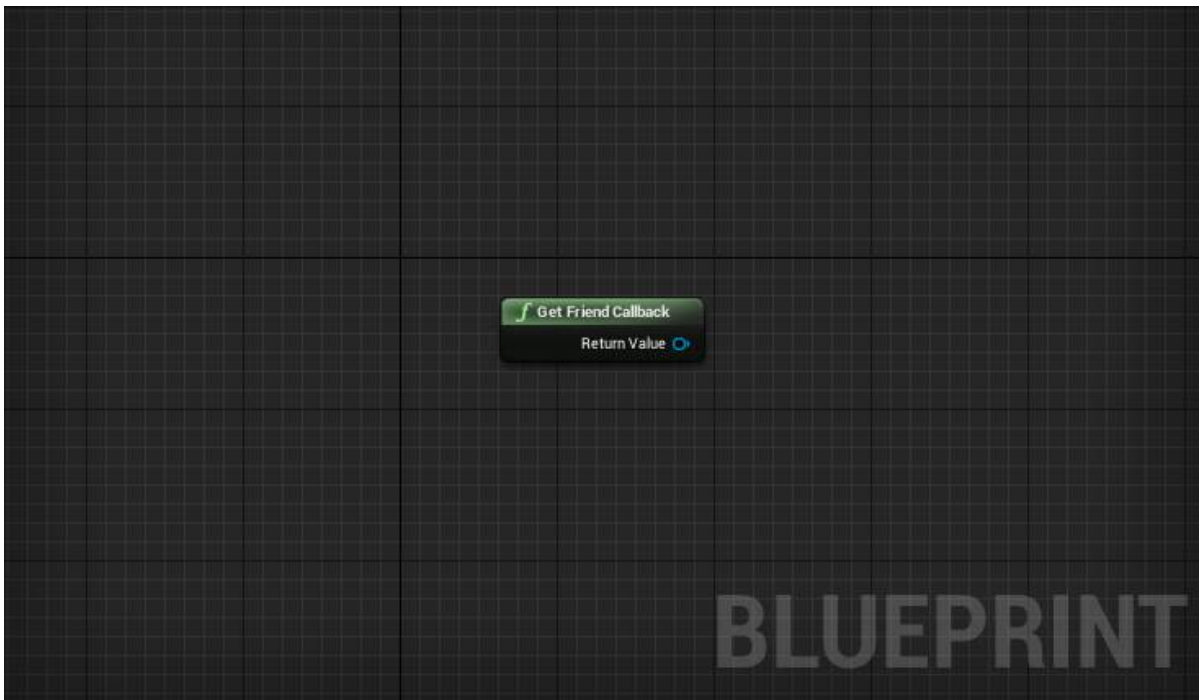


Table 1: Returns **PFFriendCallbacks**

Name	Type	Description
Return Value	PFFriendCallbacks	Pointer to friend callback handler.

3.2 SteamGameOverlayActivatedCallback

When the steam overlay activates or deactivates. Can be used to pause or resume single player games.

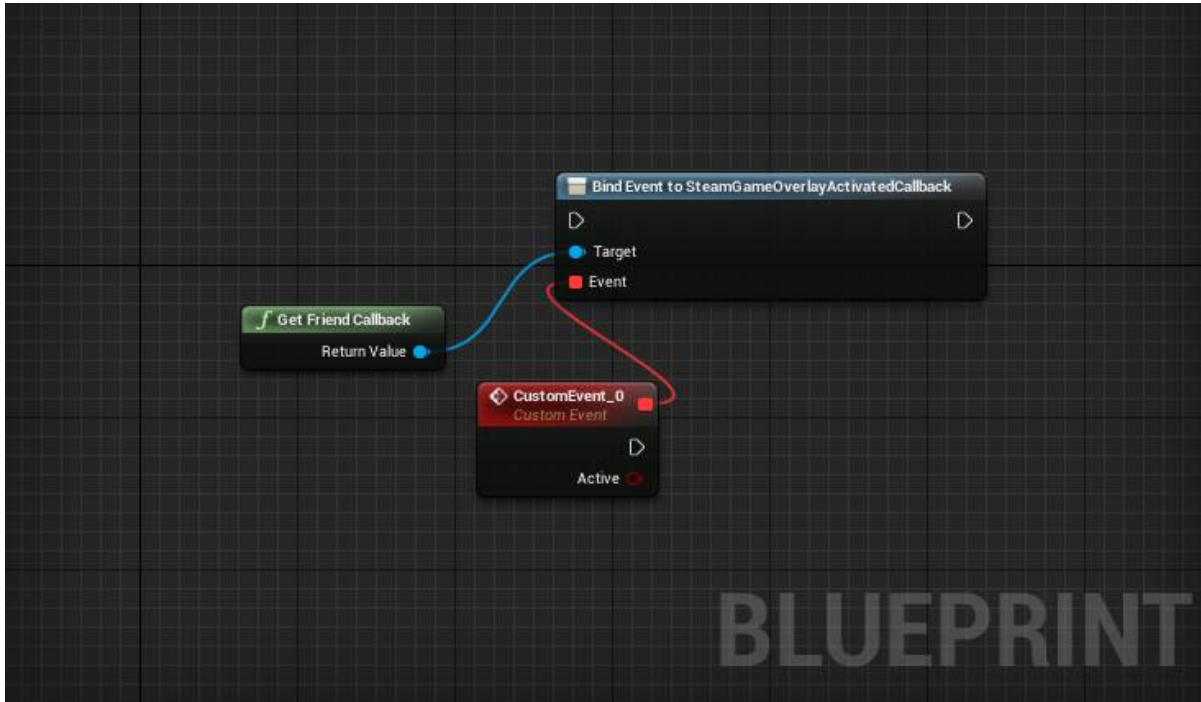
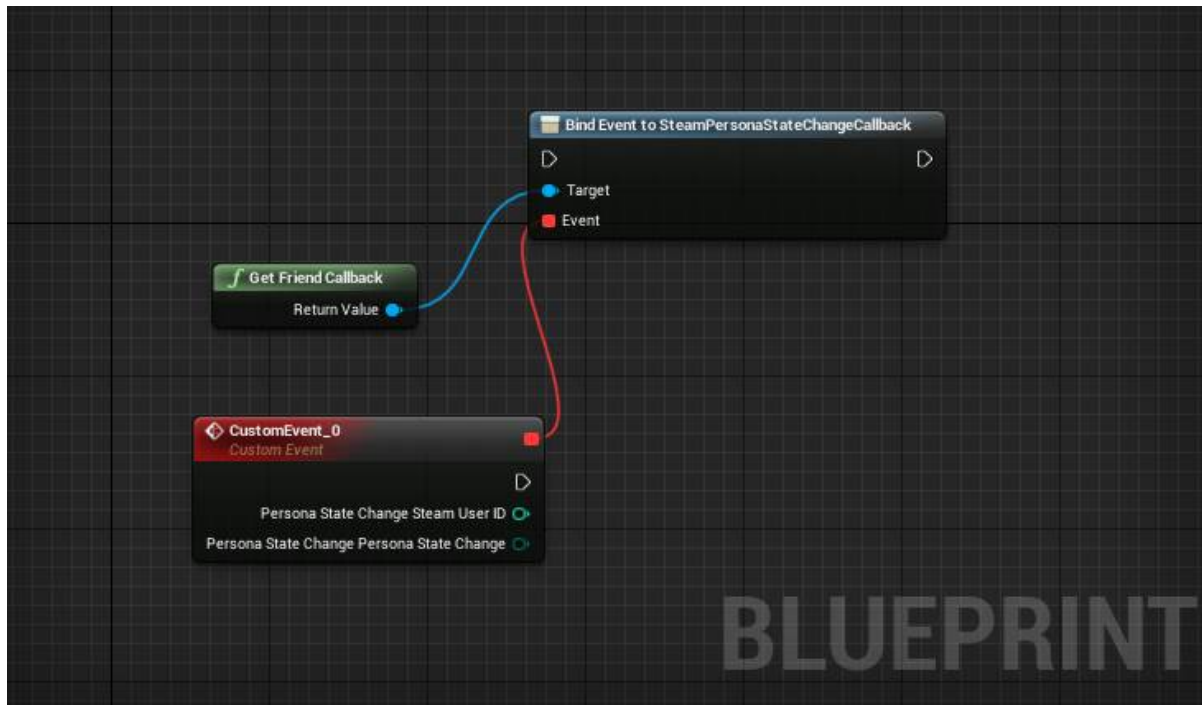


