OpenPrivateer Player Manual

Public Alpha

OpenPrivateer is a modern reimplementation of *Wing Commander: Privateer*. It runs the game natively on modern systems while using the original game data.

OpenPrivateer does not include the original Origin/EA game assets. You need your own copy of *Privateer* and, if you want to play it, *Righteous Fire*.

This is a player manual. The reverse-engineering notes and file-format documentation in the source tree go much deeper.

1. Quick Start

1. Download and extract OpenPrivateer.
2. Start `openprivateer.exe` on Windows, or the corresponding executable on Linux or macOS.
3. The launcher opens. Select **Privateer** or **Righteous Fire** at the top.
4. Open **Game Data** and provide the original game files:
   - Privateer: `PRIV.TRE` and `INITCFG.PAK`
   - Righteous Fire: `RF.TRE` and `INITRF.PAK`
5. Use **Import to Data Folder...** to copy files from another folder, or **Add Folder...** to use files where they already are.
6. Click **PLAY**, then **NEW**.

The launcher shows the exact files it found and the folder each one comes from. You can also put the files next to the OpenPrivateer program; the launcher searches the Program Folder automatically.

The INIT files contain the original new-pilot starting state. If the matching INIT file is missing, **NEW** is disabled.

You do not need *Righteous Fire* to play the original campaign.

SDL3 on Linux and macOS

The Linux and macOS archives include the SDL3 library (`libSDL3.so.0` or `libSDL3*.dylib`) next to the `openprivateer` program, and the program uses it automatically. The Linux archive needs no special unpacking: any extractor and any drive will do. To use your system's SDL3 instead (3.2.16 or newer), delete those `libSDL3` files from the program folder.

GOG or Steam installation

The GOG and Steam releases both include *Privateer* and *Righteous Fire*. In their installation folder, `INITCFG.PAK` and `INITRF.PAK` are ordinary files, and `PRIV.TRE` and `RF.TRE` are inside a disc image: `GAME.GOG` in the GOG release, `GAME.bin` in the Steam release. The launcher extracts them for you:

1. Open **Game Data**.
2. If the launcher found your installation, it shows “Found a GOG install in …” or “Found a Steam install in …” with an **Extract** button. Click it.
3. Otherwise click **Extract from GOG/Steam…** and select `GAME.GOG` or `GAME.bin` in your installation folder. **Import to Data Folder…** on that folder does the same.

`PRIV.TRE` and `RF.TRE` are copied out of the image, and `INITCFG.PAK` and `INITRF.PAK` from beside it, into the Data Folder. The progress bar shows how far it is; **Cancel** stops it and leaves no partial file behind. Each copy is checked against the image before it is used. If a different `PRIV.TRE` or `RF.TRE` is already in the Data Folder, the launcher asks before replacing it. If the image lacks one of the files, the others are still copied and the launcher says what is missing. The GOG or Steam installation itself is never changed.

Extraction runs on your own computer, from your own copy of the game. Nothing is downloaded, uploaded or shared, and OpenPrivateer never ships game files.

The launcher looks for installations in the usual folders: on Windows `GOG Galaxy\Games\Wing Commander Privateer`, `GOG Games\Wing Commander Privateer` and `SteamLibrary\steamapps\common\Wing Commander Privateer` on every local drive, and the default GOG Galaxy and Steam folders under Program Files; on Linux and macOS `GOG Games/Wing Commander Privateer` and Steam’s own library in your home folder. On Linux it also looks in every Steam library Steam itself lists (a microSD card or a second drive included, also with the Flatpak Steam), in the folders directly inside `~/Games`, and on mounted storage under `/run/media`, `/media`, `/mnt` and `/var/mnt` (their `SteamLibrary`, `steamapps`, `GOG Games` and `Games` folders); network drives are skipped. Anywhere else, use **Extract from GOG/Steam…** and select the file: any location works. On macOS, if your GOG copy is an application bundle, `GAME.GOG` may be inside it; in Finder, right-click the bundle and choose **Show Package Contents** to find it.

Steam Deck and Steam Frame

OpenPrivateer runs as a native Linux program on SteamOS; use the Linux archive that matches the device’s processor.

1. In Desktop Mode, extract the archive into a folder of your own, for example `~/Games/OpenPrivateer`.
2. Mark `openprivateer` as executable (in Dolphin: right-click it, **Properties**, **Permissions**).
3. Start it once from Desktop Mode and set up the game files in the launcher (it finds a Steam copy of the game on the internal drive and on a microSD card).
4. To play in Gaming Mode, add `openprivateer` to Steam with **Add a Non-Steam Game to My Library**. Do not choose a compatibility tool (Proton) for it: Proton runs Windows programs, and this one is a Linux program.

Keyboard and mouse work as on a desktop. The game’s own gamepad layout (section 6, “Default gamepad layout”) applies when Steam passes the controls to the game as a gamepad; if Steam Input sends them as keyboard and mouse instead (its desktop layout), pick a gamepad layout for OpenPrivateer in Steam’s controller settings or bind the controls there.

The commands that have no button of their own (Autopilot, Jump, the MFD pages, the wingman orders, Options, Quit and more) are on the **control wheel** (section 6, “Control wheel”): click the left stick to

open it. Under Steam's desktop layout, map a Deck button to `F12`, or use the trackpad's middle click. On the Deck's touchscreen, tap the round handle in the black bar on the left to open the wheel and tap a command. When the game receives the touchscreen as touch, the handle appears after your first touch; when it receives it as a mouse (this depends on Steam's settings), set **Touch Handle** to **Always** under Settings, **Control Wheel**.

Steam's on-screen keyboard comes up when you click or tap into a text box (a pilot's name and callsign, a save name, a ship's name, the server address and the other fields in the launcher), placed beside the box, and closes when you leave it. Steam's keyboard closes itself on Enter; to bring it back while the box is still open, click or tap into the box again. See "The on-screen keyboard does not appear" in section 27.

The session log is in `~/.local/share/OpenPrivateer/logs/`; send the newest file with a report. To remove OpenPrivateer's saves, settings and copied game files from the device, see "Removing OpenPrivateer's files" in section 2.

2. First Start and the Launcher

The launcher is OpenPrivateer's startup screen. It selects the game, locates the original game data and provides the main startup settings before the engine loads the game.

At the top, click the **Privateer** or **Righteous Fire** picture to choose the game. Your own `wallpaper/` artwork is used when present (section 19). On the left, the launcher shows the selected game's TRE and INIT file, their status and the exact path being used. **PLAY** stays disabled until the selected game's TRE file is found.

When a newer OpenPrivateer build is available, a line under the build number says **Update available** with the build's name and a **Download** button. Download opens the newest release in your web browser; nothing is downloaded or installed automatically (section 23).

The launcher sections are:

- **Experience**: choose Classic, Enhanced or Modern (section 4).
- **Language**: choose the language of OpenPrivateer's own texts and, with Game Data folders added for a language, of the game files ("Language" in section 3, "A folder for one language" below).
- **Multiplayer**: enter a server to join and your callsign, check the server before joining, or pick one from your favourites, your recent servers, the local network or a server list under **Servers**, or start a server for your friends under **Host a session** (section 22); a server that asks for a password gets a **Password** field.
- **Game Data**: manage where OpenPrivateer finds the four original game files.
- **Mods**: switch installed mods on and off, set their load order, choose the start mod and see what the last start did with each mod (section 20, "Managing mods in the launcher").
- **Music**: choose the music backend and an optional SoundFont (section 17).
- **Original Developer Switches**: set the original `-K`, `-B` and `-D` startup switches (section 24).
- **Files and Diagnostics**: open user-data folders, move the user-data location, reset settings, remove OpenPrivateer's files, copy diagnostics and control update checks.

Untick **Show this screen at startup** to go straight to the game next time. Hold **Shift** while starting OpenPrivateer to bring the launcher back. The launcher is always shown when required game data is missing.

Game Data

OpenPrivateer searches the Game Data rows from top to bottom and uses the first copy of each required file it finds. The table shows all four files and marks the copy currently selected as **Used**.

The **Type** column describes each search location:

- **Custom**: a folder added by the player. Its files are used in place.
- **Language**: a folder added for one language (see below). It is searched first, and only while the **Language** setting is that language.
- **Data Folder**: the `data/` folder where the launcher stores imported or copied game files.
- **User Home**: the current user-data folder.
- **Program Folder**: the folder containing the OpenPrivateer program. In portable mode this is also the user-data folder.

Command-line and environment search locations are identified by their corresponding source instead.

Other table states help explain duplicate files. **differs** means a different copy exists further down the search path but is not used. **same** means the later copy is identical. **case!** reports a filename with the wrong letter case on systems where that matters.

Folders added as **Custom** can be moved up or down or removed. A path can also be entered directly and added with **Add**. **Rescan** checks the search locations again.

A folder for one language. The small field between the path and **Add** takes a language code (`es`, `en`, ...). A folder added while it holds a code, by **Add** or **Add Folder...**, is used only while the **Language** setting is that language, and then it overrides every other choice of game files: it comes before every other row, before `--data` and `OPENPRIV_DATA`, and before a game file named on the command line. The line above the table says which files the chosen language's folders supply. This lets you keep the English game and a translated edition side by side: add the English folder with `en` and the Spanish one with `es`, then pick the language in the launcher's **Language** section; the table shows the chosen language's folders at once and **PLAY** starts with them. Switching **Language** in the debug overlay while playing restarts the game with the matching game files. A file the language's folders do not have (for example a translation of Privateer only) comes from the other rows. A language with no folder of its own uses the other rows as usual, a game file named on the command line first. Folders registered for another language than the current one are listed under the table, where they can be removed. Saves, settings, mods and the `content/` folder are the same for every language.

The four file actions are different:

- **Add Folder...** adds a folder to the search path and leaves its files where they are.
- **Import to Data Folder...** selects a folder and copies recognized game files from it into the Data Folder (from a GOG or Steam installation folder, it extracts them from `GAME.GOG` or `GAME.bin`).
- **Copy Found Files to Data Folder** copies the game files currently being used into the Data Folder.

- **Extract from GOG/Steam...** selects a GOG `GAME.GOG` or a Steam `GAME.bin` and extracts the game files from it into the Data Folder (section 1, "GOG or Steam installation").

When OpenPrivateer recognizes a likely existing installation, the launcher can also offer to use its files in place, copy the files it finds, or, for a GOG or Steam installation, extract them.

Where your files live

Settings (`openprivateer.ini`), savegames, logs and optional folders live together in OpenPrivateer's user-data folder.

By default this is:

System	Folder
Windows	`%APPDATA%\OpenPrivateer`
Linux	`~/.local/share/OpenPrivateer`
macOS	`~/Library/Application Support/OpenPrivateer`

A portable installation instead keeps user data in the Program Folder. An existing `openprivateer.ini` next to the program keeps older portable installations working. An empty `portable.txt` can also be placed there to request portable mode.

Open **Files and Diagnostics** to see the folder currently in use. The section has direct buttons for the user-data folder, savegames, logs, music, wallpaper, portraits and content.

Change Data Folder... selects another location for user data. When switching folders, OpenPrivateer can either use the destination as it is or copy the current settings, saves, content, music, wallpaper, portraits and copied game files there before switching. Nothing is deleted from the old location. Depending on the current mode, the launcher also offers **Move to User Folder...** or **Use User Folder...**.

If the location was fixed with `--home` or `OPENPRIV_HOME`, it cannot be changed from the launcher for that run.

Reset Settings... renames `openprivateer.ini` to `openprivateer.ini.bak` and restarts OpenPrivateer. Savegames are not touched.

OpenPrivateer creates its optional folders and README files in the user-data folder automatically. The README files are rewritten on startup so their instructions stay in sync with the installed engine.

Folder	Purpose
`savegame/`	Original and OpenPrivateer savegames
`music/`	Optional replacement music
`wallpaper/`	Optional game-selection artwork
`portraits/`	Optional: your pilot's own portrait (see section 9),
	in `portraits/pilots/` the copies of the portraits
	you chose, and in `portraits/examples/` a drawn

Folder	Purpose
	example of talking and blinking pictures
`content/`	Content overrides, translations, wingman names, the
	experimental `content/mods/universe/` map extension,
	the experimental `content/mods/weapons/` weapon mods,
	`content/mods/items/` Ship Dealer mods,
	`content/mods/init/` start mods,
	`content/mods/missions/` mission mods,
	`content/mods/ships/` ship mods,
	`content/mods/interiors/` base interior mods and
	`content/mods/packs/` mod packs (see section 20), and
	`content/hud/hud_style.json`, your own style for the
	cockpitless HUD, and `content/hud/radar_colours.json`,
	your own radar colours (see section 14)
`mods.json`	Optional: the mods' load order and the disabled mods
`alpha/`	Optional material from the August 1993 Privateer alpha
`data/`	Game files imported or copied by the launcher
`logs/`	Session logs

You can ignore all of these folders when starting out.

Removing OpenPrivateer's files

Deleting the OpenPrivateer program does not delete its user-data folder: your saves, settings and any game files the launcher copied stay there. To remove them, open **Files and Diagnostics** and click **Remove game data…**. The dialog shows the folder in use and, for each of its two choices, every entry it would delete with its size. Nothing is deleted until you confirm.

- **Remove copied game data only** (the highlighted choice) deletes the game files the launcher copied or extracted into the `data/` folder: `PRIV.TRE`, `INITCFG.PAK`, `RF.TRE` and `INITRF.PAK`, and any half-finished copy of them. Your pilots and saves, settings, portraits, mods, music and logs stay. Use it to free the space of the copies, for example when you keep the game files in place elsewhere.
- **Remove everything…** deletes the whole user-data folder: your pilots and saves, settings, portraits, mods, music, logs and the copied game files. It asks a second time: tick **I understand that my pilots, saves and settings are deleted**, then click **Delete everything**. OpenPrivateer closes when it is done, and starts with default settings and no saves the next time. In a folder you chose yourself (**Change Data Folder…**, `--home` or `OPENPRIV_HOME`), only OpenPrivateer's own files and folders are deleted and anything else in it is listed as kept.

Neither choice touches your original game: a game folder added under Game Data, a Steam or GOG installation, or files on a memory card or another drive stay as they are. A link (symbolic link or junction)

inside the folder is removed as a link; what it points to is kept. Afterwards the dialog says how much space was freed, or which entries could not be removed (usually because another program has them open; close it and try again).

The launcher does not offer the removal for a portable installation (the user-data folder is the program's own folder: delete that folder to remove everything), for a folder reached through a link, or for a drive or your home folder itself.

Removing the files by hand. Close OpenPrivateer, then delete the user-data folder:

System	Delete this folder
Windows	`%APPDATA%\OpenPrivateer` (type it into the Explorer
	address bar; usually
	`C:\Users\<name>\AppData\Roaming\OpenPrivateer`)
Linux	`~/.local/share/OpenPrivateer` (or
	`$XDG_DATA_HOME/OpenPrivateer` when that is set); on a
	Steam Deck in Desktop Mode, show hidden files in Dolphin
	to see `.local`
macOS	`~/Library/Application Support/OpenPrivateer` (in Finder,
	Go > **Go to Folder...**)

If you moved the user-data folder, delete the folder shown in **Files and Diagnostics** instead, and also the default folder above, which still holds the setting that points to the other one and any files a move left behind there.

3. Privateer and Righteous Fire

OpenPrivateer supports both the original *Privateer* campaign and *Righteous Fire*.

Choose the game by clicking its picture at the top of the launcher. With the launcher turned off and both `PRIV.TRE` and `RF.TRE` found, a startup selector lets you choose instead. The selection is remembered. You can also change the game later from the settings/debug overlay. Changing games while playing relaunches OpenPrivateer because the two modes use different data archives.

Importing a Privateer pilot into Righteous Fire

The original Righteous Fire import flow is supported.

If Righteous Fire is selected and the `savegame/` folder contains Privateer `.SAV` files, starting **NEW** can ask whether you want to load an old Privateer ship.

The import carries over the pilot, ship, credits, reputation, kills, guild memberships and missions-to-date state according to the original import behavior. Privateer cargo, side missions and plot progress do not become Righteous Fire campaign state. In Enhanced and Modern, a portrait you chose for the pilot comes along too.

Righteous Fire equipment in Privateer

With `RF.TRE` in the game data folder, a Privateer game can sell Righteous Fire's equipment at the Ship Dealer: turn on **All Equipment** in the debug overlay (directly above Open Universe). It is off by default, works in Enhanced and Modern, and takes effect at once. See section 11 for what it adds.

Playing a translated edition (Spanish)

OpenPrivateer can play the Spanish translation of *Wing Commander: Privateer*, version 1.3, by L@Zar0, from the old ClanDLAN (with *Righteous Fire*). OpenPrivateer ships no translated files: you install the translation over your own copy of the game as its readme describes, then use that folder as the game data.

1. Install the Spanish translation into a copy of your game folder (it replaces `PRIV.TRE` and `RF.TRE`).
2. Point OpenPrivateer at that folder: add it as a **Game Data** folder in the launcher, or start with `--data <folder>` (for example `openprivateer --data D:\PrivEsp`). To keep your English game as well, add the Spanish folder with the language code `es` and the English one with `en` (see "Game Data", "A folder for one language"); the **Language** setting then picks the game files.
3. Start as usual. The game finds the Spanish fonts by itself and shows accented letters wherever the translation has them.
4. For OpenPrivateer's own texts in Spanish too, set **Language** to **Español** (see below), in the launcher or in the debug overlay.

What you see in Spanish: the conversations, the comm messages, the mission and guild texts, the bar and base texts, the names on the hover line of every base screen, the Ship Dealer's and the Software Booth's item names and messages, the Commodity Exchange's labels and messages, the Mission Computer's labels, the base, system and commodity names on the Mission Computer, the navigation map and the cockpit, the star system names, the cutscene captions, the translated pictures and the dubbed intro. The doors, the dealers and the booths work exactly as in English, and mods that name stock bases, systems, items or the base doors in English work with the Spanish data too.

The comm voices are English in the translation itself. The notification inbox, the save browser's details and the debug overlay stay English.

Language

Language (the launcher's **Language** section, or the debug overlay's Settings while playing) chooses the language of OpenPrivateer's own texts: the cockpit readouts and messages, the navigation map's labels, the guild and mission terminal texts, the Ship Dealer's hardpoint names, the conversation's Accept and Reject, the pause plaque and the mission progress lines. **English** (`en`) and **Español** (`es`) are built in; a file `openprivateer_<code>.i18n` in the `content/i18n/` folder of your settings folder adds a language or changes single lines of a built-in one. In `openprivateer.ini` it is `Language=en` or `Language=es`. Each language is listed by its own name (**English**, **Español**); a language a Game Data folder is added for is listed too. Changing it in the debug overlay restarts OpenPrivateer; in the launcher it takes effect when you press **PLAY**.

With a translated edition as the game data, the texts that edition translated itself (the Commodity Exchange, the Software Booth, the Ship Dealer's messages, the Mission Computer's labels) show the edition's own wording whatever the Language setting.

In Spanish, a few one-letter answers follow the language: the Weapons display's gun and missile lines start with **C:** and **M:**, the Mission Computer's standings show **A** for a friendly faction, and the Righteous Fire question "Load an old Privateer ship?" is answered with **S** (yes) or **N** (no). A Mission Computer line too long for the screen moves left so it stays readable.

