

Scar Achievements and Platform Services

Steam + EOS. One dropdown. One node.

User Guide - Version 3.0.0

Unreal Engine 4.27 to 5.8

Achievements, stats, leaderboards, friends, sessions, cloud save, rich presence, overlay.

Build-verified on UE 4.27, 5.2, 5.3, 5.5, 5.6, 5.7 and 5.8.

Black Scar Studio

Contents

1. What this plugin does
2. Install and enable
3. Quick Start (Local mode, 5 minutes)
4. Steam setup
5. EOS setup
6. Steam + EOS setup (dual mode)
7. Blueprint node reference
8. Feature walkthroughs
9. Setup wizard and diagnostics
10. Manual testing guide
11. Troubleshooting
12. FAQ
13. Support and version

1. What this plugin does

Scar Achievements and Platform Services is a Blueprint-first plugin that gives you one clean API for every online platform service a shippable game needs: achievements, stats, leaderboards, friends, sessions, cloud save, rich presence and the platform overlay.

Pick a platform in one dropdown. Call one node. The plugin routes the call to the right platform, handles login, retries, queuing and error reporting, and falls back to a local save file when no online platform is reachable. The same Blueprint works on Steam, EOS, both at once, or fully offline.

No third-party SDKs are bundled. The plugin uses Unreal's built-in Steamworks and EOS integrations, so it stays compatible across engine versions and adds nothing to a non-Steam build at runtime.

2. Install and enable

- Copy the **ScarAchievements** folder into your project's **Plugins** folder, or install from Fab.
- Open the project and rebuild if prompted.
- Confirm it is enabled: **Edit > Plugins**, search 'Scar'.
- For EOS or Steam + EOS, also enable **Online Subsystem EOS** and restart. Steam-only projects do not need this.

Engine support: 4.27 to 5.8. Download the zip that matches your engine version, or let the Epic Games Launcher install the matching build for you.

3. Quick Start (Local mode, 5 minutes)

Local mode needs no credentials and works immediately. Use it to build your UI before wiring up a store.

- **Edit > Project Settings > Plugins > Scar Achievements.**
- Set **Target Platform** to **Local / Offline**.
- In any Blueprint add **Award Achievement**, set Achievement Id to `ACH_FIRST_STEP`, connect it to a key press.
- Press Play in a **Standalone Game** window. You see a green toast and a log line. The unlock is saved locally and survives restarts.

That is the entire integration. Switching to Steam later is one dropdown change plus credentials; your Blueprints do not change.

4. Steam setup

Prerequisites: a Steamworks App ID (use 480 to test), the Steam client running and signed in.

- **Project Settings > Plugins > Scar Achievements.** Target Platform = **Steam**.
- **Steam App ID** = your App ID (480 for testing).
- Create a Data Table with row type **Scar Achievement Definition Row**, one row per achievement (Row Name = achievement API name).
- Assign it to **Achievement Definitions Table**.
- Click **Apply Platform Config** (writes DefaultEngine.ini and the Achievement_N_Id list Steam requires).
- Click **Validate Setup**, then restart the editor.
- Register the same achievement names in the Steamworks dashboard and publish to Live.

Steam does not authenticate in Play-In-Editor. Always test with the Play dropdown and Standalone Game.

5. EOS setup

Prerequisites: Product Id, Sandbox Id, Deployment Id, Client Id and Client Secret from dev.epicgames.com. Enable the Online Subsystem EOS plugin and restart.

- Target Platform = **Epic Online Services (EOS)**.
- Fill in the five EOS fields. Leave the **Encryption Key** blank to auto-generate a stable one.
- Click **Apply Platform Config**, **Validate Setup**, then restart.
- Register achievements and stats in the EOS Dev Portal.

EOS login uses Epic Account Services. For local testing use the EOS Developer Auth Tool and launch with the developer auth arguments below.

```
-AUTH_TYPE=Developer -AUTH_LOGIN=localhost:6300 -AUTH_PASSWORD=<yourCredentialName>
```

6. Steam + EOS setup (dual mode)

Dual mode dispatches every compatible write to both platforms at once. Steam is the primary; EOS is mirrored on top.

- Enable both **Online Subsystem Steam** and **Online Subsystem EOS**. Restart.
- Target Platform = **Steam + EOS**. Both credential groups are now shown; fill in both.
- Click **Apply Platform Config** (emits both platforms' INI), **Validate Setup**, restart.
- Call the same nodes you always call. Unlocks, stats, scores and presence go to both platforms.