Table 2: Returns **bActive**

Name	Type	Description
bActive	bool	Returns true if steam overlay was activated, and returns false when steam overlay deactivated.

3.3 SteamPersonaStateChangeCallback

Called when a friends' status changes.

Table 3: Returns **FLobbyCreatedStruct**

Name	Type	Description
SteamUserID	Integer	Steam ID of user who changed.
SteamPersonaChange	<i>ESteamPersonaChange</i>	Steam persona change result.

3.4 ESteamPersonaChange

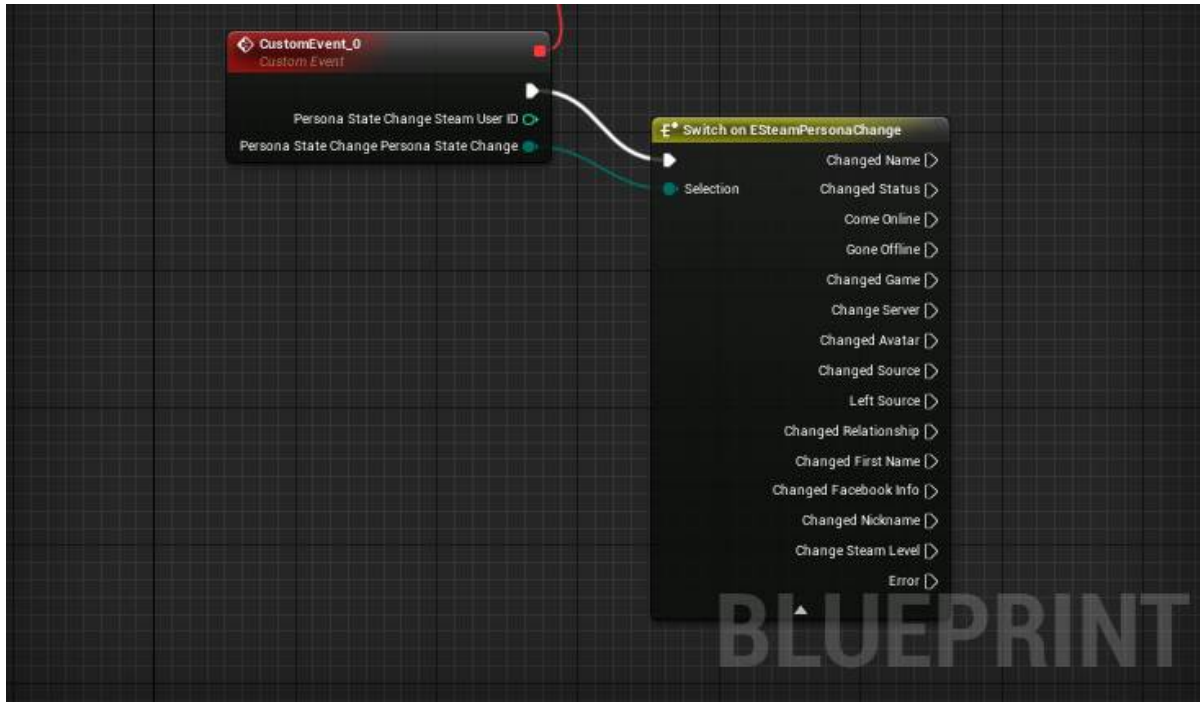
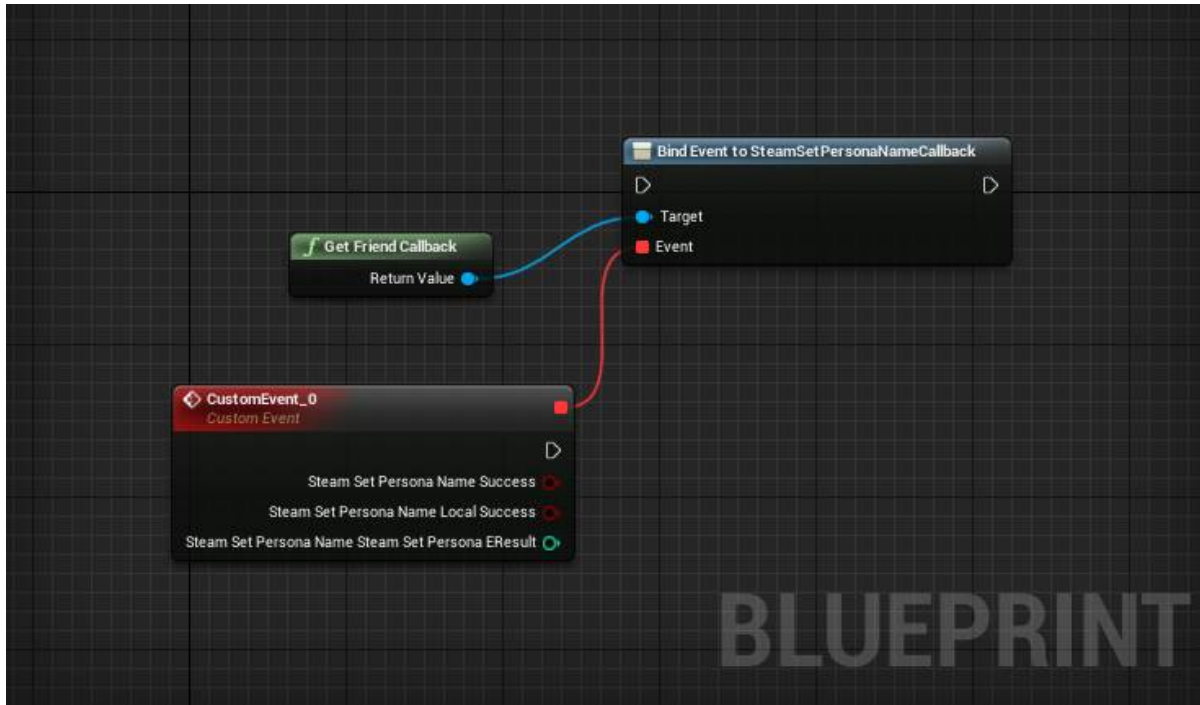