Font Charset (debug overlay, under **Language**) chooses how text is drawn with the game's fonts: **auto** (the default) looks at the fonts and picks **spanish** for the translation's font set and **english** for the original fonts; **latin1** is for a custom font with accented letters at their ISO-8859-1 codes. In `openprivateer.ini` it is `FontCharset=auto`, `english`, `spanish` or `latin1`. Changing it restarts OpenPrivateer. With the Spanish fonts, a translation file (`content/i18n/`) and a mod's Spanish texts show their accented letters; with the original fonts accents are dropped (`á` shows as `a`).

The pilot name and callsign you type on the Mission Computer stay plain letters: a letter typed with an accent is taken without it (`José` becomes `Jose`), so saves and multiplayer callsigns read the same everywhere.

Comm messages that come with a voice show no caption unless **SUBTITLES (force) > Inflight** is on in the debug overlay.

4. Experience Modes

OpenPrivateer has three experience settings: **Classic**, **Enhanced** and **Modern**. Enhanced and Modern currently use the same gameplay behavior; Modern is reserved for future changes.

Classic

Classic aims to reproduce the original gameplay, mechanics and feel. It is not a DOS emulator and does not deliberately reproduce every old hardware limitation or arithmetic accident. OpenPrivateer-only gameplay features such as multiple-ship ownership are inactive.

Enhanced and Modern

Enhanced and Modern keep the original game underneath but add OpenPrivateer conveniences and optional features.

Flight

- **Escape lands at a nearby base:** `Esc` in flight lands you at the nearest base, but only while you are inside that base's landing zone (5000 units). With no base that near it does nothing. In Classic, `Esc` does nothing in flight. With **Hail To Land** on (below), `Esc` first asks that base for a landing.
- **Hail to land:** flying close to a base or planet does not land you by itself; you ask, and the base answers. Inside its landing zone (5000 units) pick the base on the Communications MFD: its list starts with "Request landing clearance.". Send it, and the base answers "Entering automatic landing zone" (spoken, or on the message line), the landing music starts, and you are cleared to land: fly in

closer than 1000 units, or press `Esc`, and you land. Pressing `Esc` before you are cleared sends the same request. Closer than 2000 units the list also shows "Requesting immediate automatic docking" at the top, which lands you at once without asking; it goes again if you move farther away, so the other lines move down or up by one (the request line is key 1 outside 2000 units and key 2 inside). Leaving the landing zone cancels the clearance. The base does not announce its landing zone when you fly in, and the flight music keeps playing until it answers. At a carrier, and at a mod base whose landing animation has music of its own, the answer starts no landing music: the flight music plays on until you land, and the landing animation then plays its own music, or the landing music when it has none. A base whose station has turned hostile to you refuses ("Docking refused"). To land automatically as in Classic (within 3500 units, no request needed, with "Entering automatic landing zone" and the landing music on entering the zone), untick **Hail To Land** in the GAME section of the settings overlay. Classic always lands automatically.

- **Bases by name on the Communications MFD:** the list of contacts names each base and planet as the Navigation Map does ("Jolson" rather than "Agricultural Base"), and the messages a base sends start with that name too ("Jolson: …" rather than "Agricultural Base: …"), on the message line and over the landing movie. A name too wide for the MFD ends in "…", and so does a very long name in a message. Ships keep their usual names. Classic names bases as the original does.
- **Cockpit hide:** `.` hides the cockpit background art while the HUD stays visible, in every cockpit view including the turret views (`F5` / `F6`). The setting applies to all views at once and lasts until you press `.` again, the next flight included. It is unbound in Classic.
- **Red crosshair on the lead:** in the front view the crosshair of every cockpit turns red, the red of the box around an enemy target, while it is on a gun's lead marker (the ITTS markers, switched with `I` on scanners that have them) and for half a second after, so even a brief pass shows. A shot then is aimed where the target will be. Classic keeps the crosshair's own colour.
- **ITTS plaque:** `I` shows "ITTS Engaged" or "ITTS Disabled" for 3 seconds at the top of the screen, where F7's "Missile Camera On" plaque appears, following the **Plaque Position** setting (Off hides it). Classic shows nothing.
- **Cockpitless HUD:** `,` (comma) switches between the cockpit and a HUD drawn as thin lines over open space, on every ship (see "The cockpitless HUD" in section 14). It is unbound in Classic.
- **Third-person view:** `F10` (or a gamepad's right stick click) puts the camera above and behind your ship with a light HUD around it, in the style of an arcade flight game; you fly and fight from it as from the cockpit (see "The third-person view" in section 14). `F8` stays the original chase camera. Classic has no such view.
- **Your view is remembered across a launch:** the game remembers whether you were flying with the third-person view (`F10`), with the cockpitless HUD (`,`), with the cockpit art hidden (`.`) or with the third-person view's HUD hidden when you land, and starts the next flight the same way, for as long as the game is running. Loading a save or starting a new game begins in the cockpit. Until you press `,` once, each ship starts in its own usual HUD; after that, every ship starts in the one you chose last. Every flight still starts looking forward: the turret views and the `F7` to `F9` cameras are not remembered.
- **Rolled sprites keep their shape:** a planet stays round at any roll. Classic keeps the original oval distortion.
- **Look around your ship in F8 and F9:** in the chase view (`F8`) and the view from above (`F9`), moving the mouse turns the camera around your ship: mouse right looks right, mouse up looks up, as far as nearly straight above or below the ship. Press `F8` or `F9` again to straighten the view; switching views straightens it too. It works while the game is paused, and the missile camera (`F7`) is

not turned. Whichever input device you fly with, the mouse turns the camera in these two views; with Mouse selected the ship flies on without mouse steering there and the mouse steers again when you return to the cockpit. **View Look Speed** and **Invert View Look** set how fast it turns and which way up and down go. Classic has no mouse look. On Windows the game keeps the hidden pointer in its window and moves it back to the middle as you turn, instead of reading the mouse's raw movement (**Mouse Relative Warp** in the debug overlay's EXPERIMENTAL section, On by default there, Off on other systems). If the mouse does not turn the camera on your computer, try switching it.

- **The keyboard steers alongside your input device:** the keyboard adds to the device selected on the Options screen, so you can, for example, roll with the keypad keys while turning with the mouse or stick. The mouse flies the ship only with Mouse selected and a joystick or gamepad only with Joystick selected. In Classic only the device selected on the Options screen steers. Each device's own roll works the same in every mode: the mouse's right button and the original joystick's button 2 switch from turning to rolling, while a gamepad's right stick or a joystick axis assigned to Roll rolls while you turn.
- **Mouse flight is easier to control:** one even dead zone, the same sensitivity on both axes, and pushing the mouse up pitches the nose up. A second crosshair follows the pointer so you can see where you are steering, and **Mouse Sensitivity** applies. Classic keeps the original mouse flight: no visible pointer, and mouse up pitches the nose down.
- **Weapon state is remembered across a launch:** the gun on/off combination, last selected ordnance type, ITTS and shield power level (`S` / `Shift+S`) are restored for as long as the game is running. ITTS only comes back on a ship whose scanner has ITTS. Loading a save or starting a new game begins from the defaults. Classic launches with all guns off, ITTS off and shields at full power.
- **Gun keys 1-4 address hardpoints:** each key flips one forward hardpoint. In Classic they flip a weapon-type group, so four identical guns all flip together on `1`. A ship from a ships mod can carry up to eight forward guns; keys 5-8 flip its guns 5-8 (they also pick cargo manifest categories, as 1-4 do), in multiplayer too.
- **Autopilot switches the tractor beam off.** In Classic it stays on.
- **Autonomous turrets:** a turret with no manual input slews toward the nearest hostile ship, leads it and fires. In Classic turrets are aimed by hand only.
  - The turret you are sitting in (`F5` / `F6`) shoots at your selected target instead (`T` picks along the turret's aim, `L` locks), as long as that target is hostile, within the turret's gun range and inside the arc the turret can swing to. Otherwise it goes for the nearest hostile. Moving the aim yourself always takes over.
  - **Turrets Hold Fire** in the GAME section of the settings overlay (off by default): your turrets hold fire until you lock a target with `L`, then engage only that target while it is hostile. Without a lock they stay where they point.
  - **Turrets Shoot Rocks** in the GAME section of the settings overlay (on by default): with no hostile around, your turrets shoot asteroids that are about to hit your ship. Not while Turrets Hold Fire is on.
  - **Turret Mining** in the GAME section of the settings overlay (off by default in Enhanced, on in Modern): parked in an asteroid field with Mining Eqpmnt aboard, idle turrets shoot any rock they can reach, and rocks your turrets burst can leave ore pods. See "Mining with turrets" under Asteroids.
  - **Piracy** in the GAME section of the settings overlay (off by default in Enhanced, on in Modern): you can demand a trader's cargo over the comm. See "Piracy" under Flight.

    - In multiplayer there are no autonomous turrets: a turret only moves when you aim it from its seat and only fires when you pull the trigger there.
    - Each turret has its own energy pool, so it cannot starve the forward guns. Classic uses one shared pool.
    - NPC turrets get a gunner too. In Classic they are fixed to the hull. **AI Turret Tracking** in the debug overlay (on by default) controls whether they turn toward their target at all: off, they stay fixed to the hull as in Classic and only fire when a target crosses their line, which makes fights easier. **AI Turret Lead** controls whether they lead. In multiplayer the NPC turrets always track.
- **Turret targeting uses the turret's aim:** `T` and auto-acquire in a turret view use the turret's own aim. Classic uses the ship's nose: from a turret, `T` and `L` still work, but only pick ships within 25 degrees ahead of your ship, not the ones the turret is looking at.
- **Other ships' missile and torpedo launches are audible** within 5000 units. In Classic they are silent.
- **AI Missile Locks** in the debug overlay (off by default): while it is on, a computer-flown ship's Heat Seeker only homes when fired from behind its target, and its Dumb Fire flies straight. Off, as in the original, their missiles home from any angle. It has no effect in Classic or in multiplayer.
- **Skipping a cutscene consumes the input:** the key or click that skips it does not also fire your guns. In Classic it also reaches the game, as in the original.
- **More cutscenes can be skipped:** the autopilot flyby and death cutscene can be skipped by any key or click. They cannot in Classic. The jump charge-up cannot be skipped in any mode.

Navigation

- **The Navigation Map opens while docked:** `N` opens it from any docked screen except the Mission Computer, where `N` types into the save name. Classic allows it only in flight.
- **Mission progress in flight:** when you complete a mission objective, the cockpit message line says so, for example **GO TO DELTA, NAV 2: COMPLETE**. See section 16. Classic shows nothing.
- **Right-click zooms out:** Quadrant View to quadrant overview, then quadrant overview to sectors overview. This needs **Open Universe**.
- **Arrow keys walk between quadrants** in Quadrant View. In the other map views they move the pointer. The numeric keypad and the D-pad move the pointer in every view.
- `Home` **returns to the current system** in Quadrant View.
- `Shift` + `N` **goes backwards** through the nav points, the same way the `NEXT` button goes forwards, wrapping from the first back to the last. In Quadrant View it steps back through the quadrants, and on the mission page back through your missions.
- `Shift` **+click a system** to show only its jump lines and list its destinations. A plain click shows every line again.
- **Hold** `Ctrl` to grey out systems your remaining jump fuel cannot reach, and `Ctrl` **+click** a system to see how many jumps away it is.
- **Plot a route:** select a system in Quadrant View and press `P`, or press `Ctrl` + `N` in flight or on the map for your mission's destination. After every jump the next jump point is selected for you. See section 15.
- **Hold** `Shift` to colour systems by faction.
- **Open Universe is on by default:** mod systems, jumps, bases and zones can become reachable. It is inactive in Classic. Built-in content currently adds Gemini in Humboldt, with the planet Gemini, the Heliopolis Array and Novaya Riga stations, two asteroid fields, random encounters and jumps to

Troy, Junction and Aquila. **Open Universe** and **Built-in Content** in the debug overlay control this content. See section 20.

Bases and screens

- **Escape backs out one level** on base screens. It does nothing in Classic.
- **Mission Computer from Repair & Upgrade** is allowed. Classic blocks it.
- **Mission Computer REJCT button:** the alternate keypad can remove a mission from your list, docked or in flight. In flight the mission's own ships leave at once; your other missions keep their progress. Classic has only the original keypad.
- **No unreachable mission offers:** without a jump drive, the First Sector Mission Machine and both guild boards offer only missions inside the current system. A board with nothing local to offer says so; a guild computer adds that a jump drive would open up missions in other systems. The Merchants' Guild, whose work is almost all cargo runs and bounties elsewhere, is often empty until you fit a jump drive. As on every board, the offers you see last until you next launch: missions you accept leave the board, and new offers appear only after your next landing. Classic offers missions anywhere, as the original does.
- **Harder missions pay more:** the pay offered rises with how tough the encounter will be, which grows with your credits, your missions flown and your ship (and is higher at the Mercenaries' Guild). A new pilot sees the same pay as in Classic; it is never less. Classic pays exactly what the original does.
- **Side missions survive unrelated landings:** they are consumed only when they actually complete and pay. Classic drops every carried side mission on any landing.
- **Notification center:** press **M** on a Launch Pad to read the results of your missions. It also opens by itself after a landing that completed or failed a mission. See section 16.
- **Cargo Manifest last page:** it lists its items instead of the original's empty last page when the hold has exactly 8, 16 or 24 item types.
- **Bar brush-offs are local and temporary:** only that bartender or patron is affected, until you leave. In Classic one brush-off silences every bar for the rest of the career.
- **Fixers in the bars:** now and then (about one landing in four) a stranger stands in the foreground of a base's bar, between the tables and the counter, with a job to offer. Point at the figure ("Talk to Stranger") and click to hear the offer, with where to go, who the enemy is and what it pays in the conversation itself. A stranger may offer an attack, a bounty, a cargo run, a sabotage job (attack the militia at a nav point in this system), a spy job (fly to a nav point in another system where the Confederation or the militia patrol, and touch it), or a smuggling run (only at bases that deal in contraband: take a load to a base elsewhere). A smuggling run's load sits in your hold as Mission Cargo, like any delivery job's, and is handed over when you land at the destination. At the prompt, point at the lower left of the picture until the box reads **Accept** and click (or press `A`); anywhere else reads **Reject** (or press `R`). An accepted job is added to your missions exactly like one from the Mission Computer (it counts toward the three-mission schedule and shows on the Mission Computer's MISSIONS page with its short summary), and the stranger leaves the bar. A rejected offer stays: you can talk to the stranger again until you launch. With a full schedule the stranger has nothing for you and the message line says so. At most one stranger is in a bar per landing; leaving and re-entering the bar, or saving and loading there, shows the same one. Without a jump drive the jobs stay in the current system, as on the boards. Fixers appear in multiplayer sessions too (the job then flies on the server, as a board's), but not at Open Universe bases or in a bar an interior mod replaces. Their dialogue is unvoiced, and it is in English in the Spanish edition too. A mission mod

can add strangers with jobs and lines of its own (see **Mission mods** in section 20). Untick **Fixers** in the GAME section of the settings overlay to turn them off. Classic has no fixers.
- **Entrance slides are quicker.** Classic uses the original 10 steps.
- **Clearer duplicate-purchase message:** buying a single-instance item you already have says **ALREADY OWNED**. Classic reuses the original **NO ROOM ON SHIP** message.
- **Higher ordnance capacity:** missile and torpedo capacity is 20 per matching launcher. Classic uses the original 10 per launcher.
- **Faster hardpoint-purchase cancel:** right-click cancels the Ship Dealer hardpoint picker. Classic keeps only the on-screen **Never mind** region.

Enhanced and Modern also permit multiple ships and hired wingmen when their settings are enabled. Use **Classic** for direct comparison with DOS Privateer, and **Enhanced** or **Modern** for OpenPrivateer's extra features.

The Experience setting can be changed while playing. In multiplayer it is fixed by the server.

5. Settings and Debug Overlay

During the Alpha, some settings still live in the debug/settings overlay rather than a finished front end.

Press:

```
~
```

This is normally the key to the left of `1` on an English keyboard. Its label depends on your keyboard layout.

In Classic, while flying, the key sits right above `Tab` (the afterburner), so there it opens the overlay only together with `Ctrl` (`Ctrl+~`); pressing it alone closes the overlay. Holding the key down does not open and close it again and again.

Choosing another overlay key

If your keyboard has no `~` key, or it does not reach the game (some on-screen and emulated keyboards), you can choose another key. With OpenPrivateer closed, add a line like this to `openprivateer.ini` :

```
DebugOverlayKey=Insert
```

That key then opens and closes the overlay instead of `~` , everywhere `~` would: in flight, on bases, in menus, the launcher, and while typing in one of the overlay's own text fields. The overlay's hint line names the key in use.

The value is a key name, optionally preceded by `Ctrl+` , `Alt+` and `Shift+` . Upper and lower case do not matter. Examples:

Value	Key
`Insert`	Insert

Value	Key
`Pause`	Pause
`\`	the backslash key
`Ctrl+F1`	F1 while Ctrl is held

The names are SDL's key names, the same ones the Controls screen writes to `openprivateer.ini` for keys you remap. Most keys are named as they are labelled (`F1`, `Home`, `Keypad 5`); a few punctuation keys are named by their character, such as `\` for backslash and `` ` `` for the key left of `1`.

With modifiers in the value, exactly those must be held: `Ctrl+F1` does not react to F1 alone or to Ctrl+Shift+F1. If the key also has a game action (for example `\`), it opens the overlay and the game action no longer reacts to it; the session log names that action at startup.

If the value is not a key name, `~` stays the overlay key and the session log says so at startup. An empty value also keeps `~`. Delete the line to go back to `~` for good. OpenPrivateer keeps the line as you wrote it and never adds it on its own.

The overlay contains settings for areas such as Experience mode, Privateer / Righteous Fire mode, audio, SoundFont, rendering, subtitles, Mission Computer style, controllers, Righteous Fire equipment in Privateer (All Equipment), multiple ships, hired wingmen, save file naming (Preserve Savegames, see section 7), the mod switches (Weapon, Item, Mission, Ship, Interior and Init Mods, Open Universe and Built-in Content, see section 20), Hull Atlas (section 18), the cockpit message line's position (Plaque Position, section 14) and update checking.

Some settings are disabled in Classic because they would change original gameplay.

6. Controls

OpenPrivateer supports keyboard, mouse, gamepads, joysticks and HOTAS-style controllers.

Open **Controls…** from the settings interface to inspect or remap controls. Controller bindings are stored per device. The same screen can also be reached by pressing `K` on the Options screen, opened in flight or on a base with `Alt+O`.

Assigning a controller

Selecting a controller and assigning it are separate steps:

1. Open **Controls…**, then **Remap Game Controls**.
2. Choose the physical controller from the controller drop-down at the top.
3. Click **Assign Controller**.

The drop-down only selects which connected device the Controller column shows and edits. **Assign Controller** makes that device active in OpenPrivateer. With several controllers connected, check the drop-down before changing a binding.

Assigning buttons, hats and axes

After assigning the controller:

1. Find the game action you want to change.
2. Click one of its slots in the **Controller** column.
3. Press the desired button, move the hat, or move the desired axis.