Guarantee: a Steam failure never stops EOS and an EOS failure never stops Steam. If EOS is not logged in, mirrored writes are skipped and logged; the Steam copy still succeeds.

7. Blueprint node reference

Every node lives under a **Scar** category and needs no subsystem reference unless noted. Async nodes deliver results through an event dispatcher on the relevant subsystem.

Achievements (Scar | Achievements)

Node	Purpose
Award Achievement	The one-stop node: unlock or partial progress (0..1).
Unlock Achievement	Full unlock.
Set / Add Achievement Progress	Partial progress; commits at 1.0.
Increment Achievement (Steps)	Step-count wrapper for 'do X 100 times'.
Query All Achievements	Async refresh from the platform.
Is Achievement Unlocked	Cached bool check.
Get Achievement Info	One achievement: title, description, progress, unlocked.
Get All Achievements	The full cached array.
Get Active Achievement Platform	Local / Steam / EOS / Steam+EOS.
Print Achievement System Status	Full diagnostics to log and screen.

Stats (Scar | Stats)

Set Stat Int/Float, Add Stat Int/Float, Get Stat Int/Float, Get Stat Value, Get All Stats, Push All Stats To Online Platform.

Player Info (Scar | Platform | Player)

Get Player Name, Get Player ID, Get Platform, Is Player Logged In.

Overlay (Scar | Platform | Overlay)

Open Friends Overlay, Open Achievements Overlay, Open Invite Overlay, Open Store Overlay (Steam), Open Community Overlay (Steam), Open Web URL, Is Overlay Available.

On Overlay Toggled is an event dispatcher on the Scar Platform Subsystem. It fires with `Is Opening = true` when the platform overlay opens and `false` when it closes, on Steam and on EOS, so you can pause and unpause a single player game with no platform-specific nodes.

Rich Presence (Scar | Platform | Presence)

Set Rich Presence, Set Rich Presence Key/Value, Clear Rich Presence.

Friends (Scar | Platform | Friends)

Read Friends (async, bind On Friends Read), Get Friends. Friend fields: Display Name, Player Id, Is Online, Is Playing This Game, Status Text, Platform.

Leaderboards (Scar | Platform | Leaderboards)

Submit Leaderboard Score (bind On Leaderboard Submitted), Read Leaderboard (bind On Leaderboard Read), Get Leaderboard Entries. Entry fields: Player Name, Player Id, Rank, Score.

Sessions (Scar | Platform | Sessions)

Get Scar Session Subsystem, Create Session, Find Sessions, Join Session By Index, Destroy Session, Start / End Session, Get Found Sessions, Has Active Session. On Session Joined travels you to the host automatically.

Cloud Save (Scar | Platform | Cloud)

Cloud Save Game, Cloud Load Game, Is Cloud Available. Always writes a local mirror.

Diagnostics and Setup (Scar | Platform | Setup)

Print Platform Diagnostics, Run Setup Validation, Print Setup Report.

8. Feature walkthroughs

Achievement and progress

```
Event (enemy defeated)
  -> Award Achievement (Achievement Id = "ACH_FIRST_KILL")

On kill (progress achievement):
  -> Add Achievement Progress (Id = "ACH_100_KILLS", Delta = 0.01)
```

Stats

```
On kill:      Add Stat Int   (Stat Id = "STAT_KILLS", Delta = 1)
On lap done:  Set Stat Float (Stat Id = "STAT_BEST_LAP", Value = LapTime)
Read:        Get Stat Int   (Stat Id = "STAT_KILLS")
```

Leaderboards

```
Submit: Submit Leaderboard Score (Name = "HIGH_SCORES", Score = 12345)

Read:   Get Scar Platform Subsystem -> bind On Leaderboard Read
        Read Leaderboard (Name = "HIGH_SCORES", Count = 10)
        On Leaderboard Read (Success, Entries) -> For Each -> build a row
```

Friends

```
Get Scar Platform Subsystem -> bind On Friends Read
Read Friends
On Friends Read (Success, Friends) -> For Each -> show name + status
```

Sessions

```
Host:   bind On Session Created -> Create Session (4)
        On Success -> Open Level (YourMap, Options = "listen")

Client: bind On Sessions Found, On Session Joined
        Find Sessions (20) -> one server row per result (store Search Index)
        Join button -> Join Session By Index (Search Index)
        On Session Joined (Success) -> the plugin travels you to the host
```