Table 4: EChatMemberStateChange

Name	Value	Description
EPersonaChangeName	ChangedName (1)	The user changed their persona name.
EPersonaChangeStatus	ChangedStatus (2)	The user changed their account status (online, away, busy).
EPersonaChangeComeOnline	ComeOnline (3)	The user has come online.
EPersonaChangeGoneOffline	GoneOffline (4)	The user has gone offline.
EPersonaChangeGamePlayed	ChangedGame (5)	The user has changed games.
EPersonaChangeGameServer	ChangeServer (6)	The user has changed game servers.
EPersonaChangeAvatar	ChangedAvatar (7)	The user has changed their steam avatar.
EPersonaChangeJoinedSource	ChangedSource (8)	The user has changed source.
EPersonaChangeLeftSource	LeftSource (9)	The user has left source.
EPersonaChangeRelationship	ChangedRelationship (10)	The user has changed their relationship.
EPersonaChangeNameFirstSet	ChangedFirstName (11)	The user has changed their first name.
EPersonaChangeFacebookInfo	ChangedFacebookInfo (12)	The user has changed their facebook info.
EPersonaChangeNickname	ChangedNickname (13)	The user's nickname has changed.
EPersonaChangeSteamLevel	ChangedFacebookInfo (14)	The user's steam level has changed.
EPersonaChangeErr	Error (15)	Result Error.

3.5 SteamSetPersonaNameCallback

Result of entering a lobby.

Table 5: Returns **FSteamSetPersonaName**

Name	Type	Description
bSuccess	bool	true if name change completed successfully.
bLocalSuccess	bool	true if name changed was locally.
SteamSetPersonaEResult	(Integer) <i>EResult</i> .	result of the operation as an integer of <i>EResult</i> .

Steam Matchmaking Callbacks

4.1 GetMatchmakingCallback

Get reference to Matchmaking Callback Object to assign callback events.

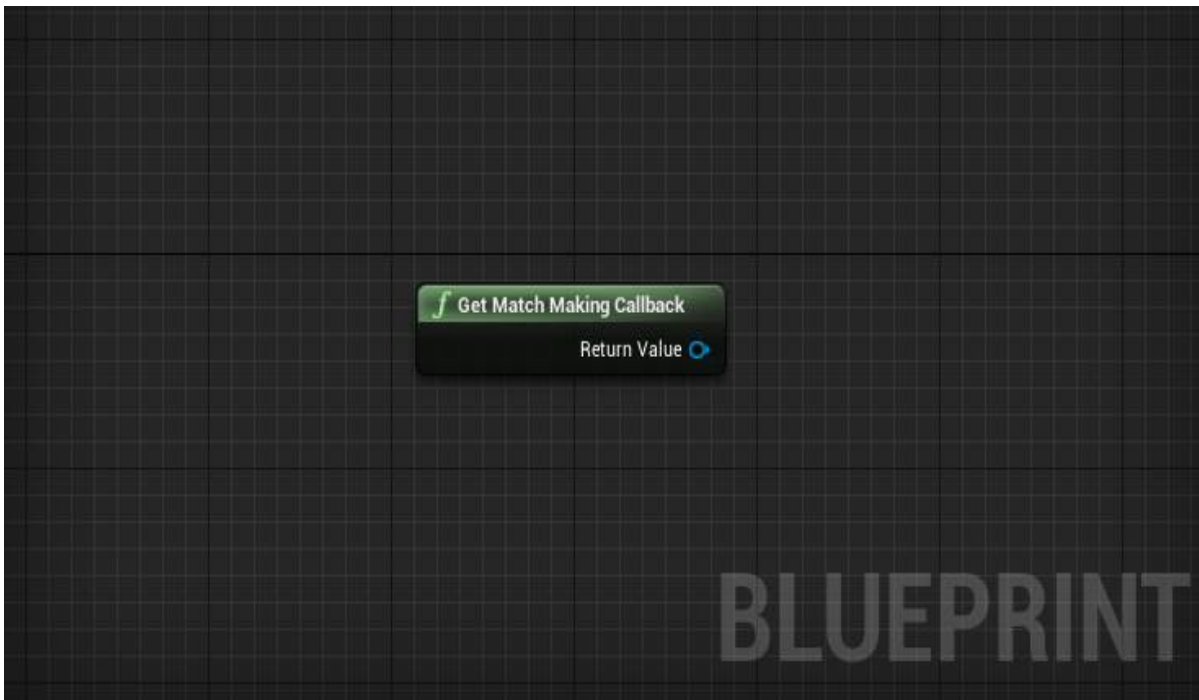


Table 1: Returns **PfMatchmakingCallbacks**

Name	Type	Description
Return Value	PfMatchmakingCallbacks	Pointer to matchmaking callback handler.

4.2 SteamLobbyUpdatedCallback

The lobby room state has changed, usually sent when a user has joined or leaves the lobby.

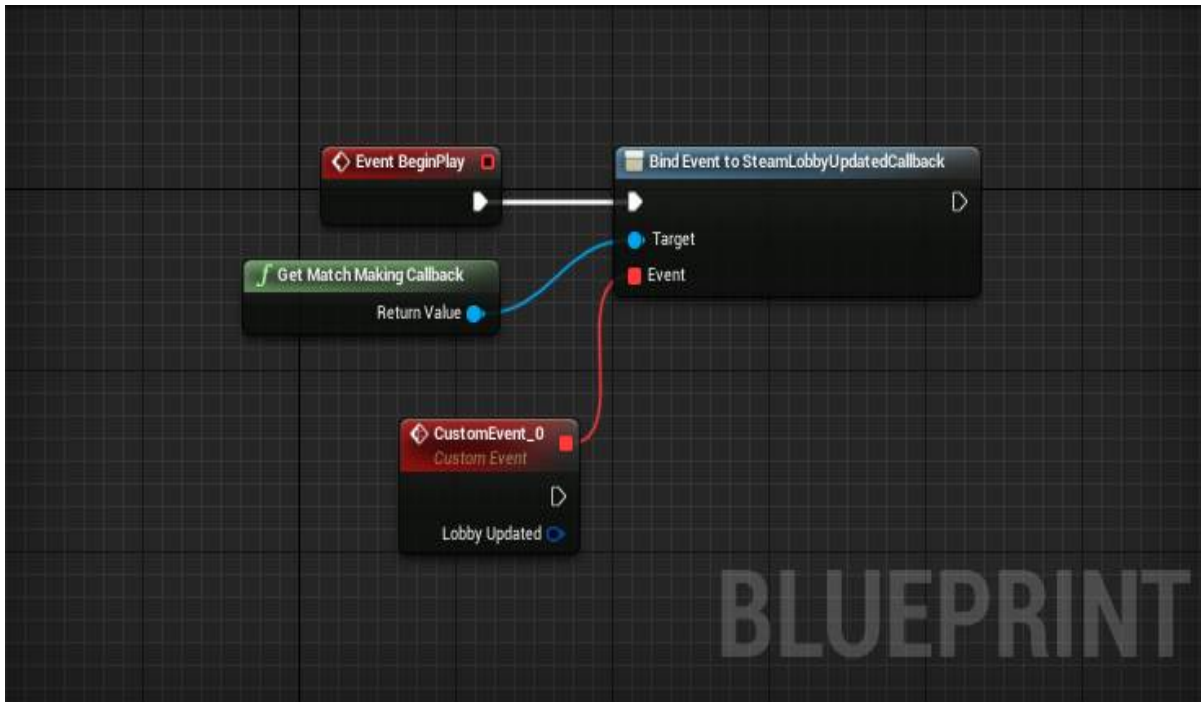


Table 2: Returns **FLobbyUpdatedStruct**

Name	Type	Description
SteamLobbyID	Integer	Steam ID of the lobby.
UserChangedSteamID	Integer	Steam ID of the user who's status has changed.
MakingChangeSteamID	Integer	Steam ID of user who made the change.
MemberStateChange	<i>EChatMemberStateChange</i>	Field of EChatMemberStateChange.