For example, to assign a joystick X axis to Yaw: select and assign the joystick, click the Controller slot on the **Yaw** axis row, then move the joystick's X axis.

Right-click a binding slot to clear it. `Esc` cancels capture without changing the slot. If the input you press is already used by another action that works at the same time (for example two flight actions), it is moved to the new action and the status line says where it was removed from. Actions that never work at the same time can share an input: a key can mean one thing in flight and another on a base screen or in the Mission Computer. Global keys such as fullscreen and quit work everywhere, so they share with nothing.

To put several actions on one input, tick **Allow one input on several actions** on the Settings page. A new binding then keeps the other actions on that input, and the status line says which ones ("Also used by: …"); inputs used by more than one action show in orange on the remap page. When you press a shared input, the action that can act right now gets it: with Jump and Autopilot on one button, the button jumps inside a jump point and engages the autopilot anywhere else. Actions that act while held (steering, guns, afterburner) all act together. Global keys and whole controller axes cannot be shared. Turning the option off keeps the shared bindings you already made; new bindings then take their input over again. A button or key you are still holding when it starts the autopilot, a jump, a landing or the self-destruct does nothing more until you let go of it.

Clear Controls clears the selected controller's bindings. **Remove** makes it inactive; it stays in the list and can be assigned again later.

OpenPrivateer can bind controller buttons, hats, axis directions and full analog axes. On Pitch, Yaw, Roll and Throttle, the round button next to the controller axis reverses it (filled = reversed).

Calibrating an axis or throttle

A full controller axis must be assigned before **Calibrate** becomes available. The complete order is therefore:

1. Select the controller in the drop-down.
2. Click **Assign Controller**.
3. Click the Controller slot on Pitch, Yaw, Roll or Throttle.
4. Move the physical axis to assign it.
5. Click **Calibrate** on that row.

The dialog shows the live raw value. Confirm with `Enter`, **Confirm**, or a tap of any button on that controller. To cancel, hold any button on that controller for about 1.5 seconds (a bar fills while you hold it), or press `Esc` or **Cancel**. Cancelling keeps the previous calibration.

- **Pitch, Yaw, Roll:** move through the full travel in both directions, let the control rest at centre, then confirm. OpenPrivateer records minimum, centre and maximum.
- **Throttle:** confirm three points: **zero**, **full military power** and the point where **afterburner** should begin. Choose **No afterburner zone** if the lever has no afterburner detent. The order establishes the lever direction, so a calibrated throttle normally does not need the reverse control.

Throttle axis modes

The Throttle row has a **Mode** button:

- **Auto** (default): Absolute for a joystick/throttle axis, Relative for a gamepad stick.
- **Absolute:** lever position directly sets speed. Moving past full military power into a calibrated afterburner zone engages afterburner.
- **Relative:** deflection from centre raises or lowers speed, like the original joystick throttle. On a calibrated throttle the calibrated **zero** point is the centre: moving towards full military power raises speed, moving the other way lowers it.

In Absolute mode the lever only takes over after you move it. Starting a flight does not immediately force the ship to the lever's physical position. Keyboard and mouse throttle controls still work; moving the lever again gives it control.

Calibration

OpenPrivateer also supports general joystick calibration. A raw joystick needs at least two axes and one button to qualify as a flight controller. SDL-recognized gamepads are accepted directly.

If several devices are attached, the calibration screen calibrates the **Primary Flight Controller** (see Controller settings). With **Automatic**, devices that cannot actually fly the ship are skipped.

The first time the calibration screen appears on its own (Options set to Joystick and the controller never calibrated), the Controls settings open on top of it. Pick your Primary Flight Controller there and press **Done**: the calibration screen is waiting underneath and calibrates the controller you chose. While the settings are open, the calibration screen ignores the controller.

On the calibration screen, tap a button to confirm each step. Hold a button for about 1.5 seconds to cancel; a bar fills while you hold it. `Esc` also cancels. Cancelling works at any step in every experience mode and keeps the previous calibration.

Controller settings

- **Dead Zone:** in Enhanced and Modern a round dead zone around the stick centre, adjustable from 3 to 30 percent (default 15), so a diagonal push is not cut off on one axis. Classic uses the original 30 percent dead zone on each axis separately.
- **Response Curve:** Linear or Smooth in Enhanced and Modern. Smooth gives finer control near the centre and full rate at full throw.
- **Mouse Sensitivity:** for mouse flight in Enhanced and Modern.
- **View Look Speed (F8/F9):** how fast the mouse turns the camera around your ship in the `F8` and `F9` views, 1 to 10 (default 5), in Enhanced and Modern. **Invert View Look (F8/F9)** turns the up and

down of that round. Both are also in the GAME section of the settings overlay, and are saved in `openprivateer.ini` as `ExternalViewLookSensitivity` and `ExternalViewLookInvert`.

- **Third-Person View (F10):** whether `F10` (the **Third-person view** control) works, default on, and the **Camera** beside it: **Near**, **Medium** (the default) or **Far** behind your ship, or **Rigid** (Medium with the camera fixed behind the ship, no swing in turns). Enhanced and Modern; also in the VIEW section of the settings overlay ("Third-Person View (F10)", "Third-Person Camera"), saved in `openprivateer.ini` as `ThirdPersonView` and `ThirdPersonCamera` (`Near`, `Medium`, `Far` or `Rigid`).
- **Input Backend** on Windows: WinMM or SDL DirectInput.
- **Primary Flight Controller:** which controller is "the joystick". It flies with the original stick scheme (trigger, button 2 roll, double-tap afterburner), moves the menu pointer, is the one the whole-joystick calibration screen calibrates, and in Classic is the Joystick input device. **Automatic** (the default) picks the first connected controller that can fly. The list updates when you plug a controller in or out; two identical controllers show as "(1)" and "(2)". The choice applies immediately. It does not change any button or axis binding: those stay with the controller they were assigned on, set with **Assign Controller**. If the chosen controller is not connected, the game uses Automatic's choice for now, keeps your choice for next time, and the page says which controller it is using. A newly chosen controller that has never been calibrated is not calibrated automatically; use the calibration screen for it.

Default gamepad layout

Input	Default action
Left stick Y	Pitch
Left stick X	Yaw
Right stick X	Roll
Right stick Y	Throttle
Right trigger	Fire guns
Right shoulder	Fire ordnance
Left trigger	Afterburner
Left shoulder	Cycle target
South button	Target lock
East button	Cycle ordnance
West button	Cycle guns
North button	Toggle ITTS
Start	Pause
Back	Navigation Map
D-pad up	Front view
D-pad down	Rear view
D-pad left	Left view

Input	Default action
D-pad right	Right view
R-stick click	Third-person view (Enhanced/Modern)
L-stick click	Open control wheel

These are normal remappable bindings. A joystick or HOTAS that SDL does not recognize as a gamepad gets the same four views on its first hat switch. If you have put something else on a D-pad direction, your binding stays and that view default is left off. The views are flight-only; on the menu screens the D-pad (and the first hat) moves the pointer in Enhanced and Modern.

Control wheel

The control wheel is a round menu of commands that a gamepad, a trackpad, a mouse or a touch screen player can reach without the keyboard. It works in every experience mode.

Opening it: click the left stick of a gamepad, press `F12`, click the mouse's middle button, or tap the touch handle (see "On a touch screen" below). The first three are the **Open control wheel** row of the **System** group in **Remap Game Controls**; a joystick that SDL does not recognize as a gamepad has no default, so assign a button to it there. The wheel does not open during cutscenes, the jump charge-up, the autopilot flyby or the death sequence.

Two ways to use it:

- **Tap** the opener: the wheel stays open. Point with the D-pad, the arrow keys, a stick or the mouse, or press `1` to `8` (`1` is the top, then clockwise). Press **A**, `Enter` or click to choose; **B**, `Esc`, `Backspace`, a right click or a click on the centre goes back one ring, and closes the wheel from the first ring.
- **Hold** the opener: push the right stick (or move the mouse) toward a sector and let go of the opener to choose it. Letting go with nothing pointed at closes the wheel; letting go on a group opens that group's ring, which then stays open.

The first ring holds groups of commands for the screen you are on. In flight: Navigation (top), Comms, Wingmen, Ship, System (bottom), Views, MFD, Weapons. On a base: Back, Next region, Mission computer, Nav map, System, Previous region, Right click and Services (the fleet selector, notifications and ships for sale, where available); while one of those windows is open, the wheel offers **Close window**. The Navigation Map, the Mission Computer and conversations have rings of their own; the title and Options screens show the System ring (Options, Pause, Settings overlay, Fullscreen, Quit to title, Quit game).

Comms (in flight) works the Communications MFD for you. Its ring shows the same rows the MFD shows, in the same order: first who you can call (your wingmen, the ships and bases in range, a fuel service when one is offered), and once you pick one, what you can say to them. The `1` to `8` keys pick the same rows as the MFD's own number keys. Picking someone keeps the wheel open and shows their lines in the ring, with their name in the centre; picking a line, an order or a hail sends it and closes the wheel. The centre (or **B**, `Esc`, a right click) is the MFD's "Never mind": it closes the Communications MFD and takes you back to the first ring. If the Communications MFD is not up yet, choosing Comms brings it up and the wheel comes back on the Comms ring a moment later. Comms is dimmed ("No one to call") when there is nobody to talk to, and missing from a turret view. If a page has more than eight

rows, the last sector reads **More** and shows the next ones; the shoulder buttons turn these pages too. A row too long for its sector ends in "…"; point at it to read it whole under the wheel.

Ship (in flight) holds Jettison cargo, Shields up, Shields down and Self-destruct.

Weapons (in flight) holds Cycle guns, Cycle ordnance, Previous ordnance, Countermeasures, **Gun groups**, Lock target, Next target and ITTS (Previous ordnance and Countermeasures in Enhanced and Modern only). **Gun groups** opens one more ring with the gun group keys: Gun group 1 at the top, then clockwise to Gun group 8, so `1` to `8` on that ring are the same keys as in flight. Classic has groups 1 to 4. A group your ship does not have is dimmed ("No such gun group"). Going back from Gun groups returns to the Weapons ring.

MFD (in flight) lists the pages your cockpit can show from the view you are in; the pages on show now are outlined in amber and marked "On show" under the wheel. Choosing a page puts it up as its key would. On the MFD ring, the shoulder buttons cycle the left and right MFD.

Commands that the current mode, a setting or a multiplayer session never allows are left out. Commands that cannot act right now are shown dimmed; point at one to read why under the wheel (for example "Not in a jump point"). The line under the wheel names the command you point at and where you are.

Dangerous commands (Self-destruct, Quit to title, Quit game, Delete save) ask first: a single **Confirm** sector appears opposite. Hold **A** on it for a second, press `Enter` on it, click it or tap it. Anything else cancels.

On a touch screen: once you touch the screen, a small round **handle** with a wheel on it appears in the bottom-left corner (in the black bar beside the picture when there is one). Tap it to open the wheel, then tap a sector to choose it, or tap the centre to go back (from the first ring this closes the wheel). You can also put your finger on the handle, slide it onto a sector and lift it there: the sector under your finger lights up while you slide, and lifting chooses it. Lifting outside the wheel does nothing; lifting outside the game's window, or a touch the system cancels, closes the wheel without choosing anything. Only the finger that opened the wheel (or the first one you put on it) works it, so a second finger does not change your choice. A tap anywhere else on the screen works as a mouse click, as before. The handle follows the window when you resize or rotate it. If your touch screen reaches the game as a mouse instead (some systems and controller setups deliver it that way), set **Touch Handle** to **Always** and tap or click the handle; on that screen the handle does not appear by itself, and it does not react while you fly with the mouse.

Time: in single player the game stops while the wheel is open (the line under the wheel says "Game paused"). In a multiplayer session the game goes on: the wheel takes the right stick, the D-pad, A, B, the shoulder buttons, the arrow keys, `Enter`, `1` to `8`, `Esc`, `Backspace` and the mouse, while the left stick, the triggers and your other keys keep flying the ship.

Settings (Settings, **Control Wheel**): **Wheel Size** (a slider, 50% to 100% of the window's shorter side, 80% by default; the wheel never gets so small that its labels or touch targets become too small, and never larger than the screen), **Wheel Transparency** (a slider, 0% to 80%, 0% by default: how much of the game shows through the wheel; its labels, outlines and the highlight stay clearly visible), **Pause While Open** (single player, on by default), **Rumble on Highlight** (a short gamepad rumble when the pointed sector changes, off by default), **Touch Handle** (Auto: shown once you have touched the screen; Always; Off) and **Handle Side** (the bottom-left or bottom-right corner). **Wheel Size** and **Wheel**

Transparency are also sliders in the debug overlay's **COMMAND WHEEL** section; both places change the same setting, and a change shows at once on an open wheel. In `openprivateer.ini` they are `ControlWheelSize=80` (50 to 100; `Small`, `Normal` and `Large` are also understood, as 65, 80 and 95) and `ControlWheelTransparency=0` (0 to 80).

The original joystick scheme

A joystick or pad that SDL does not recognize as a gamepad flies with the original game's stick scheme, in every experience mode:

- its first two axes (X and Y) turn and pitch the ship; on many simple USB pads these two axes are the D-pad;
- **Button 1** (the trigger) fires guns;
- holding **Button 2** turns the stick's X into roll and its Y into throttle;
- Button 1 with Button 2 held fires ordnance, and tapping Button 2 twice and holding it lights the afterburner.

Your own bindings come first: if you assign Button 1 or Button 2 to any flight action in **Remap Game Controls**, that button does only what you assigned. With Fire guns on Button 2, holding it fires the guns and the stick keeps turning and pitching the ship; with Jump on Button 1, it jumps without firing. In Classic, a button you have not assigned keeps its original role.

In Enhanced and Modern, on a joystick you have set up in **Remap Game Controls** (any button assigned to anything, a hat direction changed, or **Clear Controls** used), Buttons 1 and 2 do only what you bind them to: left unassigned, they do nothing in flight. An unconfigured stick, or one where you have only assigned axes such as a twist for **Roll** or a lever for **Throttle**, keeps the original buttons. **Restore Defaults** brings them back.

On the menu screens Button 1 clicks and Button 2 right-clicks; see **Joystick pointer** below. These two menu bindings do not count as setting the stick up, and assigning Button 1 to a flight action as well keeps it clicking on the menus.

To steer with a hat or D-pad direction, a button or one direction of an axis, assign it to **Pitch nose up**, **Pitch nose down**, **Yaw left**, **Yaw right**, **Roll counterclockwise** or **Roll clockwise**; this works with Joystick selected on the Options screen in every mode. To roll or change speed without Button 2, assign a twist axis to **Roll** or a lever to **Throttle**, or put **Roll counterclockwise**, **Roll clockwise**, **Throttle up** and **Throttle down** on buttons.

Joystick pointer

A calibrated joystick can move the pointer on base and menu screens, matching a feature of the original game. SDL gamepads click with the A/South button.

A joystick that SDL does not recognize as a gamepad clicks with **Button 1** (the trigger). In Enhanced and Modern these clicks are the **Pointer click** and **Pointer right click** rows of the **Pointer** group in **Remap Game Controls**: by default Button 1 clicks and Button 2 right-clicks. Assign either row to another button to click with that button instead, or right-click its slot to clear it; a cleared click leaves the joystick without one (the stick and hat still move the pointer, and the keyboard and mouse still click). **Restore**

Defaults puts both back on Buttons 1 and 2. In Classic, Button 1 always clicks and there is no right click from the joystick.

Keyboard pointer

As in the original game, the arrow keys move the pointer on every screen that shows it except the Mission Computer, and `Enter` clicks. The pointer moves in small steps and speeds up while you hold the key. Holding `Enter` while moving drags. This works in every experience mode.

In Enhanced and Modern the keys are remappable (the **Pointer** group on the remap page), and these also work:

Input	Action
Numeric keypad 8, 2, 4, 6	Move the pointer
Numeric keypad 5	Click
Numeric keypad 0	Right click
D-pad / first hat	Move the pointer
Joystick Buttons 1 and 2	Click and right click

The pointer also works at a conversation's Accept/Reject prompt there. In the Navigation Map's Quadrant View the arrow keys walk between quadrants instead; use the numeric keypad or the D-pad to move the pointer there.

Chat keys

In a multiplayer session, in flight and in docked scenes:

Input	Action
`Y`	Chat to everyone on the server
`U`	Chat to the pilots in your star system
(no key)	Chat: reply to the last whisper

All three are in the "Chat (multiplayer sessions)" group of Remap Controls and have no controller button until you bind one. The chat box and its commands are described in "Chat" in section 22.

7. New Game, Load and Save

The title screen has **NEW**, **LOAD**, **OPTIONS** and **QUIT**. In Enhanced and Modern it also shows **Multiplayer** at the top right, which opens the list of servers (see "The Servers page in the game" in section 22); Classic's title screen is the original's.

NEW

NEW starts from the original data in `INITCFG.PAK` or `INITRF.PAK`. If the matching file is missing, NEW is greyed out.

In Enhanced and Modern a start mod can change where and how a new game starts (experimental, see "Start mods" in section 20).

NEW asks for your name and callsign. Press `Esc` on that screen to cancel and go back to the title menu. Righteous Fire's "Load an old Privateer ship?" question has to be answered with `Y` or `N`.

LOAD

Privateer saves use `.SAV`. Righteous Fire saves use `.PRS`.

Put existing DOS saves in the `savegame/` folder of the user-data folder (the launcher's **Savegames** button opens it):

```
savegame/
```

OpenPrivateer reads the original save format.

A damaged save (cut short, or with a broken or missing part, or a file that is not a save at all) is still listed under its name, but choosing it shows "Save game is damaged." and it is not loaded; the log says what is wrong with it. You can delete it from the browser.

SAVE

Use the Mission Computer and choose **SAVE**.

By default, saving under a typed name never overwrites a file. Matching the original game, the name always gets `_NEW` appended (then `_NEW2`, `_NEW3`, ...) rather than reusing the plain name. Typing `PILOT` creates `PILOT_NEW.SAV`, not `PILOT.SAV`; saving under that same name again creates `PILOT_NEW2.SAV`, and so on.

If you would rather keep one file per name, uncheck **Preserve Savegames** in the debug overlay's Experimental section (or set `PreserveSavegames=0` in `openprivateer.ini`). Saving then writes exactly the name you typed: typing `PILOT` writes `PILOT.SAV` and replaces an existing `PILOT.SAV`, together with its `.openpriv.json` sidecar. Picking a save in the list and saving again overwrites that save. If a save can not be written, the old file stays as it was. The setting is remembered between runs.

Some OpenPrivateer-only state cannot fit inside the original DOS save format. That state is stored in a matching `.openpriv.json` sidecar while the `.SAV` or `.PRS` remains in the original format.

If you use parked ships, the notification center's messages or other OpenPrivateer-only persistent features, keep the sidecar together with its save.

A sidecar written by a newer OpenPrivateer build is ignored by an older one, which loads the plain save: the ship flies as the stock ship the save records, and parked ships and the other OpenPrivateer-only state are missing in that build. The files themselves stay as they are, so the newer build still loads everything, unless you save under the same name in the older build with **Preserve Savegames** off, which replaces the sidecar.