Cloud save

```
Save: fill a USaveGame -> Cloud Save Game (obj, "PlayerProfile")
Load: Cloud Load Game ("PlayerProfile")
      On Cloud Load Complete (Success, SaveGame) -> Cast -> read fields
```

Rich presence and overlay

```
Set Rich Presence (Status Text = "In Main Menu")
Open Friends Overlay / Open Store Overlay / Open Web URL (Url)
```

Pause the game when the overlay opens

Bind **On Overlay Toggled** once on **BeginPlay**. It fires for the Steam overlay and for the EOS overlay, so one graph covers both platforms.

```
Event BeginPlay
-> Get Scar Platform Subsystem
-> Bind Event to On Overlay Toggled

On Overlay Toggled (Is Opening)
-> Set Game Paused (Paused = Is Opening)
```

In multiplayer do not pause; hide the HUD or stop input instead, because pausing a networked game only affects the local client.

9. Setup wizard and diagnostics

- **Project Settings buttons:** Apply Platform Config (writes INI, encryption key and achievement IDs), Validate Setup (pre-launch check), plus dashboard links.
- **At runtime:** Print Setup Report (red/yellow/green with fix hints), Print Platform Diagnostics, Print Achievement System Status.
- **Developer console (non-Shipping):** Scar.Diag, Scar.Award, Scar.Stat, Scar.Leaderboard, Scar.Friends, Scar.Presence, Scar.Session, Scar.Cloud.
- Launch with `-ScarAutoValidate` to run a one-shot self-test of every service after login and print the results.

10. Manual testing guide

Follow these steps to validate a build with real credentials.

- Set Target Platform and credentials, click Apply Platform Config, Validate Setup (expect all green), restart.
- Play > Standalone Game with the Steam client running (and EOS logged in for EOS or dual).
- From BeginPlay call **Print Setup Report** and **Print Achievement System Status**. Confirm Login = LoggedIn and, in dual mode, EOS Mirror Ready = YES.
- **Award Achievement:** confirm the green toast and the Steam overlay notification. In dual mode confirm the '[Dual] EOS mirror unlock committed' log line.
- **Set Stat Int:** confirm the stat write log lines.
- **Submit Leaderboard Score** then **Read Leaderboard:** confirm On Leaderboard Read fires with rows.
- **Read Friends, Open Friends Overlay, Set Rich Presence:** confirm each.
- **On Overlay Toggled:** press Shift+Tab with the Steam client running and confirm the event fires true on open and false on close.
- Host a session on one client and Join from a second: confirm travel.
- **Cloud Save Game** then **Cloud Load Game:** confirm the loaded value round-trips.
- Repeat for Steam, EOS and Steam + EOS. Read results in Saved/Logs/* . Log at full verbosity.

11. Troubleshooting

Symptom	Fix
'Using local fallback' when you picked Steam or EOS	Run Print Setup Report. Usually Apply Platform Config was not clicked, or the editor was not restarted.
Achievement write failed	The ID is not registered on the dashboard, or the schema is not published to Live.
EOS OSS not loaded	Enable Online Subsystem EOS in the .uproject and restart.
EOS AutoLogin failed	Credentials missing, or not logged in through Epic Account Services or the Dev Auth Tool.
Nothing happens in PIE	Steam does not authenticate in PIE. Use Standalone Game.
On Overlay Toggled never fires	The Steam client must be running and the overlay enabled for the app. It does not fire in PIE.
Cloud save reports success but nothing online	Enable Cloud for your app on the dashboard. The local mirror still works.

12. FAQ

- **Do I need C++?** No. Every feature has a Blueprint node.
- **Does it work offline?** Yes, Local mode saves to a file.
- **Can I switch platforms without changing Blueprints?** Yes. Change the dropdown and re-apply config.
- **Does dual mode double my code?** No. One node dispatches to both.
- **Which engine versions?** 4.27 to 5.8.
- **Is anything added to a non-Steam build?** No. The Steam SDK is delay-loaded.
- **Can I pause the game when the Steam overlay opens?** Yes. Bind On Overlay Toggled (see section 8).

13. Support and version

Version 3.0.0. Unreal Engine 4.27 to 5.8. Platforms: Windows, Mac, Linux. Build-verified on UE 4.27, 5.2, 5.3, 5.5, 5.6, 5.7 and 5.8.

For the full written manual, Blueprint and C++ API references, architecture and performance notes, the migration guide and the example content spec, see the **Docs** folder inside the plugin.

Black Scar Studio. Support: the Discord link in the plugin description.