4.3 EChatMemberStateChange

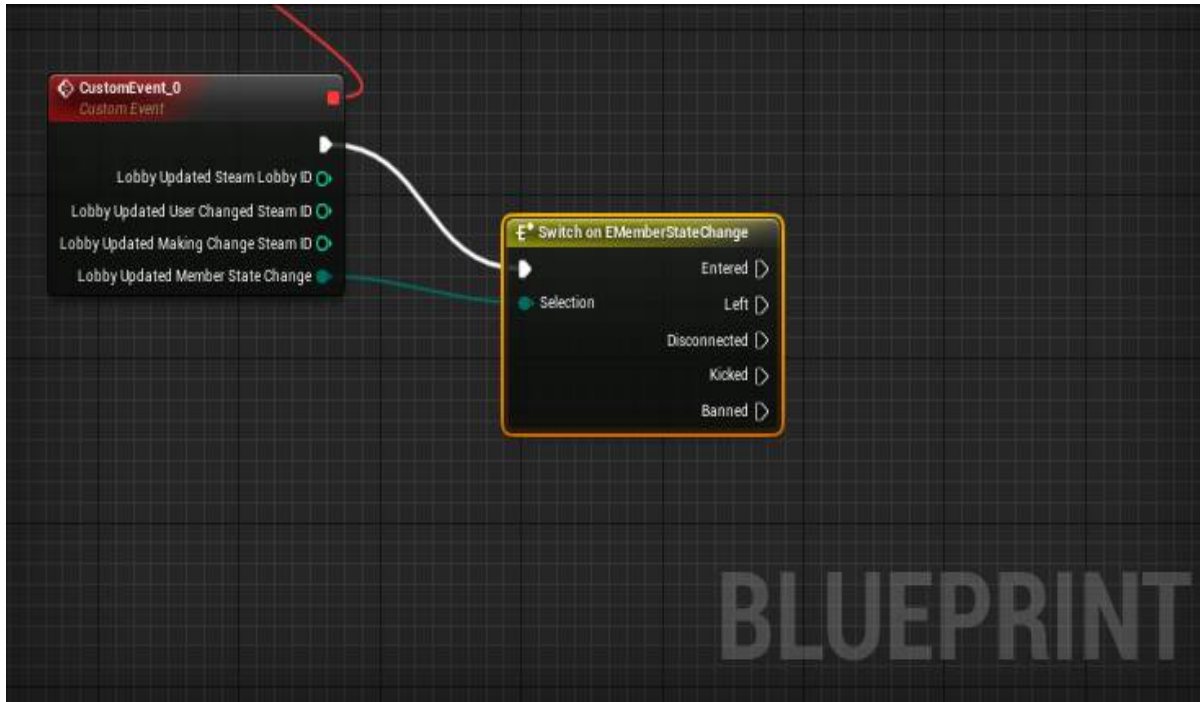
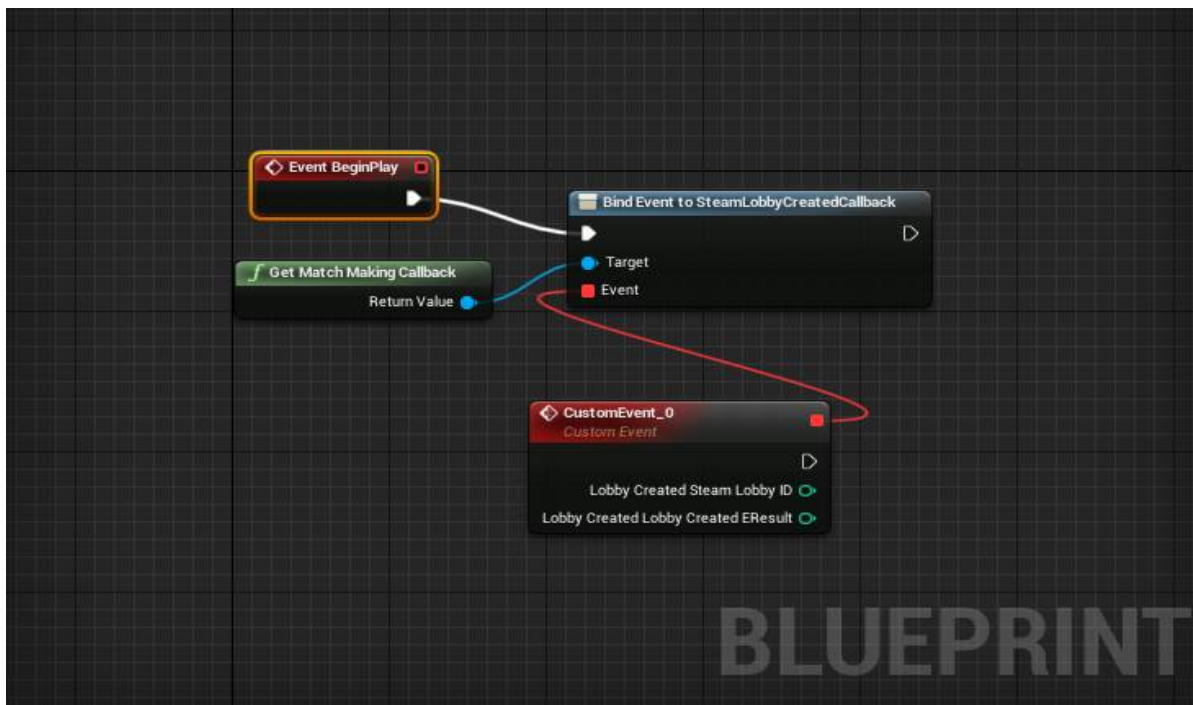


Table 3: **EChatMemberStateChange**

Name	Value	Description
ChatMemberStateChangeEntered	Entered (0x0001)	The User has joined or is joining the lobby.
ChatMemberStateChangeLeft	Left (0x0002)	The User has left or is leaving the lobby.
ChatMemberStateChangeDisconnected	Disconnected (0x0004)	The User has disconnected from the lobby.
ChatMemberStateChangeKicked	Kicked (0x0008)	The User has been kicked.
ChatMemberStateChangeBanned	Banned (0x0010)	The User has been kicked or banned.