8. Mission Computer

The Quine 4000 handles saves and pilot information.

Its important pages are:

- **MISSIONS**: active plot mission and accepted side missions
- **FIN**: cash, kills and faction standing
- **MAN**: cargo manifest
- **SAVE / LOAD**: save browser

A pilot can carry up to three side missions. The active plot mission is separate and does not consume one of those slots.

Classic uses the original Mission Computer presentation. Enhanced and Modern can use the alternate OpenPrivateer presentation through the **Quine Model** setting.

Save browser

The SAVE and LOAD browsers list saves alphabetically in every mode.

The name box takes letters, digits, `-` and `_`, up to 8 characters. In LOAD you can keep typing past 8 as long as the name is the start of a save in the list, so a save such as `PILOT_NEW2` can be typed in full and loaded with `Enter`, as well as picked from the list.

In Enhanced and Modern the save whose name is in the name field (picked from the list or typed) is highlighted, and these extra controls work:

- mouse wheel over the list: scroll it one save per notch
- `Page Up` / `Page Down`: move one page (10 saves) through the list
- `Home` / `End`: jump to the first or last page
- `Delete`: delete the highlighted save

`Delete` opens a confirmation showing the file, pilot, location, ship, credits and when it was saved (for a damaged save: what is wrong with it). Click the red **DELETE**, or press `Enter` or `Y`, to delete it; click **CANCEL**, or press `Esc` or `N`, to keep it. A controller can click either button with the pointer. Deleting removes the save and its `.openpriv.json` sidecar for good: it does not go to the recycle bin. A read-only save is not deleted; the window says why.

`Delete` also works in the browser opened by the title screen's LOAD, but not when Righteous Fire is importing a Privateer pilot. All five keys can be changed in the controls settings, in the Mission Computer group. In Classic the browser has only the original controls: the `[` and `]` keys (or the arrow keys) and the keypad's arrow buttons.

9. Bases

Bases use the original game's scenes, people and layout.

Move the pointer over doors, people and terminals to see available actions, then click normally. On several launch pads (New Detroit's, Perry's and those of the pirate, pleasure and refinery bases) the bottom edge of the screen leads into the base as well as the door.

You can also use the keyboard: `Tab` moves the pointer to the next place you can click, showing its name, and `Enter` clicks it. In Classic, as in the original game, the first `Tab` after you enter a scene skips to the second place and `Shift+Tab` works like `Tab`. In Enhanced and Modern the first `Tab` goes to the first place and `Shift+Tab` goes back; both keys can be changed in the controls settings (Base group).

Some places play a short animation before you go on (the train at Oxford, the elevator at New Constantinople, Admiral Terrell). Press `Enter` to skip it; clicking with `Enter` skips it from the start.

Depending on the base you may find the Launch Pad, Main Concourse, Bar, Commodity Exchange, Ship Dealer, Repair & Upgrade, Mission Computer, Merchants' Guild or Mercenaries' Guild.

Not every base contains every service.

Your pilot's portrait

In Enhanced and Modern your pilot can show a face of your own in the base conversations, in place of the stock pilot's face, on every line your pilot speaks (stock conversations and those of mods alike). The intro films and the cockpit's comm display stay as they are.

1. Put a picture in the `portraits/` folder of your user-data folder (see "Where your files live"): a PNG or a JPEG.
2. On a Launch Pad press `S` and click **PORTRAIT** at the top of the Fleet Selector.
3. Pick the picture in the list. The preview shows it as a conversation does, over this base's bar, talking and blinking (see "Making your portrait talk" below).
4. Click **USE**.

STOCK FACE goes back to the stock face, **OPEN FOLDER** opens the `portraits/` folder, **REFRESH** reads it again after you added a file, and **CANCEL** or `Escape` leaves everything as it was.

How your picture is fitted:

- It is scaled to the stock pilot's height and never stretched. It stands at the bottom of the picture, under the stock pilot's head.
- A picture wider than the stock figure (about 2.3 times as wide as it is high) has its sides cropped evenly, so keep the face in the middle.
- A picture of at least 248 pixels in height is shown in full colour at up to four times the game's resolution. A smaller one is matched into the scene's colours, like the game's own art, and its colours can shift a little from bar to bar.
- Pixel art smaller than 124 pixels in height is enlarged by whole steps, so its pixels stay square.
- A PNG with a transparent background shows the bar behind your pilot, as the stock face does. A photo or a JPEG is drawn as a rectangle.
- A file may be up to 8 MiB, up to 4096 pixels on a side and up to 2048 x 2048 pixels in all. A file the game cannot use is listed with the reason, and USE stays off for it.

Making your portrait talk

Your portrait moves its mouth while your pilot talks and blinks now and then. The game makes ten mouth shapes from your picture, the same kinds the stock pilot's face shows (closed, a small opening, wide with the teeth showing, small and large rounded ones, a slit, a dark line between the lips), and shows them letter by letter as a stock face does. You do not have to draw anything:

1. Pick your picture in the picker and leave **ANIMATE** on.
2. On the preview, click the middle of the mouth and right-click one eye (a right-click moves the eye mark nearer to it). The marks show as small crosses: magenta for the mouth, cyan for the eyes. Click again to move them. To place them precisely, zoom in: turn the mouse wheel over the preview to zoom in or out around the pointer (up to 8 times). Drag with the middle mouse button, or hold Space and drag with the left button, to move the zoomed picture. Press Home, or zoom all the way out, to see the whole picture again.
3. Set **Mouth opening**: further right opens the mouth wider. Set **Mouth width** under it: further left keeps the talking mouth narrower than your lips (the corners stay closed, and the rounded shapes look more like an "o"), further right stretches the lips wider while the mouth talks; the middle is the usual width. Only the lips, the skin right around them and, a little on the more open mouths, the chin move; the nose, the jaw's sides and the neck stay still.
4. Watch the preview, which talks a sample line the way a conversation does, then click **USE**.

A picture with the mouth closed and the face seen from the front works best; a face in profile, an open smile or a hand at the mouth do not. The marks start where a passport-style head-and-shoulders photo has the eyes and the mouth, so check them on any other picture. **RESET MARKS** puts them back there and sets the mouth opening and width to the middle again. Turn ANIMATE off for a portrait that stands still. Your marks are kept in `me.face.json` beside `me.png`, so the picture keeps them the next time you pick it.

You can also draw the frames yourself, the mouth and the blink each on its own (a picture with its own mouth pictures may still get a blink made by the game, and the other way round). Beside `me.png`:

File	What it is
`me.talk1.png` … `me.talk10.png`	mouth pictures, numbered 1, 2, 3
	without a gap (the list ends at the
	first missing number); ten is best
`me.blink.png`	the eyes closed

- Each is a PNG of exactly `me.png`'s pixel size and is drawn over it: paint only the mouth (or the eyelids) and leave the rest transparent. On a picture under 248 pixels in height, a pixel is either fully see-through or fully painted.
- While your pilot talks, every 150 ms the next letter of the line shows one mouth picture, the way a stock face picks its own ten mouth pictures; the letters do not follow the sounds. With ten pictures each one stands for the stock face's picture of the same letters:

Picture	Letters	The stock face shows
`talk1`	b l v	the lips closed and pressed

Picture	Letters	The stock face shows
`talk2`	c m w	a small opening
`talk3`	d n x	wide, the upper teeth showing
`talk4`	e o y	small and rounded
`talk5`	f p z	rounded, a little larger
`talk6`	g q	a round “o”, the widest open
`talk7`	h r	a wide slit
`talk8`	i s	closed, a dark line between the lips
`talk9`	j t	a medium opening
`talk10`	a k u	parted, a line of teeth

With fewer pictures the letters cycle through the ones you have.

- Spaces, digits and punctuation show `me.png` itself, and so do the letter pairs “er”, “ch” and “th” and a word’s last “e”, where a stock face keeps its mouth closed too, so `me.png` should have the mouth closed.
- A line with a recorded voice opens the mouth wider when it is louder: with ten pictures `talk6` for the loudest, then `talk9`, `talk4`, `talk2` and `talk7`; with five, `talk5` widest, then `talk4`, `talk1`, and `talk2` barely open (`talk3` shows for the letters only).
- `me.blink.png` shows for 150 ms now and then.
- These files are not listed on their own in the picker.

EXPORT FRAMES writes the frames of the selected picture, made by the game or your own, into the `portraits/` folder as `me-talking.png` with its mouth and blink files, and selects it. Open them in a paint program, touch them up, click **REFRESH** and pick `me-talking.png`. An export never overwrites a file: a second one is `me2-talking.png`.

The `portraits/examples/` folder holds a drawn pilot with ten mouth pictures, one for each row of the table above, and a blink. Copy its `example.*` files up into `portraits/` to try them. Its `sheet.png` shows them: the top row the pilot and the pilot with each mouth (1 to 10) and the blink (B) as the game draws them, the bottom row each file alone, transparent but for the mouth or the lids.

The portrait is saved with your pilot. When you click USE, a fitted copy goes into `portraits/pilots/`, and your saves use that copy: keep the folder with your saves when you move them to another computer. A save whose copy is missing shows the stock face, and the notification center tells you to choose the portrait again. Your pictures stay on your computer: in a multiplayer session your portrait shows only in your own conversations. Classic always shows the stock face.

Renaming your pilot

In Enhanced and Modern you can change your pilot’s name and callsign after NEW. The top row of the portrait picker shows them, for example `Pilot: Markus "Eagle"`.

1. On a Launch Pad press `S` and click **PORTRAIT** at the top of the Fleet Selector.
2. Click **RENAME PILOT**. The Rename Pilot box opens with your name and callsign in it; you type into the name.
3. Type the new name. `Tab` moves to the callsign and back.
4. Press `Enter` or click **OK**. **CANCEL** or `Escape` keeps both as they were.

The rules are those of the NEW screen: letters and spaces only, a name of up to 17 and a callsign of up to 14 characters, neither left empty. Spaces at the start and the end are dropped. A callsign one of your hired wingmen already flies under is refused. When something is refused, the box says why and keeps what you typed.

The conversations use the new name and callsign at once. Your saves keep their file names; the next save writes the new name and callsign into it. In a multiplayer session the other pilots still know you by the callsign you joined with, and the Rename Pilot box says which one. Classic has no RENAME PILOT.

10. Commodity Exchange

The Commodity Exchange buys and sells cargo. Left-click “Select Items” to buy or sell one unit. Right-click it to buy as many as you can afford and carry, or to sell everything you hold of that commodity, in every experience mode.

Prices and availability depend on the base economy. Each time you land at a base, or load a game, its prices and stock are set anew: what you bought out or sold there is not waiting for you on your next visit, while within one stay the Exchange keeps the prices and the stock your trades left. Cargo consumes hold space. Mission cargo always comes aboard in full when you accept a mission, even when that fills the hold past its capacity; while the hold is over capacity you cannot buy more cargo until you sell or deliver some.

The secret compartment, once obtained through the story, has separate hidden capacity of 20 units, one of them taken by the Ancient Artifact. Tobacco, Ultimate, Brilliance and Slaves you buy, or pick up with the tractor beam, fill the compartment first; whatever does not fit goes into the normal hold. Patrol searches find contraband only in the normal hold. The Exchange sells compartment cargo too: selling one unit takes it from the normal hold first. With commodity mods (see “Mods”), the goods a mod makes contraband are hidden and searched the same way, and the ones it makes legal are not.

With hired wingmen (Enhanced and Modern), the Exchange can also trade into a wingman’s own hold; see “Your wingmen’s cargo” in section 13.

11. Ship Dealer and Repair & Upgrade

With normal single-ship play, buying another hull follows the original trade-in model. The current hull and sellable equipment contribute to the trade-in value.

Repair & Upgrade handles ship equipment.

Ships for sale (Enhanced and Modern): click the salesman standing in the middle of the Ship Dealer’s showroom, or press `B` there, to see every ship this dealer sells: the Orion, the Centurion and the Galaxy first, then, with a ship mod that brings ships you fly installed and **Ship Mods** on (in multiplayer the

server's ship mods, whatever **Ship Mods** says), the mod ships this dealer sells. Each row shows the ship's picture, its name, for a mod ship the stock ship it is registered as, and its price. **BUY** buys it the same way as clicking a ship in the showroom: your ship is traded in (or parked, with multiple ships), the salesman makes his offer and you accept or reject it. The ship you fly is marked YOURS. `B`, `Esc` or CLOSE closes the list. A mod ship may trade in for less than its price, as its mod sets it; the salesman's offer shows what it is worth.

A mod ship may come with equipment: its row says what ("Comes with …") and its price includes it. You get it fitted when you buy the ship, onto the empty places of the new ship: a jump drive, scanner or Steltek gun you keep from your old ship stays, and equipment of the mod ship's that one of them leaves no room for comes off the price instead. Because the price includes the equipment, trading the ship in or selling it parked pays for the hull less what that equipment is worth, and for the equipment on it one item at a time, as on any ship.

In Repair & Upgrade a mod ship shows its own picture where the mod brings one, and you fit guns and launchers on its own positions.

A mod ship's hold can be bigger than the stock ship it is registered as (a 1000-ton freighter registered as a Galaxy, for one). Your save keeps all the cargo aboard. If you load that save without the mod (or in the original game), you fly the stock ship with all that cargo still in its hold: you can sell it, but you cannot take on more until it is below the stock ship's hold. A hold that big can be filled with a single commodity: one kind of cargo takes up to 65535 units, in the hold and again in a secret compartment.

Buy mode lists only what your hull can take, as in the original: each hull has its own shield and engine ceiling and its own turrets, and the Centurion cannot fit the Cargo Expansion. Sell and Repair list whatever the ship carries.

Missiles need a missile launcher and torpedoes a torpedo launcher, and each launcher carries a limited number of rounds (see section 4). Selling a launcher also sells the rounds the launchers you keep cannot carry, each paid at its normal sell price on top of the launcher's: your last missile launcher takes all your missiles, your last torpedo launcher all your torpedoes, and selling one of two launchers sells only the rounds above what the other one holds. When missiles are sold this way, the cheapest go first. A tractor beam does not fire ordnance.

In Classic, entering Repair & Upgrade always starts in **Buy** mode on the first item, matching the original. Enhanced and Modern remember the previous mode and item where possible.

All Equipment (Privateer, Enhanced and Modern, with `RF.TRE` in the game data folder): the dealer also sells Righteous Fire's equipment, with Righteous Fire's prices and pictures: the Fusion Cannon, Gun Cooler, Shield Regenerator, Speed Enhancer, Thrust Enhancer, Isometal Armor, Advanced Droid, and Shield Generator and Engine Upgrade 6 and 7. Shield Generator 6-7 and Engine Upgrade 6-7 fit the Orion only; everything else fits every hull. Without `RF.TRE` the box is greyed out and its hover text says so. In Classic it is greyed out and the dealer is the original's. Weapon mods can change the Fusion Cannon and add guns based on it, as for any other gun.

Equipment you bought this way stays on your ship and keeps working when you turn the setting off or play Classic: Sell and Repair still list it. A Privateer save with this equipment is meant for OpenPrivateer: keep its `.openpriv.json` file next to it. That file tells the Fusion Cannon apart from the Steltek gun, which the save itself stores the same way; without it, and in the original game, a Fusion Cannon loads as a Steltek gun.

A damaged scanner can be repaired in the Software Booth's Repair Mode. The cost depends on the scanner and how badly it is damaged, as in the original game.

In the Software Booth's Sell Mode, Previous and Next step through what you own: your scanner first, then each quadrant map, and the Gemini Sector Map while you own all four. Select sells the item shown. A quadrant map sells for 1000 credits and the Gemini Sector Map for 4000 (it takes all four quadrant maps with it); the Navigation Map then shows "No information available." for that quadrant's systems until you buy the map again.

With hired wingmen (Enhanced and Modern), Repair & Upgrade can also work on a wingman's ship parked at the base; see "Servicing your wingmen's ships" in section 13.

12. Multiple Ships and Fleet Selector

Multiple ships are an OpenPrivateer feature.

To use them:

1. Switch to Enhanced or Modern.
2. Enable **Allow Multiple Ships**.
3. Buy another hull.

Instead of trading in the old ship, OpenPrivateer parks it at the current base and makes the new hull active. The parked ship keeps all its equipment. The new hull comes without equipment except a basic scanner (an Iris Mk. I), so it can target from its first launch; anything else, including a better scanner, is bought as usual. With a trade-in (one ship per pilot) your old scanner moves to the new ship instead.

On a Launch Pad press:

```
S
```

to open the Fleet Selector.

The active ship is marked as current. Parked ships can be selected or sold. Remote selection of a ship parked at another base is supported.

Multiple ships are not active in Classic.

Naming your ships

In Enhanced and Modern you can give each of your ships a name of its own. Press `S` on a Launch Pad (without Allow Multiple Ships the Fleet Selector then lists just the ship you fly) and click **RENAME** on the ship. Type the name and press `Enter` or click **NAME**; `Escape` or **CANCEL** leaves the name as it was. Clear the box and confirm to remove a name.

A name is up to 24 characters: letters, digits, spaces and the usual punctuation the game's font can draw. If a character cannot be drawn, or the name is too long, the Fleet Selector says so and keeps the old name.

The name stays with the ship: selecting another ship, swapping ships with a wingman or saving and loading keeps each ship's name, while a ship you sell or lose takes its name along. A newly bought hull has no name until you give it one.

Your ship's name, followed by its type, appears:

- in the Fleet Selector, for example "Rosinante (Tarsus)", and in its confirmations;
- on Repair & Upgrade's ship line at the top of the screen ("Ship: Rosinante (Tarsus)", or just the name when the type does not fit);
- on the Mission Computer's FIN page, below Cash;
- in the cockpitless HUD, above the left display (below it in the third-person view).

Names are not shown in Classic, and other pilots in a multiplayer session do not see them.

Service plan

In Enhanced and Modern you can have your ships repaired and re-armed automatically every time you land. Press `S` on a Launch Pad and click **SERVICE PLAN** on the row of the ship you fly, then pick a plan:

- **Repair**: 50 credits per landing.
- **Re-arm**: 50 credits per landing.
- **Repair and re-arm**: 100 credits per landing.
- **No plan**: ends the plan.

The plan's fee comes on top of the normal 50-credit landing fee and on top of the dealer's own prices: every repair costs what Repair & Upgrade charges for it, and every missile, torpedo or round costs what the dealer sells it for.

When you land with a plan:

- **Repair** works at every base, also one without a ship dealer, and repairs everything Repair & Upgrade would list.
- **Re-arm** works only at a base whose ship dealer sells equipment. It buys your missiles, torpedoes and gun ammunition back up to what you launched with. Ammunition this base's dealer does not sell is not refilled.
- The plan covers the ship you fly and, with hired wingmen, your wingmen's ships parked at that base. Wingmen who flew with you are back at the base you land at, so their ships are covered too.
- Repair is done first, then the re-arm. If your credits do not cover a part (its fee and everything it would buy for all your ships), that part is skipped entirely and nothing is charged for it. A notification tells you what was due, how many credits you had and what was skipped. In a multiplayer session the same message shows on screen as you arrive.
- A part with nothing to do costs nothing, its fee included.

The plan is saved with your pilot. In a multiplayer session the Fleet Selector shows just the ship you fly, with **SERVICE PLAN**. Classic has no service plan.