4.4 SteamLobbyCreatedCallback

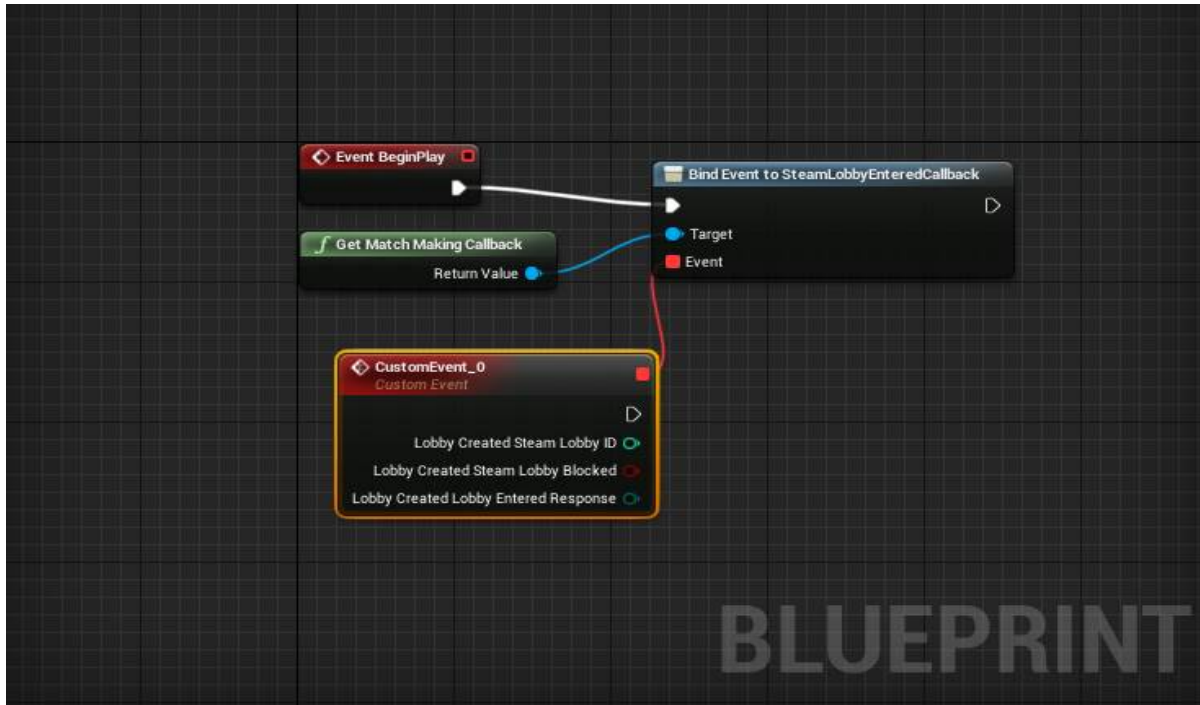
Result of a request to create a Lobby. Lobby has been joined and is ready for use at this point.

Table 4: Returns **FLobbyCreatedStruct**

Name	Type	Description
SteamLobbyID	Integer	Steam ID of the lobby.
SteamLobbyResult	(Integer) <i>EResult</i> .	result of the operation as an integer of <i>EResult</i> .

4.5 SteamLobbyEnteredCallback

Result of entering a lobby.

Table 5: Returns **FLobbyEnteredStruct**

Name	Type	Description
SteamLobbyID	Integer	Steam ID of the lobby.
SteamLobbyBlocked	bool	When true only invited users may join.
LobbyEnteredResponse	<i>ELobbyEnteredResponse</i>	Response to determine if lobby was entered successfully.

4.6 ELobbyEnteredResponse

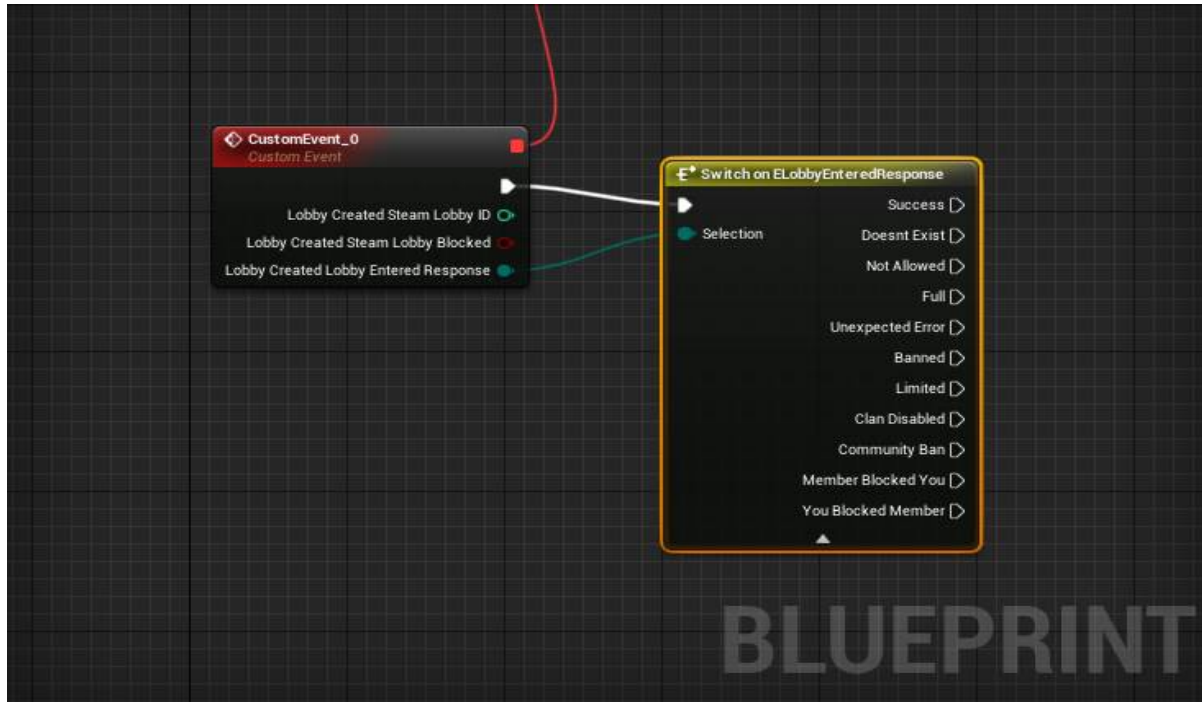


Table 6: **ELobbyEnteredResponse**

Name	Value	Description
EResponse_Success	Success (1)	Successful entry to chat/lobby.
EResponse_DoesntExist	DoesntExist (2)	Lobby/Chat doesn't exist (maybe closed).
EResponse_NotAllowed	NotAllowed (3)	Do not have permission to join.
EResponse_Full	Full (4)	Chat/Lobby room is full.
EResponse_UnexpectedError	UnexpectedError (5)	UnexpectedError.
EResponse_Banned	Banned (6)	The User has been banned from this lobby/chat and cannot join.
EResponse_Limited	Limited (7)	Cannot join this lobby because the user is limited.
EResponse_ClanDisabled	ClanDisabled (8)	Attempt to join a chat when clan chat is locked or disabled.
EResponse_CommunityBan	CommunityBan (9)	Cannot join this chat/lobby because the user is banned from the community
EResponse_MemberBlockedYou	MemberBlockedYou (10)	Cannot join this chat/lobby because a member in this chat/lobby blocked you.
EResponse_YouBlockedMember	YouBlockedMember (11)	Cannot join this chat/lobby because the user has blocked a user already in the chat/lobby

4.7 SteamLobbyInviteCallback

When an invite to join a lobby has been recieved.