The **PORTRAIT** button at the top of the Fleet Selector chooses your pilot's portrait for the base conversations (see "Your pilot's portrait" in section 9); the line beside it names the portrait you use.

13. Hired Wingmen

Hired wingmen are an OpenPrivateer feature built on top of multiple-ship ownership.

Enable **Allow Multiple Ships** and **Allow Hired Wingmen** in Enhanced or Modern mode.

A hired pilot flies one of your actual owned parked ships. If the wingman survives, the ship returns with its live equipment state. If it is destroyed, the ship is lost.

Pilots have persistent callsigns and skill. Confirmed kills can improve their skill. Launching with a wingman costs a sortie fee.

A ship assigned to a hired pilot cannot simply be selected or sold as an ordinary parked ship.

Swapping ships with a wingman

In the Fleet Selector, a ship a pilot flies shows **SWAP** instead of SELECT. After you confirm, you fly the pilot's ship and the pilot takes over the ship you were flying. The pilot keeps their callsign, skill and kills. Each ship keeps its own equipment and damage. Your cargo, credits and missions stay with you. Swapping is free.

You can only swap with a ship parked at the base you are at. The swap is refused if your cargo does not fit the other ship's hold, or if you carry hidden cargo and that ship has no secret compartment.

To repair or upgrade a wingman's ship you can also swap, use the Ship Dealer or Repair & Upgrade as usual, then swap back, or service it directly (see "Servicing your wingmen's ships" below). At your next launch the pilot takes off in whichever ship they now fly.

If your wingman carries cargo (see below), it stays with the pilot too: after a swap it is aboard the ship they now fly. The swap is refused if their cargo does not fit the ship you are leaving.

Your wingmen's cargo

A wingman's ship has a hold of its own, which you can fill at the Commodity Exchange of the base where that ship is parked:

1. Land where your wingman's ship is parked and open the Commodity Exchange.
2. Below the trading computer a line names the hold you are trading into, for example "Hold: Galaxy 12/150" for your own ship or "Hold: Rook, Galaxy 0/150" for your wingman Rook's. Click it to switch to the next hold. Each wingman whose ship is parked there has their own.
3. Buy and sell as usual. You pay with your own credits and get the money from a sale. The wingman's ship can take as much as its own hold holds.

Each visit to the Exchange starts with your own hold.

- **No contraband.** A wingman will not carry Tobacco, Ultimate, Brilliance or Slaves (or a good a commodity mod makes contraband): buying them into a wingman's hold shows "CONTRABAND REFUSED". A wingman's ship has no secret compartment.
- **No mission cargo.** Mission cargo always goes into your own hold.

- **The cargo stays with the wingman** when they fly with you, land, or when you save and load. It is shown in the Fleet Selector next to the pilot ("Cargo 30").
- **Dismissing.** You cannot dismiss a wingman who still has cargo aboard. Sell it at the Exchange first.
- **Losing a wingman.** If the wingman's ship is destroyed, the cargo aboard is lost with it.

Servicing your wingmen's ships

You can repair and equip a wingman's ship at Repair & Upgrade without swapping ships:

1. Land where your wingman's ship is parked and open Repair & Upgrade.
2. Along the top of the screen a line names the ship you are working on, for example "Ship: Galaxy" for your own ship or "Ship: Rook, Galaxy" for your wingman Rook's. Click it to switch to the next ship. Each wingman whose ship is parked there can be chosen.
3. Buy, sell and repair as usual. The picture shows the chosen ship with its equipment, and everything you buy is fitted to that ship. You pay with your own credits and get the money from a sale. Your own ship is not touched.

Each visit to Repair & Upgrade starts with your own ship. The same rules apply as for your own ship: what the hull can take, launchers and their ordnance, turrets and mod equipment. At your next launch the wingman takes off with the new equipment.

You cannot sell the Cargo Expansion of a wingman's ship while it holds more cargo than the ship could carry without it: the dealer shows "CARGO ABOARD". Sell some of that cargo at the Commodity Exchange first. Ships themselves are bought and sold only for the ship you fly, and the Software Booth always works on your own ship.

Giving orders to your wingmen

In flight you can give your wingmen orders with the keys Wing Commander 2 used:

Key	Order	What happens
`Alt+A`	Attack my target	They attack the ship you have targeted until it is destroyed, then go back to what they were doing.
`Alt+H`	Help me out	They attack the nearest enemy that is after you. If nobody is, they form up on you.
`Alt+B`	Break and attack	They engage any enemy, however far away.
`Alt+F`	Form on my wing	They take up a V formation behind you, one on each side, and hold it. They ignore enemies and do not fire.
`Alt+R`	Status report	Each wingman reports their armor, the most damaged first, and says if their ship is damaged.

Key	Order	What happens
`Alt+T`	Taunt	Sends your insult to the ship you have targeted, as the Communications MFD does.
no key	Return to base	They fly away from you and, once well clear, go home to the base you launched from.
no key	Radio silence	Switches their replies off or on.

Return to base and Radio silence have no key by default; give them one under Controls if you want them.

Every reply appears on the plaque at the top of the screen, for example “Wing: Forming up” or “Rook: Armor 75 of 100”.

- **One wingman or all of them.** Orders go to the whole wing. If you target one of your own wingmen first, Form on my wing, Break and attack, Help me out, Return to base and Status report go to that wingman only.
- **They always obey**, unless the order is impossible: “No target”, “Target not hostile” or “No wingmen in flight”.
- **Radio silence** hides their acknowledgements. Refusals and status reports still show.
- **What lasts.** Form on my wing and Break and attack stay in force, also after a jump. An attack order ends with its target, a jump or an autopilot. Every launch starts with your wingmen escorting you as usual, and radio silence off.
- **Sending a wingman home.** A wingman returning to base parks at the base you launched from, even if you jump or land somewhere else first. Launch from that base to fly with them again. Any new order calls a wingman on the way home back.
- **Flying in formation.** On Form on my wing your wingmen keep their places behind you as you fly, also when you use the afterburner (if their ships have one). A hard turn breaks the formation up for a few seconds; they rejoin once you fly straight again. They keep clear of you and of each other, and a wingman left far behind uses its afterburner to catch up.
- **Orders from the Communications MFD.** While your wingmen are in flight, the Communications MFD (`C`) lists “Wing” as its first contact. Pick it (`1`) to see every order as a numbered list: Form on me, Break, attack, Attack target, Help me out, Status report, Return to base and Radio silence (End silence while it is on). A pick there works exactly like the key, so it also reaches the orders that have no key. With “Wing” taking one of the four contact rows, the list shows at most three bases or ships besides it.
- The order keys and the “Wing” contact only work while Allow Hired Wingmen is on. They are never active in Classic.

Custom callsigns

The callsign pool is stored in:

```
content/wingmen/callsigns.json
```

OpenPrivateer creates the default file if it is missing and does not overwrite an edited version.

The file is a JSON array of non-empty strings. It is read as JSON5, so `//` and `/* */` comments and a comma after the last name are allowed:

```
[

  "Rook",

  "Nova",  // my favourite

]
```

If the file has an error, the built-in names are used and the log says where (line and column). Existing pilots keep their callsigns. Changes affect later hires.

Plain ASCII names are safest with the original game fonts.

14. Flight

Flight follows the original game's own flight model.

Flight includes throttle, afterburner, guns, ordnance, targeting, ITTS, radar, scanners, navigation, autopilot, jumps, docking, tractor beam, turrets, equipment damage and the Repair Droid.

In a turret view (`F5` / `F6`) your flight controls aim the turret instead of steering the ship: the arrow keys, the mouse, a joystick or a gamepad's left stick, whichever you fly with (in Classic, the input device chosen on the Options screen). Up aims the turret down and Down aims it up, as in flight. The ship holds its heading meanwhile.

A comm message that comes in while you sit in a turret or look out of the left, right or rear window is not heard or shown, not even when you look forward again; one already playing when you switch goes on to its end. This includes a base's landing-zone announcement and the lines of a contraband search. In Classic the same goes for the outside camera views.

A mod ship can have more than two turret seats. On such a ship `F5` moves you to the next seat that has a turret and `F6` to the previous one, going round from the last seat to the first; the top of the screen shows which seat you are in (`Turret 3`), and each seat aims, fires and recharges on its own. A seat that has no turret is skipped. At the Ship Dealer such a ship takes one turret per seat. When a turret could go on more than one empty seat of its kind, or you sell one and more than one seat carries it, the dealer asks which seat: the pointer becomes the turret's icon, the seats show as boxes on the ship picture (hover one to see `Turret 2` , `Turret 3` and so on), and clicking a seat buys or sells there; "Never mind" below them cancels. With only one seat to choose, the turret goes straight on (or comes off) that seat. All of this works in a multiplayer session too; there the seat you sit in is the one that fires.

To jump, fly within range of a jump point and press `J` . Jumping needs a jump drive: without one, `J` does nothing, and a damaged drive can fail to fire, in which case the press does nothing either. As in the original, `J` gives no message when no jump is possible there; only an empty jump-drive tank shows "No fuel" on the Destination page. A full tank holds 6 jumps on every ship (a Ship Dealer mod's jump drive can hold more), the FUEL gauge shows what is left of a full tank, and landing at any base fills the tank again for free. All of this works the same way in a multiplayer session.

`[` cycles the left display screen (the only one on ships with one) to its next page and `]` the right one, in the cockpit and in a turret view. Both can be rebound, to a key or a controller button, as **Cycle left MFD** and **Cycle right MFD** in the Controls settings.

The page keys bring up one page each: `C` Communications, `G` and `W` Weapons, `R` Damage Report, `M` Cargo Manifest, `D` and `A` Destination, `E` Examine Target, `V` View Object. In Enhanced and Modern they can be rebound in the Controls settings, in the **Targeting and MFD** group: **MFD: Communications page**, **MFD: Damage Report page**, **MFD: Cargo Manifest page**, **MFD: Destination page**, **MFD: Examine Target page** and **MFD: View Object page**. **Cycle guns**, **Cycle ordnance** and **Autopilot** bring up their page on whatever key or button they are on, as `G`, `W` and `A` do, and **MFD: Weapons page** (no key by default) brings up the Weapons page without stepping the guns or missiles. A page key you give to another action only does that action: with `W` on **Pitch nose down** for WASD steering, `W` steers and the Weapons page comes up on the key you gave **Cycle ordnance**. In Classic the page keys are always the original ones.

`W` selects the next missile or torpedo type your ship carries, and the Weapons display shows it on its `W:` line with the rounds left. In Enhanced and Modern, `Shift+W` steps backwards through the same list (rebindable as **Cycle ordnance backwards** in the Controls settings); with a weapon mod's extra missiles and torpedoes on board the list gets longer, and `Shift+W` reaches the last ones quickly.

`Enter` fires one missile or torpedo per press, from one launcher at a time. The joystick's trigger with Button 2, both mouse buttons together and a controller button bound to **Fire ordnance** keep firing while you hold them: each launcher fires as soon as it is ready, so with two missile launchers one press usually fires two missiles. A damaged launcher sometimes fails to fire; holding the button or pressing again tries it again at once. In a multiplayer session every press fires once. A damaged gun misses shots too: the more worn it is, the more often it fails and the slower it fires, and a shot it fails after drawing energy still costs that energy.

The cockpitless HUD

In Enhanced and Modern, `,` (comma) switches between your ship's cockpit and a HUD drawn as thin lines over open space, and back again. It works on every ship, from every view, and can be rebound as **Cockpit / generic HUD** in the Controls settings. Until you press `,` for the first time after starting the game or loading a save, each flight starts in the cockpit, except a mod ship that has no cockpit of its own: that one starts in the cockpitless HUD, and `,` shows its alias's cockpit. After that, every flight starts in the HUD you chose last.

In the front view the HUD shows, across the top, the **AUTO** light (lit when the autopilot can engage), your set speed and actual speed, and the **LOCK** light, which blinks red while a missile is homing on you (a fast flicker while the missile is far, slower as it closes in). The crosshair is your cockpit's own, in the same place, so you aim exactly as in the cockpit, and turns red on a lead marker as in every cockpit. At the bottom are two display screens with the radar between a fuel bar (shown with a jump drive) and an energy bar; a bar shows only its filled segments. The left screen shows your shields and armour by default: a bar outside and a block inside for each side (left, right, front at the top, rear at the bottom), each shrinking as it drains, with one tick per shield power level below them. The comm menu, the cargo manifest and the destination page appear there as in the cockpit, and an incoming message shows the speaker's face on a black screen until it ends, when the shield display comes back. The right screen shows the target by default (with a scanner fitted); `G` and `W` bring up the Weapons page there. `[` and `]` cycle the two screens. In a turret view the HUD shows the turret's screen, its energy bar and the

crosshair; the side and rear views show only space. Switching resets the screens' pages, except that a message being received stays up. The message line keeps its place (see "Message line" for moving it).

Two settings in the debug overlay's GAME section change the cockpitless HUD's look, in Enhanced and Modern: **HUD Opacity** (Auto, or 10% to 100%) fades its lines, bars, lights, radar circles and screen corners, while the display screens' pages, the speed readout, your cockpit's own crosshair and the comm face stay solid; **HUD Brackets** (Auto, On or Off) shows or hides the corner marks around the two display screens; **HUD Page Opacity** (Auto, or 10% to 100%) fades the display screens' text and pictures, the speed readout and the chase view's readouts (the comm face stays solid); **HUD Text Shadow** (Auto, On or Off) puts a thin black shadow under that text, so it stays readable over a bright planet or sun. Auto leaves them to the HUD style (full strength, corners on and no shadow unless a style says otherwise). All four apply at once and are kept between sessions.

With a wide Aspect, **HUD Layout** in the GRAPHICS section (Centred or Spread, `HudLayout=` in `openprivateer.ini`) decides where the cockpitless HUD sits: Centred (the default) keeps it in the middle 4:3 part; Spread moves the left display screen, the fuel bar and the AUTO light to the left edge of the wide picture and the right display screen, the energy bar and the LOCK light to the right edge, with the radar and the speed readout staying in the middle (in the chase view the corner parts go to the wide corners). A HUD style can move a part to another edge with `anchor`. At 4:3 and in Classic nothing moves.

Your own HUD style. A file `content/hud/hud_style.json` in the user-data folder (beside `content/mods`, see section 2) restyles the cockpitless HUD on every ship: move or recolour the bars, the radar, the lights and the speed readout, hide parts, draw a simple line crosshair instead of the cockpit's, set the opacity, and choose which pages each display screen shows, for example the Weapons page on the left screen:

```
{
  parts: {
    left_mfd: { pages: ['weapons', 'shields'] },
    right_mfd: { pages: ['target', 'damage'] },
  },
}
```

Each screen then cycles through its own list with `[` or `]` and starts a flight on the first page of its list; a page's key (`G` or `W` for Weapons) brings it up on the screen that lists it. Listing `comm` on the right screen moves the comm menu and incoming messages there. A mod ship may bring a style of its own; your file wins over it wherever both set something, and the two settings above win over both. A file with a mistake is left out and the reason is written to the log. Every member and its limits are listed in `doc/formats/ship_mod_format.md`, "The generic HUD's style". The file is read when the game starts and again with **Reload ship mods** in the debug overlay's MODS section (with **Auto reload** on, every time you save it; in flight the game must be paused).

Your own radar colours. A file `content/hud/radar_colours.json`, beside `hud_style.json`, sets the colours your radar shows contacts in, on every radar (the cockpit's, the turret views' and the cockpitless

HUD's), and which objects the target box colours. It helps when the usual red and blue are hard to tell apart; for red-green colour blindness, for example, hostile ships yellow and the others light blue on every scanner:

```
{

  all: { colours: { hostile: 34, friendly: 119 } },

}
```

`all` applies to every scanner; `scanners` sets one scanner by its Software Booth name and wins over `all` (a mod's new scanner follows the entry of the usual scanner it stands in for):

```
{

  all: { colours: { hostile: 34, friendly: 119 } },

  scanners: {

    'Iris Mk. I': { colours: { missile: 64 }, target_box: 'ships' },

  },

}
```

The colours are palette numbers from 0 to 255 for the classes `friendly` and `hostile` (ships), `base` (bases and planets), `missile`, `neutral` (pods and ejected pilots), `jump` and `nav` (the selected nav point); 0 is black and cannot be seen. `target_box` is `'none'` (every target box in grey), `'ships'` (ships in their friend or foe colour) or `'all'` (pods, pilots and bases coloured too). Your file wins over any mod and works in Enhanced and Modern, also on a multiplayer server; it is only yours and is never compared when you join. Classic ignores it. A file with a mistake is not used and the log says why. The file is read when the game starts and again with **Reload ship mods**, like `hud_style.json`.

The third-person view

In Enhanced and Modern, `F10` (or a gamepad's right stick click; the **Third-person view** control in the Controls settings) moves the camera above and behind your ship, and `F10` again, or `F1`, brings you back to the cockpit. It works from every view, the turret seats and the `F7`, `F8` and `F9` cameras included, and each flight starts in the cockpit. The ship always sits in the same place, a little below the middle of the screen; the camera follows its turns with a short swing, so you see the ship bank and turn, and comes straight back behind it when the turn ends.

You fly and fight as from the cockpit: the arrow keys, the mouse, a joystick or a gamepad steer the ship, the guns and missiles fire as usual and the missile lock and the radar work as in the front view. With the mouse the ship steers around the point the reticle rests on when you fly straight.

The HUD keeps out of the middle of the picture:

- **The reticle**, a small cross above the ship, shows where your guns hit: on a selected target within gun range it marks the target's distance, so a target under the reticle is in the line of fire, and with the

ITTS on it marks the lead point's distance, so the reticle turns red when a lead mark covers it, as the cockpitless HUD's crosshair does. Without a target it marks your longest gun's range. During a hard turn it swings with the camera and can leave the screen.
- **Target boxes, the missile lock, the lead marks and the nav marker** are drawn on the targets themselves, and the selected ship has its name and distance beside its box. When the selected target is off the screen, a small arrow at the screen's edge points toward it.
- **SET and KPS** sit left of the reticle, one above the other, the **AUTO** light under SET and the **LOCK** light under KPS.
- **The left display screen** (communications, cargo, destination) is at the top left and **the right one** (Weapons, Target, damage, View Object) at the top right. While the right one shows another page than Weapons, the gun and missile lines of the Weapons page are shown under it.
- **The radar** is at the bottom left and the fuel and energy bars side by side at the bottom right, with a small **shield and armour display** left of them: the Shields page's picture, smaller, always shown.

Messages at the top of the screen are drawn over the top display screens; in this view, setting **Plaque Position** to **Bottom** keeps them clear.

Switching to the third-person view puts the Weapons page up on the right display screen, when the ship has one, and the cargo page (or the destination page) on the left one if it showed the shields; the other pages stay as they were, and `[` , `]` or the page keys bring up any other page. The Shields page itself is not on the display screens in this view, since the shield and armour display shows it; back in the cockpitless HUD the left screen shows the shields again. Switching between the two keeps the communications menu and a message in progress as they were, and `F1` to `F4` take you back to the cockpit you had. `,` (comma) hides the whole HUD for a clean picture and shows it again. The missile camera (`F7`) returns to this view when the missile is gone. If you land in this view, the next flight starts in it too, with its HUD shown or hidden as you left it.

Third-Person View and **Camera** in the Controls settings (see "Controller settings" in section 6) turn the view off or choose how far behind the ship the camera sits: **Near**, **Medium** or **Far**, or **Rigid**, which keeps the camera fixed behind the ship without the swing. HUD Opacity and HUD Brackets apply as to the cockpitless HUD. In a multiplayer session the view works the same and nobody else sees it.

A ship or asteroid that comes closer to the camera than about half its distance to your ship is not drawn in this view, so nothing passing between the camera and your ship fills the screen; the selected target is always drawn.

With the **Medium** camera the view pulls back a little as you speed up and a little more under afterburner, and comes back in when you slow down.

Your own `content/hud/hud_style.json` (see "Your own HUD style") can also restyle this view's HUD with a `chase` member: move or recolour its parts, size the shield and armour display, move the target's label, or hide the off-screen arrow.

Shields

`S` steps the shield generator's power level down and `Shift` + `S` steps it back up, one level at a time, in every experience mode, as in the original. A lower level holds less charge per facing but draws less energy; level 0 switches the shields off and empties them. Pressing `S` again at level 0 switches them

back on at full power, and they recharge from empty. `Shift` + `S` stops at full power. A smaller generator has fewer levels; one with only on and off switches between the two on every `S`.

Scanner

Targeting requires a working scanner.

With no scanner fitted, the radar is absent and scanner-dependent targeting functions are unavailable. There is no hidden free scanner.

Jump points appear on the radar as dots when they are within scanner range: light blue with a B&S scanner, grey with the others.

A selected target reads “Unknown” on the Target display until the scanner identifies it; a better scanner identifies sooner. Steltek ships and Righteous Fire’s Salthi always read “Unknown”.

The Steltek drone

Only the boosted Steltek gun damages the Steltek drone. Every other gun, the Steltek gun before it is boosted, and every missile and torpedo hit it without effect, and ramming it does not hurt it either (your own ship still takes the collision).

Tractor beam

Select a cargo canister or an ejected pilot with `T` first; it does not need to be locked. With nothing selected, or with a ship, base or planet selected, `Enter` does nothing. A tractor beam in a turret is switched on and off with `Enter` from that turret’s view (`F5` / `F6`). A tractor beam on one of the ship’s own launcher hardpoints is selected with `W` from a forward view - it comes after the missiles and torpedoes and the Weapons display reads `W: Tractor Beam` - and then switched on and off with `Enter`. A beam keeps pulling the object it was switched on against, even if you change views or select something else. It switches itself off without a message when that object is gone, more than 10000 units away, does not fit in the hold, or the ship runs out of energy. A beam on a launcher hardpoint is drawn from that hardpoint, below the nose.

As in the original, several tractor beams can run at once and each one is drawn. Each beam that pulls an object in adds it to the cargo hold, so two beams on one object give you two of it.

`X` jettisons the cargo hold: the goods of one cargo class leave together in one canister (Grain, Generic Foods and Luxury Foods share a Food canister) and rescued pilots leave as one pod each. Anything in the secret compartment stays. The Ancient Artifact stays too, unless another alien relic of its class is listed before it in the hold: then it leaves in that relic’s canister. A canister tractored back needs room for everything it holds. Mission cargo leaves too. A canister tractored back in returns the same goods (contraband fills a fitted secret compartment first), so recovered mission cargo is mission cargo again and still counts for its mission.

Autopilot

Autopilot follows the original instant-travel behavior. It drops you out at the edge of the destination’s own area, so the ships waiting there are already present when you arrive: about 12,000 m from a jump point, further out from a base or planet (17,000 m or more), and nearer to some smaller nav points. It

drops out early when the route enters another encounter zone on the way, including asteroid fields. You come out of autopilot still moving, at the speed the autopilot flew (your ship's top speed), facing the area you stopped in; your SET speed is the one you had before, and the ship then speeds up or slows down to it as after any throttle change. Autopilot reports "Navpoint near" once you are already inside the destination's area, and "Receiving message" while a comm message is coming in: it waits until the message has ended, in single-player and in a multiplayer session alike. In single-player no base or planet announces its landing zone while the autopilot is flying you, and the flyby ends any comm message that started during it; a base whose landing zone you arrive in announces itself as the cockpit comes back. In Classic you do not hear or see other pilots talking during the flyby; in Enhanced and Modern their messages play over it.

In Enhanced and Modern a Repair Droid keeps working through the trip: when you arrive it has done the repairs it would have made flying the distance at your top cruise speed (a damaged drive makes the trip slower, so more gets repaired). It repairs what it always repairs, in the same order, never armour or weapons. In Classic, and in multiplayer, the trip itself repairs nothing.

Asteroids

In an asteroid field the rocks are solid: a rock that hits your ship damages it and bursts. Gunfire bursts rocks too, yours and every other ship's: a shot, missile or torpedo that meets a rock is spent on it, so a rock between you and your target takes the shot. Bursting a rock earns nothing and counts as no kill, and the field keeps the same number of rocks around you. In multiplayer the rocks are scenery: ships and shots pass through them.

Mining with turrets

In Enhanced and Modern a ship with turrets can mine an asteroid field:

1. Buy at least one ton of **Mining Eqpmnt** (sold at New Detroit and at refineries) and keep it in the hold. It is not used up.
2. Fly into an asteroid field and come to a full stop: `Backspace` (SET 0), 10 kps or less on the speed readout, no afterburner, no autopilot.
3. Burst rocks with your turrets: from the turret seat (`F5` / `F6`), or let the turrets do it. With no hostile around and nothing on a collision course, an idle turret shoots any rock it can reach.
4. About every other rock a turret bursts leaves an ore pod (Iron most often, then Tungsten, Uranium and, rarely, Plutonium, 1 or 2 tons). Tractor the pods in as any cargo canister. A fresh pod cannot be shot for a few seconds, so the turrets do not destroy what they have just mined.
5. Sell the ore at a refinery or New Detroit.

The first time in a flight that everything is in place in a field, the message line says "Mining ready" with the pods the field has left.

Only turret fire mines: the forward guns, missiles and torpedoes burst rocks and leave nothing, and so does any shot while the ship is moving. Each field gives at most 12 pods per flight, fewer as it thins out; the message line says "Asteroid field depleted" when it is spent. Landing refills every field. A full field is worth roughly 3,000 credits. The message line also tells you once a flight when you need Mining Eqpmnt or a full stop. Of the stock ships only the Galaxy can mine: its two turrets reach the whole field, while the Orion's and Centurion's tail turrets point behind the ship, where the field keeps no rocks. Never in multiplayer.

Turret Mining in the GAME section of the settings overlay switches it on or off: off by default in Enhanced, on in Modern. Turrets Hold Fire stops the turrets mining by themselves; mining from the seat still works.

The numbers above are the game's own. An item mod can change them (how often a rock yields, how many pods a field gives, what they hold, the equipment needed and whether it wears out: then a pod now and then uses up a ton of it and the message line says "Mining equipment worn out"), also only in some systems; `--commodity-report` lists the mining rules in effect (see section 20). A ship mod can make one of its ships unable to mine.

Piracy

In Enhanced and Modern, with **Piracy** on (the GAME section of the settings overlay: off by default in Enhanced, on in Modern), you can rob a trader instead of shooting it:

1. Target a ship (`T`) within 3000 of you: a merchant (a Drayman, a Galaxy or a Tarsus), or a bounty hunter, pirate or Retro that is not hostile to you, carrying goods.
2. Open comms (`C`). After its usual lines the ship's menu offers **"Drop your cargo. Now."** Pick it.
3. The ship answers a moment later, in one of three ways:
    - **It complies** ("All right! Take it and leave us alone!"): it dumps half of each kind of goods it carries, at least one ton of each, in cargo canisters beside it, and flies off. Tractor the canisters in as any cargo. It keeps the rest and its pilots.
    - **It runs** ("Not a chance, pirate!"): it turns away from you and flies off as fast as it can, without firing. Chase it down: once your shots reach its armour it turns hostile and fights back.
    - **It fights**: it turns hostile and attacks you, and so does the rest of its group nearby.

How a ship answers depends on who flies it: timid merchants usually give in, confident ones less often, fanatical ones and bounty hunters mostly fight, and pirates and Retros hardly ever comply. A novice gives in more easily than an ace, a damaged ship more easily than an intact one, and armed friends nearby (its own wing, the Militia or the Confederation) make it braver. Your ship counts too: a Centurion is more persuasive than an Orion, an Orion more than a Galaxy, a Tarsus least. The answer is decided when you ask, so loading a save and asking again gets the same answer.

Every ship you rob costs standing, the first time you ask it, whatever it answers: half of what destroying it would cost. A Drayman costs 5 with the Merchants, 7 with the bounty hunters and 5 each with the Confederation and the Militia, and gains you 7 with the pirates. With **Show Mission Feedback** on, the message line shows it ("Piracy: Merchant -5, Pirate +7"). Every Militia or Confederation ship within 6000 that sees the demand turns hostile to you on the spot ("Militia patrol witnessed the demand"). After about six Draymen the Militia is hostile and its patrols attack you on sight; after four, pirates stop attacking you. Destroying a ship you robbed costs its full standing on top, as any kill does.

A ship is asked at most three times, ten seconds apart, and not again once it has complied or while it is running. The row is never offered to the Militia, the Confederation, the Kilrathi, ships of a mission or the story, stations or capital ships, a ship with nothing but its pilots aboard, or while the autopilot flies. Never in Classic or multiplayer. With pirates friendly to you, ships give in a little more easily.

Stolen goods. The goods a ship dumps for you are stolen. Tractored in, they go into your secret compartment first, if your ship has one, as contraband does. A Militia or Confederation patrol that searches you finds stolen goods in the normal hold as it finds contraband, and attacks; in the

compartment they stay hidden. Only a pirate base buys stolen goods; at any other Commodity Exchange you can sell only the goods of that kind you came by honestly, and the rest is answered with "STOLEN GOODS NOT BOUGHT HERE". A pirate base takes the stolen ones first. Goods you jettison and pick up again stay stolen.

The Merchants' Guild. When your standing with the Merchants falls to hostile, the guild revokes your membership the next time you land (a message in the notification center). It takes you back only once your standing is neutral again; until then the guild master does not offer the membership. A pirate base also buys the Mission Cargo of your cargo missions (250 credits a ton): the guild revokes your membership at once, you lose 50 standing with the Merchants and 100 with the bounty hunters and gain 25 with the pirates, and every cargo mission whose cargo is no longer all aboard fails.

Bounties for piracy. While Piracy is on and Kill Bounties are on, the pirates pay a bounty for every merchant you destroy, at a pirate base.

The Mission Computer's FIN screen shows **Demands:** (the demands ships gave in to, out of all you made) and **Fenced:** (the credits pirate bases paid you for stolen goods and Mission Cargo).

These are the game's own rules. A mission mod can change them (the range, how ships answer, what a demand costs, who witnesses it, how stolen goods are searched and sold, when the Merchants' Guild revokes) and can let you rob the ships of its own missions, also after loading a save. A mission mod can also post pirate jobs on the Mission Computer of the pirate bases: raiding a merchant convoy before its freighters reach the jump point (a freighter that gets away fails the job, and you may demand the freighters' cargo instead of destroying them), a smuggling run past a Militia patrol, or a hit on a Militia captain. An NPC mod can give the ships their own answers and the row its own words. `--missions-report` lists the piracy rules in effect (see section 20).

Refuelling in flight

In Enhanced and Modern you can call a fuel tender in any system instead of landing to fill your jump drive:

1. Open comms (`C`). While your ship has a jump drive that is not full and you have not refuelled yet on this flight, the comm menu lists **Fuel Service** after the other contacts (alone, it opens straight on its line). Pick it, then **"Hail to refuel"**.
2. A fuel tender, a Drayman freighter, sets out toward you: the message line says how much it costs (1,000 credits) and about when it arrives (10 to 15 seconds). It flies to you and stops about 1000 away; you can fly toward it to meet it sooner.
3. Once it is within 1500 of you, come to a **full stop** (speed 0). Refuelling takes 5 seconds; moving during them stops it, and it starts again at your next full stop.
4. When it is done your tank is full, the price is taken from your credits, and the tender flies off and jumps out.

One refuel per flight, from takeoff to landing (a jump does not give you another; landing does). The Fuel Service row does not show while your tank is full, after this flight's refuel, or while a tender is already on its way. The hail is refused while you are in combat (you hit a ship or were hit in the last 30 seconds) or while a hostile ship is within 11700 of you, and when you cannot pay the price (then a message in the notification center names the price and your credits). The refuel is called off, at no charge, if a hostile

ship comes within 11700, you jump, land or engage the autopilot, the tender is destroyed, or about a minute passes without a refuel (the tender gives up).

In a multiplayer session the server sends the tender, and every pilot there sees it as an ordinary ship; the price is taken from your own credits when the refuel completes, and a refused hail for credits shows on the message line. Classic has no fuel tender. An item mod can change the price, the hostile range, the times and the stop tolerance, or turn the service off (see section 20); `--dealer-report` lists the rules in effect.

Message line

Comm messages, mission progress and other short notices appear on the cockpit message line, a bar at the bottom of the screen. **Plaque Position** in the GAME section of the settings overlay moves it, in every cockpit and every experience mode, Classic included:

- **Bottom** (the default): at the bottom of the screen, as in the original.
- **Top**: at the top of the screen. While the missile camera's "Missile Camera On/Off" message is up (`F7`), the line sits just below it.
- **Off**: the message line and the missile camera message are not shown. Spoken comm messages are still heard; a message that is only text is not seen.

The setting also places a base's greeting caption shown while you land. The "Game Paused" message always shows in the middle of the screen. The setting is remembered between runs.

Pause

Press `P` in flight.

15. Navigation Map

The Navigation Map uses the original sector and quadrant data.

It shows systems, nav points, bases, jump points and mission objectives. As in the original, only jump points are blue circles; bases are green squares, and pirate bases and empty nav points are green circles. Story-locked systems and routes remain hidden until the game makes them available. Once a story jump point is shown (the Kilrathi systems beyond Rygannon), it can be flown through like any other jump point.

Active mission targets are highlighted.

Your ship is the white dot. When you fly beyond the map's edge, the dot is not shown until you come back within range.

The map's keys work in every experience mode, as in the original:

Key	System view	Quadrant view
`N`	Next nav point	Back to the system view
`B`	Previous nav point	Previous quadrant

Key	System view	Quadrant view
`Q`	Quadrant view	Next quadrant
`M`	Missions; again: next mission	Missions; again: next mission
`I`	Back to the nav point information	Back to the nav point information

`M` does not turn the missions off: press `I` or click `NAV/MIS`. While the missions show in the system view, `N` and `B` page through them. The keys and the `QUAD/SYS` and `NAV/MIS` buttons can be mixed freely.

In Enhanced and Modern the map remembers the asteroid fields you have flown into. The first time your ship enters a field, the system view marks it with a dashed brown ring the size of the field, under the nav point markers. Fields without a nav point of their own are marked too, so the map shows rocks you would otherwise only meet by chance. The rings appear only in the system view, never in the quadrant view or in a quadrant whose map you do not own. They are saved with your pilot: an older save or a new pilot starts with none. To hide them, uncheck **Asteroid Field Marks** in the GAME section of the settings overlay; fields you fly into are still remembered and show again when you check it. Classic draws no rings and remembers nothing.

Plotting a route

In Enhanced and Modern the map can plan your jumps for you:

- `P`: in Quadrant View, select a system (click it) and press `P`. The shortest route there is drawn in yellow and its destination is ringed red; the info panel lists the systems on the way.
- `Ctrl` + `N`: plots a route to the destination of your mission, in flight or on the map. With the mission page (`NAV/MIS`) showing, it uses the mission shown there; otherwise your first mission.

Plotting selects the jump point toward the next system right away, so `A` (autopilot) flies you there and the Destination page names it. Press `J` there as usual. After every jump the next jump point is selected again. In the destination system the mission's nav point (or the system's only base) is selected and the route ends. If you jump off the route, a new route from where you are is planned. If you pick another nav point on the map, your choice counts for that system.

A message at the top of the screen tells you what happened, for example **Route: 3 jumps to Pentonville**, or **Refuel: 7 jumps to …** when your jump drive does not hold enough fuel for the whole trip (the route is still set; land somewhere or call a fuel tender to refuel, see "Refuelling in flight"). Without a jump drive no route is set (**No jump drive**). Pressing `P` on the route's destination again clears the route.

Routes also pass through quadrants whose map you have not bought, but nothing can be selected there, so you pick the jump point yourself. Story-locked jump points are only used once the story opens them.

A route is not saved. It ends when you land, load a game, start a new game or quit to the title menu. Both keys can be changed under the Navigation Map controls. To turn route plotting off, uncheck **Nav Route Plotting** in the GAME section of the settings overlay. Classic has no route plotting.

The optional **Open Universe** setting exposes additional universe content and is not part of Classic behavior.

16. Guilds and Random Missions

The Merchants' Guild, Mercenaries' Guild and Mission Computer provide the original non-plot mission types:

- Cargo
- Patrol
- Scout
- Defend
- Attack
- Bounty

Guild fees are:

Guild	Fee
Merchants' Guild	1,000 credits
Mercenaries' Guild	5,000 credits

Both guild computers show the original **Scanning … records … Please stand by…** stage while generating a fresh batch of offers.

In Enhanced and Modern a stranger in a base's bar may also offer an attack, bounty, cargo, sabotage, spy or smuggling job; see **Fixers in the bars** in section 4.

Notification center

In Enhanced and Modern, every mission the game pays or fails leaves a message in your notification center. On a Launch Pad press:

```
M
```

to open it. After a landing that completed or failed a mission it opens by itself once you are on the Launch Pad.

- A **completed** mission shows what it paid. A **failed** one shows the reason the game decided, for example **Nav point 2 not reached**, **Target not destroyed**, **Not delivered to Liverpool** or **Cargo missing (10 x Grain needed)**, and whether the mission cargo was taken back. Each message also shows where it happened, the mission number and the time.
- A mission you remove with the Mission Computer's **REJCT** button leaves no message.
- A mission whose destination base was destroyed fails on your next landing, with the reason **<base> was destroyed**, and its mission cargo is taken back. The missions a mod adds to the boards never send you to a destroyed base, and the boards never offer a destroyed added base or carrier as a destination.

- Loading a save whose ship or equipment came from a mod that is no longer installed leaves a message naming what your ship carries in its place.
- An unread message has a bold title. **MARK READ** marks it read.
- The red **DELETE** button removes a message after you confirm.
- **MARK ALL READ** and the red **DELETE ALL** (after you confirm) act on every message.
- Newest messages are at the top; scroll the list with the mouse wheel. **M**, **Escape** or **CLOSE** closes it; Escape first cancels an open confirmation.
- The key can be changed under **Notification center** in Remap Controls. It has no controller button by default; you can assign one there.

Messages are saved with your game. Classic keeps recording them without showing them, so they appear once you switch to Enhanced or Modern. The notification center is not available in multiplayer.