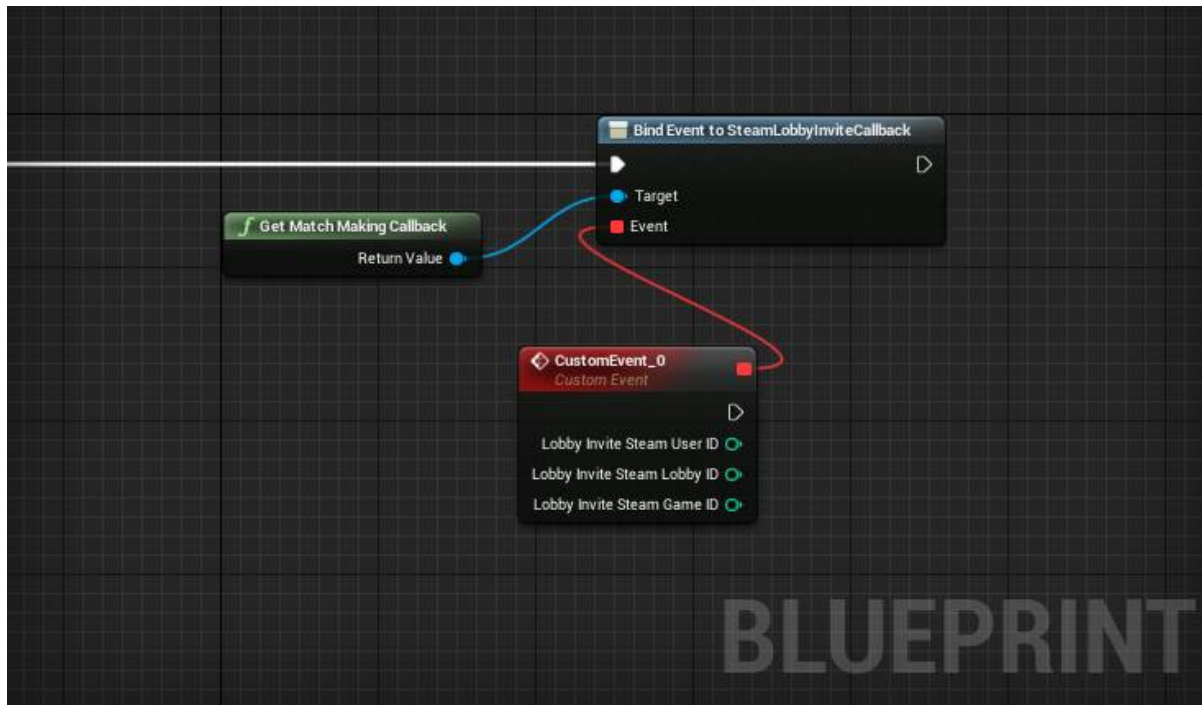


Table 7: returns FSteamLobbyInvite

Name	Type	Description
SteamUserID	integer	Steam ID of the person who sent the invite.
SteamLobbyID	Integer	Steam ID of the lobby we're invited to.
SteamGameID	Integer	Game ID of the lobby we're invited to.

Steam User Callbacks

5.1 GetUserCallback

Get reference to User Callback Object to assign callback events.



Table 1: Returns **PFUserCallbacks**

Name	Type	Description
Return Value	PFUserCallbacks	Pointer to user callback handler.

5.2 SteamGetAuthSessionTicketCallback

Called when creating an auth session ticket.

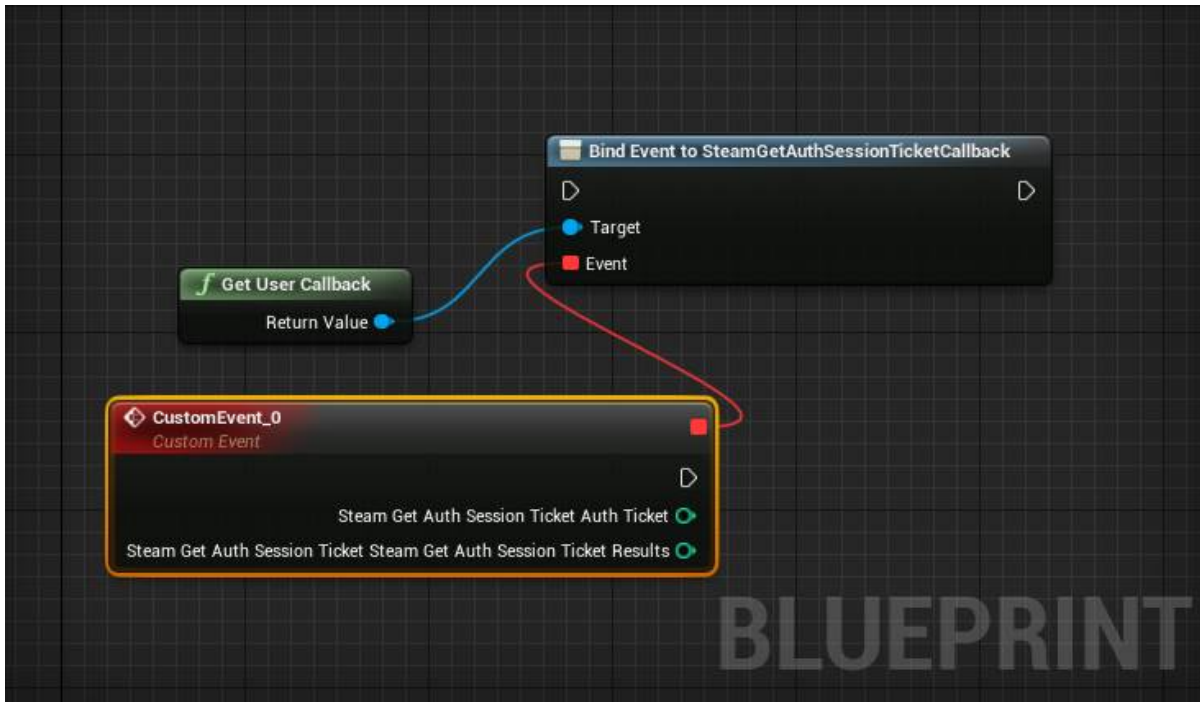


Table 2: Returns **FSteamGetAuthSessionTicket**

Name	Type	Description
AuthTicket	Integer	id of ticket created.
SteamGetAuthSessionTicketResults	Integer (EResult)	result of the operation as an integer of EResult.

5.3 SteamMicroTxnAuthorizationCallback

Called when a user has responded to a microtransaction authorization request.