Mission progress in flight

In Enhanced and Modern, the cockpit message line (see "Message line" in section 14) tells you when you complete a mission objective in flight. It shows the objective as the Navigation Map lists it, for example:

- **GO TO DELTA, NAV 2: COMPLETE** when you reach a patrol nav point
- **DEFEND PALAN, NAV 1: COMPLETE** when a nav point is clear of hostiles
- **DESTROY RHOMBUS: COMPLETE** when the target is destroyed

Once everything that needs you in flight is done, it shows **OBJECTIVES MET - LAND AT LIVERPOOL** (or just **OBJECTIVES MET** when the mission names no base). You are paid when you land. Travel legs and cargo deliveries show nothing in flight. With more than one mission carried, each message starts with the mission's number as the Navigation Map pages show it, for example **MISSION 2 - …**. In single-player, turning on Invulnerability or Unlimited Ammo in Options voids every mission you carry, and the message line says so.

Each message stays for 3 seconds, one at a time. A comm message always comes first; a mission message it interrupts is shown again afterwards. They are not shown in a turret view. The Navigation Map's objective lines also change colour as you complete them.

Untick **Show Mission Feedback** in the GAME section of the settings overlay to turn them off. They are not shown in Classic. In a multiplayer session they show as the server reports your objectives.

Kill bounties

In Enhanced and Modern, the Militia and the Confederation pay a bounty for hostile ships you destroy:

- the Militia for **pirates** and **Retros**,
- the Confederation for **Kilrathi**.

The bounty is queued when the kill counts for you (the same kills the Mission Computer's **FIN** page counts) and paid the next time you land at a base that is not a pirate base. Kills made by your wingmen do not count, and neither do the ships of a story mission. Mission kills pay a bounty on top of the mission's own pay.

What a kill pays depends on the pilot's rank (an ace pays more than a rookie), the ship (a Talon pays the least, a Kilrathi Kamekh the most) and the faction (Kilrathi pay more than pirates, pirates more than Retros), varied by up to half either way. A pro pirate Talon pays about 300 credits. One kill pays at most 1,000 credits; after a payer's tenth kill in one flight each further kill pays half; and one flight earns at most 20,000 credits in bounties.

A payer only pays while your standing with it is at least neutral (the **N** or **F** tag on the FIN page). What it will not pay yet stays queued for a later landing. The landing's payment leaves a **Bounties** message in the notification center, which also names what is still held and why. The FIN page shows the bounties still queued on its **Bounties** line. Collecting a bounty does not change your standing.

With **Show Mission Feedback** ticked, each bounty also shows on the cockpit message line as it is earned: "Bounty +300", or "Bounty +120, payable at a pirate base" when only the pirates pay it. Once a flight has earned its 20,000 credits, the next kill says "Bounty limit for this flight reached" and later kills that flight show nothing.

Bounties are on by default in Modern and off in Enhanced. Tick or untick **Kill Bounties** in the GAME section of the settings overlay to change it; ticking it back to your mode's default returns it to the default. Queued bounties are saved with your game and kept while the setting is off; they are not paid until it is on again. There are no bounties in Classic.

In multiplayer the server decides: a server started with `--kill-bounties` pays bounties to every pilot in its sessions, whatever each player's own Kill Bounties setting says, and one without it pays none. A kill counts for the pilot whose shot or ram hit the ship last, and only for that pilot, so two pilots shooting at the same pirate never both collect for it. The bounty is queued on your own machine and paid at your own landings, as in single-player, at the game's own amounts or, when the server runs a mission mod with bounty rules, at that mod's amounts (the server's rules, whatever mods you have yourself). There is no cockpit line for a bounty in multiplayer; the FIN page shows the queue. The same kill counts toward your standing and kill record in every session, with or without bounties.

The amounts and rules above are the game's own. An installed mission mod (with **Mission Mods** on) can change them: what each faction's ships pay, the caps, who pays and where, and how well you must stand with a payer (some mods ask you to be **friendly**). A mod can also have the pirates pay for merchant, mercenary, Militia and Confederation ships, paid only at a pirate base and only while the pirates think well of you, or switch a payer or the bounties off altogether; a payer that pays nothing is named in the landing's message as paying no bounties. A mod's own jobs may pay their mission pay only, with no bounty for their ships; such a job keeps that rule even if you later remove the mod. A mod can give a pilot faction of a universe mod (a pirate clan, say) a bounty and a payer of its own, named in the landing's message, and it can cut the pay for kills at one spot (the ships at a nav point come back each time you return there), so that waiting at a jump point for the same squad pays less and less.

17. Music and Sound

OpenPrivateer supports several music backends, including OPL2, MIDI with an external SoundFont, OS MIDI and CD-style replacement music. Availability depends on the platform.

Volume

The settings overlay's VOLUME section has three sliders, each from 0 to 127 and remembered between runs:

- **Music**: the music, with every backend.
- **SFX**: sound effects: engines, guns, missiles, explosions, cockpit and cutscene sounds.
- **Speech**: every spoken line: base conversations, in-flight comm messages and cutscene voices.

Music and Speech start at 64 and SFX at 48, so the effects sit a little under the music; once you move a slider your setting is kept.

The three are independent of one another, and a change is heard at once, also on music or a line that is already playing. The Options screen's **Music** and **Sound F/X** switches still turn music, or sound effects and speech, off altogether.

Under the sliders, **Mute when minimized** (off unless you turn it on, remembered between runs) silences all sound, music, sound effects and speech alike, while the game window is minimized. The game goes on playing meanwhile; when you restore the window the sound comes back at the volumes the sliders show. Switching to another window without minimizing the game keeps the sound on.

SoundFont

Select a `.sf2` file in the launcher's **Music** section or from the SoundFont setting when using the corresponding MIDI backend. The path is stored in `openprivateer.ini`. Without one, OpenPrivateer looks for `openprivateer.sf2` in the user-data folder, then next to the program.

Replacement music

The `music/` folder accepts your own music files in place of the game's built-in music. WAV, MP3 and FLAC are supported. To hear them, choose **CD (your own files)** in the launcher's **Music** section, or **CD** as the **Music Backend** in the settings overlay.

Each piece of the game's music has a slot: a subfolder and a two-digit number. Name your file with that number, an underscore and anything you like, for example `basetune/04_mining.mp3`. The subfolders are `basetune` (the music at each kind of base), `basetune2` (Righteous Fire's Gaea), `combat` (the music in space, by situation), `combat/000` (the short bridges between two pieces in space), and `opening`, `opening2` and `victory` (the intro, Righteous Fire's intro and the victory movie). A base's music, a movie's music and the music of a lasting situation in space repeat the whole file; short pieces play once. A slot without a file stays silent while this backend is chosen.

The generated `music/README.txt` lists every slot number. It is rewritten each time the game starts, so it always matches the installed build.

Phase torpedo decode music

Put one WAV, MP3 or FLAC file in `music/phase_decode/` to hear it while a phase torpedo decodes, for example `01_decode.wav`. It works with every music backend. When a decode starts, the other music stops and your file plays once from its beginning; it does not repeat, and the decode completing does not stop it. It stops, and the other music comes back, when you fire the torpedo or the decode is lost

(turning away, the target leaving the screen, another target or weapon, leaving the cockpit view, a jump, landing). The next decode plays it from the beginning again. With several files in the folder, the first in name order plays. It follows the Music volume and stays silent with Music off. The folder takes the music only; a sound when the decode completes comes from the weapon mod that makes the phase torpedo, if it brings one.

Music of an interior mod

A base with an interior of its own (section 20) can bring its own music as WAV, FLAC or MP3 files: one piece for the whole base, and pieces for single rooms, such as a bar with music of its own. It plays with every music backend, follows the Music volume and stays silent with Music off; Sound F/X does not affect it. Going between two rooms with different pieces fades one into the other; rooms sharing a piece let it play on. The shops and terminals you open from a room keep the room's music. Such a base can also play music of its own over the landing and takeoff movies; on landing it can run straight on into the base's music.

18. Graphics

OpenPrivateer renders the original game at its native logical resolution and scales it for modern displays.

Available settings include CRT filtering, image adjustments, scaling, texture filtering, pixel scaling and splash font selection.

Classic keeps the original's look, including its quirks. Enhanced and Modern apply some presentation improvements, such as keeping round sprites round while the ship rolls.

Fullscreen

`F11` or `Alt+Enter` switches the game window between fullscreen and a window, on every screen. The game remembers whether you left it in fullscreen and starts that way next time (the launcher always starts windowed). In `openprivateer.ini` it is `Fullscreen=1` (fullscreen) or `Fullscreen=0` (a window, the default). Leaving fullscreen brings back the window at the size it had before.

CRT filter

CRT Filter in the debug overlay's Graphics settings, just above COLOR GRADING, draws the picture like an old monitor. Its dropdown has three choices:

- **DOSBox Staging** (the default): the VGA look DOSBox Staging uses for this kind of game, every other screen row darkened into scanlines.
- **Hyllian**: CRT-Hyllian, softer round scanlines with a slightly sharpened picture.
- **Off**: the plain picture.

In `openprivateer.ini` it is `CRTFilter=DOSBoxStaging`, `Hyllian` or `Off`. `Ctrl+C` outside space flight switches between Off and the last choice that was on. The filter only shows when the picture is at least 800 pixels high; in a smaller window you see the plain picture.

Under the dropdown, **CRT Filter Settings** (click it to open) has a slider for every setting of the chosen filter, with its value in the label. Each filter keeps its own settings, so trying the other one does not change them. Changes show at once and are saved. **Reset CRT settings** puts back the chosen filter's defaults; **Reset graphics** puts back both filters' defaults and DOSBox Staging.

Both filters:

- **Phosphor layout**: the pattern of colored dots laid over the picture: Off, Aperture grille (thin columns) or Shadow mask.
- **Mask strength**: how visible that pattern is. 0 hides it.
- **Input gamma**: higher makes the middle tones darker.
- **Output gamma**: higher makes the middle tones brighter.

DOSBox Staging only:

- **Color boost even**: how much the darkened scanline rows are brightened again.
- **Color boost odd**: how much the rows between the scanlines are brightened.
- **Scanline min**: how dark the scanlines are over bright parts of the picture.
- **Scanline max**: how dark the scanlines are over dark parts.

Hyllian only:

- **Color boost**: brightens the whole picture to make up for the scanlines.
- **Scanlines**: how strongly each line fades towards its edges. 0 turns the scanlines off.
- **Beam min width** and **Beam max width**: how thick the lines are in dark and in bright parts of the picture. Smaller values give thinner lines and darker gaps.
- **Anti-ringing**: removes the faint light halos beside hard edges. 0 keeps them.

Each setting is also in `openprivateer.ini`, as `CRTDosbox...=` and `CRTHyllian...=` lines (for example `CRTDosboxScanlineMax=0.85`). The Hyllian mode's defaults were made for a sharper picture than this game's, so it can be worth adjusting them to taste. Higher Render Scale or Pixel Scaler settings give the Hyllian mode more, finer scanlines.

Color grading

COLOR GRADING, just below the CRT filter in the debug overlay's Graphics settings, turns the color grading on and off (it is on by default). Its settings (color space, color profile, brightness, contrast, gamma, saturation, color temperature, the red, green and blue gains and the black level) are in **Color Grading Settings** right under the checkbox: click it to open. You can open it with color grading off too; its settings are then greyed out. Changes show at once and are saved.

Aspect (widescreen space flight)

Aspect in the debug overlay's settings, under Render Scale (`Aspect=Auto`, `4:3`, `16:10`, `16:9`, `21:9` or `32:9` in `openprivateer.ini`), lets space flight fill a wide screen. The default is 4:3: the whole game is a 4:3 picture with black bars at the sides of a wider window. With 16:10, 16:9, 21:9 or 32:9 the view of space continues past the sides of the 4:3 picture: you see more of space to the left and right, while everything keeps its size and its height on screen. Auto matches your window's own shape, from 4:3 up to 32:9, and follows the window when you resize it; a window wider than 32:9 shows the 32:9 picture

with black bars. At 32:9 ships near the left and right edges look stretched, as in any very wide view. The change applies at once, without a restart; a screen recording (Ctrl+F7) running at that moment ends, since a recording keeps one picture size.

What goes wide: the generic HUD (`,`), the chase view (F10), the outside views (F7 to F9), the autopilot fly-by, the jump charge-up and the death camera, and the cockpit with its pictures hidden. The HUD itself (MFDs, radar, gauges, crosshair, messages) stays in the middle, where it is at 4:3; target boxes, the lead pips and the mouse-flight pointer follow into the sides. A base scene with a wide picture (the rendered base scenes you build with the picture pack tool, or a mod's) fills the sides too: the room continues to the left and right, while the doors, people and everything you click stay in the middle 4:3 part. Other base scenes, the shops and computers, menus, the navigation map, conversations, the intro and the landing, takeoff and jump movies stay 4:3 with black sides. A phase torpedo's decode and a missile's lock still need the target inside the middle 4:3 part of the view. Classic is always 4:3 (the setting is greyed out there).

Cockpit Sides, under Aspect (`CockpitSides=Black` or `Open` in `openprivateer.ini`), decides what the cockpit views show beside the cockpit picture in a wide picture. Black (the default) keeps black bars at the sides of the cockpit, as at 4:3. Open lets space continue beside the cockpit picture, which stays in the middle. A cockpit picture made for wide screens (from a ships mod, or the rendered cockpits you build with the picture pack tool) fills the sides by itself with either setting: the cockpit's consoles and frame continue to the edges, and you look out into space through its openings. A wide cockpit picture is made for 21:9; at 16:10 or 16:9 you see its middle part, and at 32:9 the sides beyond it are black or space as this setting says. The setting is greyed out in Classic.

Lock Zone Mark, a checkbox in the GRAPHICS section (`LockZoneMark=On` or `Off`, off by default), draws two faint grey lines at the edges of the middle 4:3 part when the forward view is wide (the cockpitless HUD, or a cockpit with open or wide sides). A target between the lines can be decoded with a phase torpedo and locked with a missile; one in the sides beyond them cannot. The lines never show at 4:3, in the side, rear, turret and outside views, or in Classic.

Vignette

Vignette, a slider in the debug overlay's GRAPHICS section, below Frame Limit (`VigStrength=0.00` to `3.00` in `openprivateer.ini`), darkens the view of space towards its edges and blacks out its corners, so the clear part of the view looks like a smaller oval, as if seen through a visor. The darkening covers the whole visible view, cockpit and HUD included, and follows its shape: on a wide picture the oval is wide too, so all four corners are equally dark at 4:3, 16:9, 21:9 and 32:9, and black bars beside a 4:3 cockpit stay outside it. 0 (the default) turns it off; 1 gives a moderate shade with clearly darker corners; 2 is strong, with black corners and a noticeably smaller clear area; 3 is the strongest, with the edges black too and only a small clear oval in the middle. Values in between blend smoothly, and the change shows at once.

It applies in space flight only: the cockpit, the cockpitless HUD, the turret, side and rear views, the chase and outside views, the autopilot fly-by, the jump charge-up and the death camera. Menus, bases and their screens, the navigation map, conversations and movies are never darkened, and neither are the mouse pointer, the pause plaque, the FPS counter and the debug overlay. It works in every experience mode, Classic included, and changes nothing but the picture. Reset graphics turns it off.

FPS Counter

FPS Counter in the debug overlay's GRAPHICS section, between Present Mode and Frame Limit (`FpsCounter=Off`, `TopLeft`, `TopRight`, `BottomLeft` or `BottomRight` in `openprivateer.ini`) shows the frame rate in that corner of the game's picture: frames per second on the first line and the time of one frame in milliseconds (MS) under it, both averaged over half a second and updated twice a second. It sits in the corner of the whole picture, so with a wide Aspect it moves out to the wide picture's corners, and it is part of the picture: Render Scale, the Pixel Scaler and the CRT filter draw it like everything else. It shows on every screen, in flight, on bases, in menus and in movies, and in Classic too. Off is the default; Reset graphics turns it off.

Present Mode

Present Mode in the debug overlay's GRAPHICS section, above FPS Counter (`PresentMode=VSync`, `Mailbox` or `Immediate` in `openprivateer.ini`), decides how finished pictures reach your display:

- **VSync** (the default) waits for each refresh of the display: no tearing, and the frame rate never goes above the display's refresh rate.
- **Mailbox** draws as many frames as the computer can and shows the newest one at each refresh: no tearing, a little less delay, but more power use and heat.
- **Immediate** draws as many frames as the computer can and shows each one at once: the least delay, but the picture can tear (a visible horizontal break while things move).

A change applies at once, in every experience mode. Not every graphics driver or system offers Mailbox and Immediate (on a Mac, Mailbox is usually missing): a mode your system does not offer is marked "(unsupported)" in the list, and if you pick it the game keeps VSync and the entry reads "(using VSync)". Reset graphics sets VSync.

Frame Limit

Frame Limit in the debug overlay's GRAPHICS section, under FPS Counter (`FrameLimit=Off`, `15`, `30` or `60` in `openprivateer.ini`), caps how many frames per second the game draws, to save power and keep a laptop or a handheld cooler. **Off** (the default) draws as many as Present Mode allows; 15, 30 or 60 FPS hold the game at that rate at most. With VSync a cap above your display's refresh rate changes nothing, and one below it still applies; with Mailbox and Immediate the cap holds exactly.

While the debug overlay is open, or the game window is minimized, the game draws at most 60 frames per second, also with Off, so the overlay itself does not run the computer at full speed. A lower cap you chose stays. Closing the overlay lifts this at once. Keep this in mind when you read the frame rate in the overlay's performance figures: close the overlay to see the game's full rate.

The cap changes only how often the picture is drawn: the game runs at the same speed, ships fly and fight the same and a multiplayer session stays connected at every setting. The FPS counter shows the capped rate. It works in every experience mode, Classic included. Reset graphics sets Off.

Hull Atlas

Hull Atlas in the debug overlay's MODS section (at its end, or `HullAtlas=Auto`, `On` or `Off` in `openprivateer.ini`) decides how the pictures of ships from ship mods reach the graphics card. With

Auto (the default) or On, each ship's pictures travel together as one large image, so a mod ship appears for the first time without a stutter. Off sends every picture on its own, which takes a little less graphics memory (the combined image holds about 6 to 20 percent more) but costs a short pause each time a new mod ship appears. Turn it off only if a system short of graphics memory has trouble with many mod ships at once. The ships look the same either way, and the original game's ships are not affected. A change applies to ships that appear afterwards; ships already shown keep theirs until the next star system. An unknown value in the ini file reads as Auto.

19. Wallpaper

The `wallpaper/` folder accepts custom artwork for the Privateer / Righteous Fire selector.

Use filenames beginning with:

```
01_
```

for Privateer and:

```
02_
```

for Righteous Fire.

PNG, JPG and BMP are supported. See `wallpaper/README.txt` for the current details.

20. Content Overrides and Translations

OpenPrivateer can load replacement files from the filesystem without changing the original TRE archives.

The basic layout is:

```
content/

    priv/

    rf/

    i18n/

    mods/universe/

    mods/weapons/
```

`content/priv/` and `content/rf/` mirror paths inside the corresponding original archive.