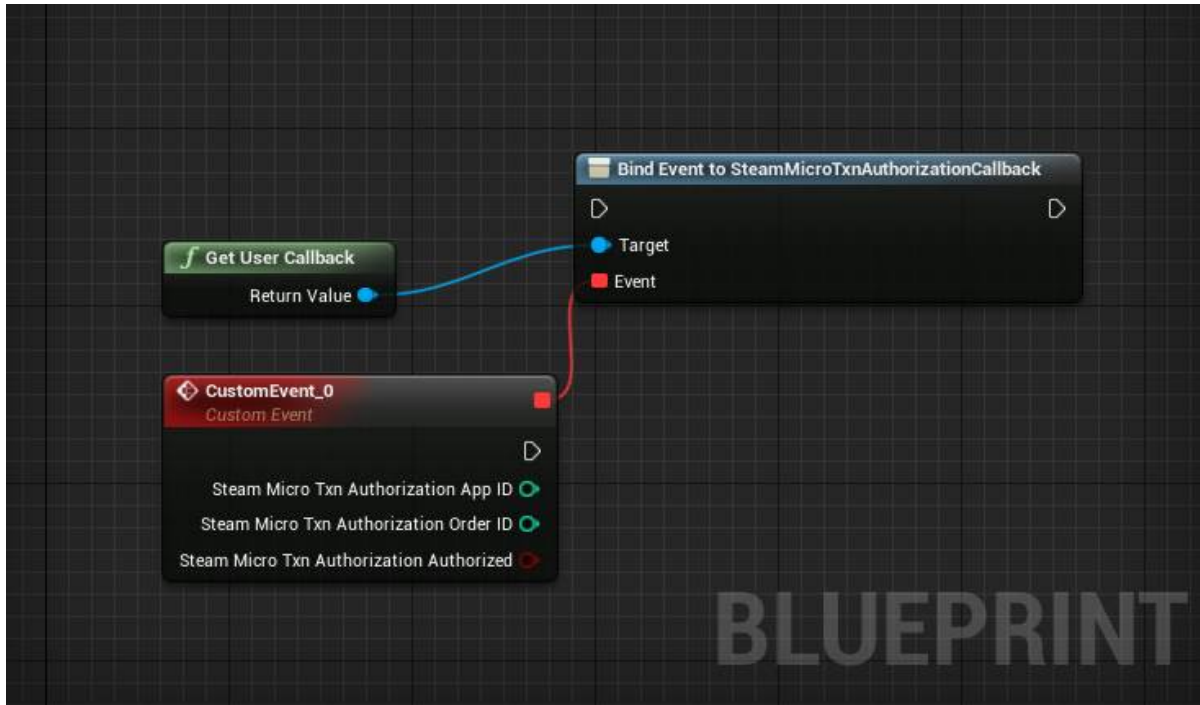
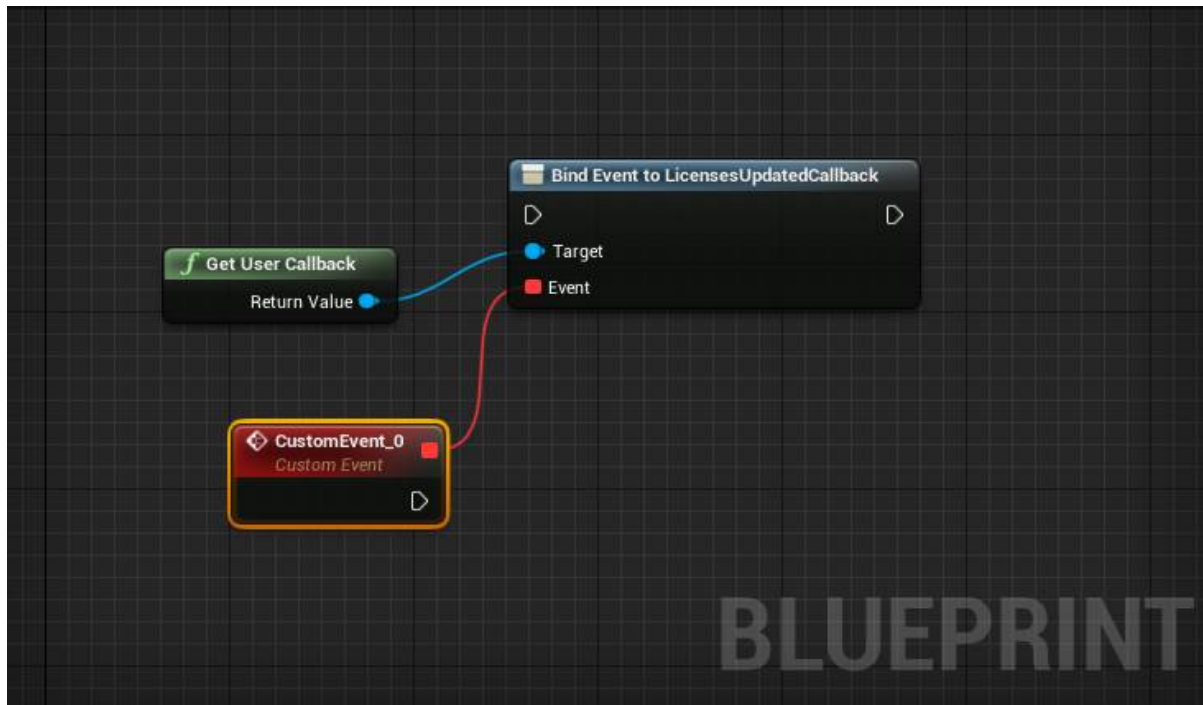


Table 3: FSteamMicroTxnAuthorization

Name	Type	Description
AppID	Integer	App ID for the microtransaction.
OrderID	Integer	Order ID provided for the microtransaction.
bAuthorized	bool	Did the user authorize the transaction.

5.4 LicensesUpdatedCallback

Called whenever the users licenses (owned packages) changes.



This callback has no return fields.

5.5 SteamServerConnectFailureCallback

Called when a connection attempt has failed.

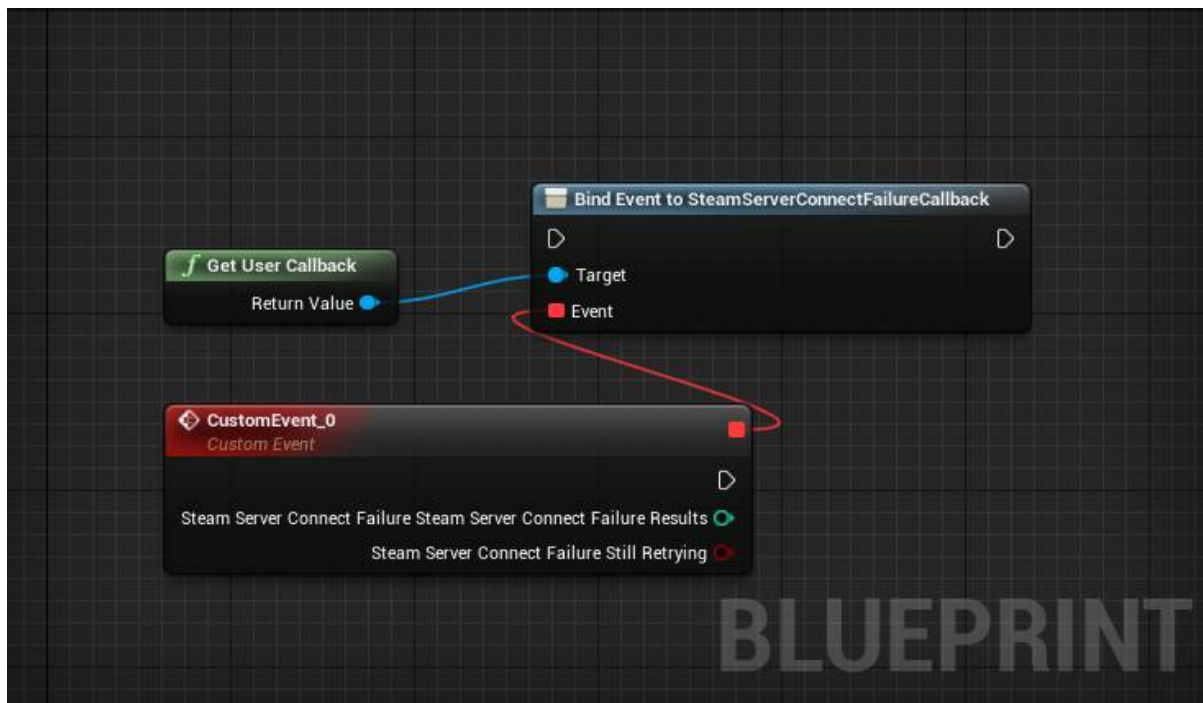
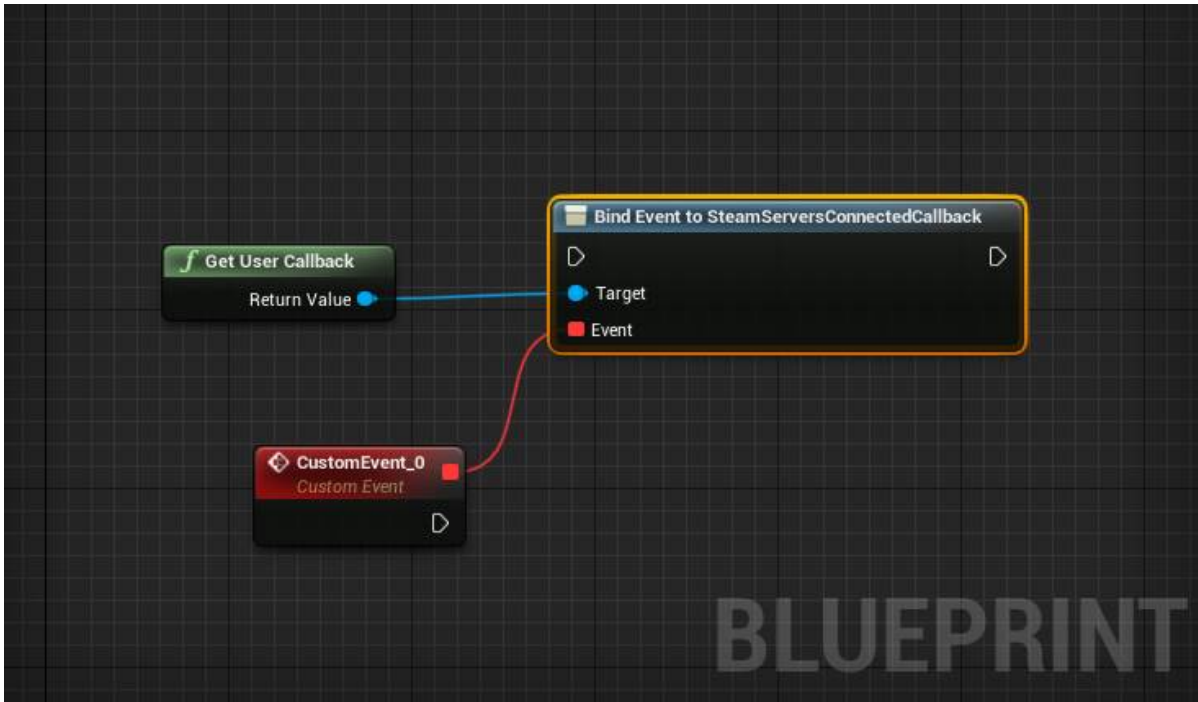


Table 4: Returns **FSteamServerConnectFailure**

Name	Type	Description
SteamGetAuthSessionTicketResult	Integer <i>EResult</i> .	result of the operation as an integer of EResult.
bStillRetrying	bool	Is steam client still trying to connect to the server?

5.6 SteamServersConnectedCallback

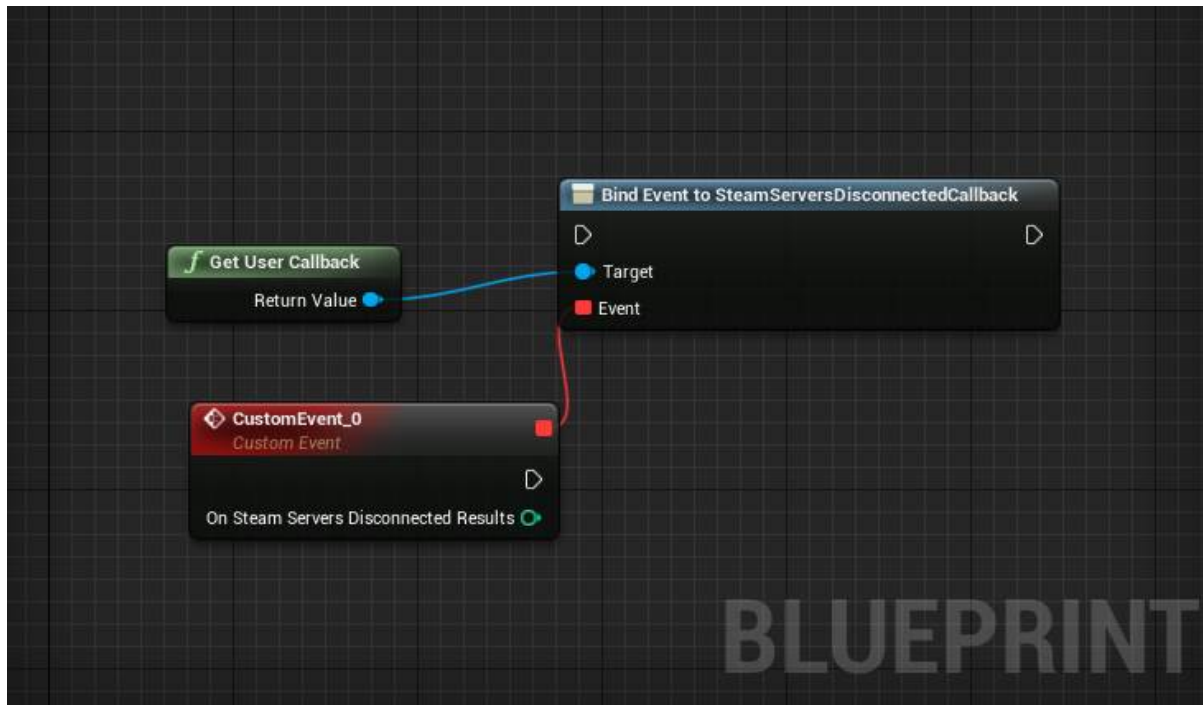
Called when a connection to the Steam back-end has been established.



This callback has no return fields.

5.7 SteamServersDisconnectedCallback

Called if the client has lost connection to the Steam servers.

Table 5: Returns **OnSteamServersDisconnectedResults**

Name	Type	Description
OnSteamServersDisconnectedResults	Integer (EResult)	result of the operation as an integer of EResult.