Each universe mod is a folder in `content/mods/universe/`, named after the mod, holding any of the files below (for example `content/mods/universe/haven/systems.json`). The folder name becomes the prefix of the mod's keys (`haven/new-haven`) unless a file sets its own `"mod"`. The name `openprivateer`

belongs to the game's own content and cannot be used for a mod. File names must be written in lower case exactly as shown (`systems.json`, not `Systems.json`); a file with another spelling is not read and the log says so. Files placed directly in `content/mods/universe/` also work, as one more mod without a folder. Several mods can be installed at once: the files directly in the folder load first, then the mod folders in name order, and each kind of file is read for all mods before the next kind, so one mod can use another mod's factions, quadrants and systems. When two mods want the same thing (the same key, the same empty place on the Sector Overview), the one loaded first gets it and the other mod's file is not loaded. Each mod may have its own `filter.json`; what they hide adds up. The Open Universe setting turns all universe mods on or off together.

`content/mods/universe/systems.json` adds systems and jump lines to the Open Universe. The file is JSON5, so comments, trailing commas, unquoted field names, single-quoted strings and hex numbers are allowed, but each field may appear only once per object. Invalid files are rejected as a whole and the log names the file and the field, or the line and column of a typo; the other files still load. Delete a file to turn it off.

`content/mods/universe/sectors.json` adds new sectors to the Open Universe map, each in one of the five empty places of the Sector Overview, with up to four new quadrants that `systems.json` can fill.

`content/mods/universe/filter.json` hides sectors and systems from the Open Universe map. Gemini is always shown. Hidden systems still exist: the filter only changes what the map shows.

`content/mods/universe/factions.json` adds factions with their own map color and name (shown as "Faction" in a selected system's info panel). A system in `systems.json` can name one in its optional `faction` field.

`content/mods/universe/navpoints.json` adds bases, planets, stations, plain nav points (empty space, an asteroid field) and jump points. Added bases can be flown to and landed at. A mod base can reuse the interior of a similar original base while keeping its own location identity. A nav point can be made a no-weapons-fire zone. A mod base can also give its own name for the Communications MFD contact list and its comm messages, and its own type line on the Mission Computer under its name (for example "Carrier" and "Confed Carrier"); without them it shows those of the original base it borrows from, except that in Enhanced and Modern its contact row and comm messages show its own name. That type line is also the base's type in the navigation map's "Bases :" list instead of its category; a type too wide for the list ends in "…".

A base or station can also be a **carrier**: a station from a ship mod stands on the nav point, and the base is aboard it. You see, target and hail it as a ship, and you dock and land on it like at any base: inside, you save, repair, trade and launch with the services the base offers, and you launch next to the carrier. A carrier can have its landing deck off its centre, at the stern for example: the landing zone, the base's contact on the Communications MFD and your launch point are then at that deck rather than at the middle of the ship. As at any base, you land by pressing `Esc` inside its landing zone or sending the deck's docking request (with **Hail To Land** on, you ask first, and the immediate docking request is listed only within 2000 units of the deck; section 4). A carrier stays on its spot when you ram it, though the ram still damages both ships. Docking is open only while the carrier is not hostile to you; fly into its landing zone while it is hostile and the message line shows "Docking refused". If the carrier is destroyed, its base stays closed until you next enter the system, when the carrier is back. A mod can make a carrier's base **destructible**: then destroying the carrier (or, depending on the mod, only your own kill of it) destroys the base for good. It is gone from space, perhaps leaving a wreck you cannot target or land

at, and the navigation map shows its nav point as "<name> (destroyed)". It stays gone in your saved game; some mods bring it back after a number of landings elsewhere, others never. Destroying such a base moves your standing as the mod sets for that base (by default the faction that flies the carrier trusts you less), and it counts as one kill. Only a weapon a mod makes a base killer destroys such a carrier with one shot; a one-shot ship killer only damages it like any gun. A mod can also make a planet destructible: only a planet-killing weapon destroys it, and then the planet and its base are gone from space in the same way. A carrier needs **Ship Mods** on (Enhanced or Modern, single player); without it the carrier's base cannot be docked at, and a game saved aboard it loads at the original base it borrows its interior from.

A mod can also put a station at **bases of the original game**: the station stands where the base was and is what you see, target and land on, while the base inside (its rooms, dealers, missions and comm contact) is the original one. As with a carrier, you can dock only while the station is not hostile to you. A mod can make such a base destructible too; destroyed, it is gone from space, the navigation map's "Bases :" list shows it as "<name> (destroyed)", a carried mission delivering there fails on your next landing, and it always comes back after a number of landings elsewhere (10 unless the mod says otherwise). The bases the story needs are never given a station, so the campaign always plays through. In single player these stations appear in Enhanced or Modern with **Open Universe** and **Ship Mods** on; with either off every base is as in the original game. In a multiplayer session the server's mods decide: every pilot sees the same stations, a base destroyed by anyone is destroyed for everyone (it shows as destroyed to a pilot who arrives later, its wreck if the mod gives one stands in its place, and the server refuses a landing there), and it comes back after the given number of landings by any pilot of the session, or when the server restarts. The server also refuses a landing at a base whose station is hostile to you.

Some carriers and added bases **answer when you ask to land** (a mod gives them a comm profile). Inside the landing zone their docking list on the Communications MFD starts with "Requesting immediate automatic docking", then a request line ("Request landing clearance." or the mod's own words). Send the request line and the base answers: cleared (with the mod's own words, or "Entering automatic landing zone" when the mod gives none), or denied, for example while enemy ships are near it. The answer shows on the message line, or, when the mod gives the base a voice (in your language, where the mod has one per language) and sound effects are on, you hear it spoken instead. Such a base may also show a face of its own on the comm display, which talks while it speaks. Once cleared you land by flying in closer than 1000 units to the deck or by pressing `Esc`; leaving the landing zone cancels the clearance, and a denied request can be sent again at once. With **Hail To Land** on, `Esc` at such a base sends the request first if you are not cleared yet. "Requesting immediate automatic docking" (key 1 on the list) lands at once, whatever the base answers; with **Hail To Land** on it is listed only within 2000 units of the deck, and the request line is key 1 until then. Such a base may also list lines of its own on the Communications MFD; it never answers them, and the page closes when you pick one. A carrier that has turned hostile may tell you so itself when you fly into its landing zone. Such a base may also **greet you as you land** there ("Welcome aboard, sir." or the mod's own words), however you land: during the landing movie and on the landing pad you hear it spoken, when the mod gives it a voice and sound effects are on, or read it for a few seconds on the bar at the foot of the screen. It greets you once per landing, never when it refuses you. In a multiplayer session on a server started with `--open-universe` these bases answer the same way, the server deciding whether you are cleared (for example refusing you while ships hostile to you are near the deck); Hail To Land and `Esc` work as your own settings say.

An added base or station can also have an **interior of its own** from an interior mod (`content/mods/interiors/<mod>/interiors.json`, or a pack's `interiors/` folder): new pictures for its landing pad, concourse, bar or showroom, new rooms, and doors that lead elsewhere. An interior either builds on an original base (every screen the mod does not change is that base's) or is entirely its own, a new kind of base whose doors open the shops of the original base it stands for. Its rooms can move (flickering lights, smoke, figures that idle and turn to look when you point at them, a train that rides off when clicked), play a sound of their own while you are in them, have a barman who tells rumours when you click him, and sell ships from their own "Purchase" spots, as a Ship Dealer's showroom does. People in its rooms can talk to you when you click them, as the original's bar characters do: a mod may give them faces that talk and blink, and voices for their lines (in your language where the mod has one); while such a face speaks with a voice, its mouth moves with the voice. A spoken line plays with Sound F/X on and lasts as long as its voice; its text shows while the debug overlay's **SUBTITLES > Cutscenes** box is on (the default), as for the original's spoken conversations. A line without a voice always shows its text. A click or Space moves on to the next line (and stops its voice), Esc jumps to the question, and Accept or Reject answers it. What you answer can matter later: a character may remember a job you took, pay or charge you, or be gone the next time you come, and doors can appear or close. A character can also hand you cargo, a sealed parcel or trade goods, or take goods from your hold for a price or a favour; contraband handed to you goes into your secret compartment first. This is kept with your saved game. The Commodity Exchange, the guild offices, the Mission Computer, Repair & Upgrade, the Software Booth and the Research Computers are always the original screens. Their Exit takes you back to the room you came from, and so does Escape in Enhanced and Modern (on a landing pad, Escape takes you into the base). A door to a service the base does not offer is not there. A room's sound is a sound effect: it plays with Sound F/X on. A room may also silence the base music while you are in it, so that only its own sound plays; the music comes back when you go to a room that has it. Such a base's sound can carry on into the original screens you open from its rooms (the bar, the Mission Computer and the others), or change to one of their own while they show. Such a base can bring its own music (section 17), for the whole base or for single rooms. Such a base can also change the landing and takeoff movies: you may see your ship approach a picture of the base's own in place of the usual one, hear music of its own over the landing or the launch, or land or launch without a movie at all. Interiors show in Enhanced and Modern with the Open Universe on and **Interior Mods** on (the debug overlay's MODS section, or `InteriorMods=0` in `openprivateer.ini`; on by default), never in Classic; with them off, or without the mod, the base shows the original interior it borrows. In multiplayer they show only on a server started with `--open-universe`, and then they are the server's interior mods, whatever your own settings (you need the same files as the server to join, as for its other mods); everything an interior does, its conversations, the credits and cargo it gives or takes and what its characters remember, stays in your own game and save. The change applies at your next landing or load. A game saved at such a base comes back there with the interior, and with what its characters remember about you (also when the mod is missing for a while: once it is back, so are they). Starting the game with `openprivateer --interiors-report` writes the loaded interiors to the log and the console; the generated `content/mods/interiors/README.txt` has an example, and the asset export (section 25) writes every original base's interior in this format to start from.

An interior mod can also give a **real Gemini base** a new interior: you land at Perry Naval Base, say, and walk new rooms there. One mod can do this for many bases at once, for example give every mining base the same new concourse, each base keeping its own bar, dealer and story. Everything else about the base stays its own: what it buys and sells, its missions, its guilds, and the story. The rooms where the story happens (a character you have to meet, a door the story opens) keep their story, so the campaign plays as before: either the room stays the original, or it is repainted and the story's characters and doors

stand in it where they stand in the original, appearing, answering and opening exactly as there. A mod may still add things of its own to such a room, such as a notice board you can click, and give the bartender new tips beside (or instead of) the usual rumours. Where something added overlaps an original character or door, the original answers your click. It shows under the same settings as the other interiors (Enhanced and Modern, Open Universe and Interior Mods on; never in Classic); otherwise the base shows its original interior. In multiplayer it shows on any server that runs the interior mod, with or without `--open-universe`, whatever your own settings (you need the same files as the server to join). A game saved there is an ordinary save at that base. With several mods changing the same base you see all of their rooms; where two change the same room, the mod later in the load order shows its own, and a mod that rebuilds the whole base shows only its own rooms.

An **NPC mod** (`content/mods/npcs/<mod>/npcs.json`) changes the people of the original bars, at the bases it chooses (all mining bases, say, or one base by name), without changing any room and without the Open Universe. In such a bar the bartender may tell new tips beside (or instead of) the usual rumours, answer with a line of his own where he would brush you off, and show another face while he talks; up to four patrons may stand in the bar (figures like the original's, or people the mod has drawn), some only on some landings, each telling rumours or the mod's own talk when you point at them and click (the hover text names them, "Talk to Patron" unless the mod names them otherwise); and the stranger with a job may come more or less often, or mostly offer some kinds of job. The story's characters stay as they are: nothing stands over them. The mod's own talk may come with a face of the mod's that talks and blinks and with voices (a woman's where the figure you clicked is a woman), played as in the rooms of a base mod: a spoken line plays with Sound F/X on, lasts as long as its voice and shows its text while subtitles are on; the usual rumours keep their usual voices. Leaving and re-entering the bar, or saving and loading there, shows the same patrons. NPC mods apply in Enhanced and Modern with **NPC Mods** on (the debug overlay's MODS section, or `NpcMods=0` in `openprivateer.ini` to turn them off; on by default), in the original bars only (not in a bar an interior mod shows), never in Classic and never in multiplayer; the stranger's settings also need **Fixers** on. The change applies at your next landing.

An NPC mod can also change the pilots you meet in space at random (not the pilots of a mission or the story): a merchant, a pirate or the pilot of a particular ship may answer under a name of its own (on the Communications display and in its messages), show another face on the comm display, say other words when it hails you, answers you, is badly hit, gloats or scores a kill, sometimes with a recorded voice of the mod (a line the mod voices only in another language shows its words instead), and your own menu choices to it may read differently while meaning the same. Who attacks you, when a pilot speaks and how your words change its mood stay exactly as in the original. This applies with **NPC Mods** on, in Enhanced and Modern, outside multiplayer, from your next launch or jump.

In the systems a universe mod adds, its own factions may patrol: their pilots answer under the faction's name (for example "Haven Wardens" instead of "Militia") on the Communications display and in their messages, but behave exactly as the kind of pilot the mod bases them on, militia or pirates for example: they are friendly or hostile to you on the same terms, and fighting them counts the same. `openprivateer --npcs-report` writes what the loaded NPC mods do to the log and the console; the generated `content/mods/npcs/README.txt` has an example.

`content/mods/universe/zones.json` adds asteroid fields to systems. A zone can use a position or a nav point to sit on, a size and a density (`"sparse"`, `"medium"`, `"dense"` or 1 to 7 rocks).

`content/mods/universe/encounters.json` adds random encounters: for a nav point or zone, a list of possible groups with their chance, number of ships, faction, ship type, skill and morale. Each time you enter the system one group per table is picked, and its ships appear when you come near that point, as in the original systems. Several tables, also from different mods, can share a nav point or zone: each picks its own group, all of their ships appear there together, and the log notes each extra table. A table can also name one of the original Gemini systems and one of its nav points as the Navigation Map shows it (`"nav": "Nav 1"`): its ships appear there beside the original's, in Enhanced or Modern with Open Universe on and outside multiplayer, and in a multiplayer session whose server runs the mod; Classic keeps the original's encounters alone. Such a table can also take away or change the original's own groups at that nav point (`replace`, `remove`, `modify`): a group that was taken out keeps its chance, so when it comes up that nav point has fewer ships or none. Groups that run a script, such as the militia's contraband searches and their escorts, always stay. `openprivateer --encounters-report` writes every original group of every Gemini system with its record number to the log and the console, and what the loaded mods do to it; `--encounters-report=Humboldt/Troy` lists one system. The game starts normally afterwards.

The generated `content/mods/universe/README.txt` contains the current formats and built-in examples.

Mod packs, load order and disabling mods

Every kind of mod (universe, weapons, items, start, mission, ship, interior and NPC mods) can be installed in two ways, which work the same:

- one folder per kind, as described in each section here, for example `content/mods/weapons/rebalance/weapons.json`;
- a **mod pack**: one folder holding all of a mod's parts, in `content/mods/packs/`, for example `content/mods/packs/rebalance/weapons/weapons.json` and `content/mods/packs/rebalance/items/shipstuff.json`. A pack is the easiest way to share a mod: zip its folder.

The folder name is the mod's name. If a pack and a per-kind folder have the same name, the pack is used. A mod may come with a `mod.json` that gives its full name, version, author and description, and which other mods it needs or does not work with. A mod that needs a mod you do not have, or that does not work with a mod loaded before it, is not loaded, and the log says why.

Mods load in alphabetical order of their names (a mod after the mods it needs). To choose the order yourself, or to disable a mod without deleting it, use the launcher's **Mods** section (below), or write `mods.json` in the settings folder (next to `openprivateer.ini`) by hand:

```
{ "version": 1, "order": ["base-pack", "rebalance"], "disabled": ["test-mod"] }
```

The mods listed under `order` load in that order. A mod you install later, which `order` does not list yet, is new: it goes after them and stays off until you tick it in the launcher or add it to `order`. The mods listed under `disabled` stay disabled. A `mods.json` without an `order` list only disables the mods under `disabled`; every other mod is on. Without `mods.json` every mod is on. Changes apply the next time you start the game. Start the game with `--mods-report` to see which mods were found, their order, which are disabled or not loaded and why; the same is written to `logs/mods_status.json` at every start. The generated `content/mods/README.txt` sums this up.

Managing mods in the launcher

The launcher's **Mods** section lists every installed mod in load order. Its title says how many are enabled. **Mods are disabled in Classic mode**: they apply only in Enhanced and Modern. A line under it lists the **Enabled features** (Weapons, Items, Missions, Ships, Interiors, NPCs, New Game, Open Universe), set with the debug overlay's switches.

- The first row, **Built-in Universe**, is OpenPrivateer's own Open Universe content. Its box is the same setting as the debug overlay's **Built-in Content**.
- **Loose universe files** (the files directly in `content/mods/universe/`) are always enabled and always load first.
- Every other mod has a box to enable or disable it, and **Up**, **Down**, **Top** and **Bottom** to change its place in the load order. A mod later in the order wins where two mods change the same thing.
- Until you first change the list, every mod is enabled. After that, a mod you install is new: it appears at the end, unticked, with **New: added at the end of the order, off. Tick it to load it.** It does not load until you tick it. The line goes away with your next change to the list; a new mod you leave unticked then stays off.
- **Version** is the mod's version from its `mod.json`; **Hash** is the first 8 characters of its content hash, which changes whenever one of the mod's files changes, even when the version does not (hover over it for the whole hash).
- Click a mod's name for its details: author, description, web address, where its files are, its content hash, files in its folder that are not part of the mod, and the warnings and errors of the last start.
- **Status** shows what the last game launch did with the mod: **loaded**, **partly loaded** (one of its files was rejected), **not loaded**, **new since the last start**, or **Not checked yet** before the first launch. Until then the section says **Mod status will appear after the next game launch.**
- A mod that needs another mod, or does not work with one, shows the problem under its row with a button that fixes it, for example **Enable base-pack** or **Move after base-pack**. A mod whose `mod.json` lists files that differ from the files on disk says which ones.
- **New Game Mod** picks the start mod used when starting a new game; **(automatic)** uses the only one when there is exactly one.
- **Open Mods Folder** opens `content/mods`; **Refresh** reads the folders again after you added or removed a mod.

Every change is saved at once (the order and the disabled mods in `mods.json`, the start mod and Built-in Universe in `openprivateer.ini`) and takes effect on the next game launch. Some mod settings can also be changed in-game, in the debug overlay: Weapon Mods and Item Mods at once, Mission Mods at the next mission board, Ship Mods the next time you enter a star system, Interior Mods at your next landing or load, NPC Mods at your next landing, Init Mods and Init Mod at the next New Game. A mod list that cannot be read is shown with its error and a button that sets it aside as `mods.json.bak`; until then nothing in the list can be changed.

In a running game, the **This run** list in the debug overlay's MODS section (click **This run** to open it) shows what this start did with each mod and how many errors and warnings it had, and **Mod manager (restarts)** restarts OpenPrivateer into the launcher with the Mods section open. Progress since the last save is lost.

The overlay's right-hand pane is the **Mods log**: every message the mods gave since the game started, warnings in orange and errors in red, long lines wrapped so they can be read in full. Above it are the

number of warnings and errors, **Problems only** to show just those, and **All log** to show the game's whole log instead.

Reloading mods while you play. Below that list, **Reload ship mods** reads your ship mods again without restarting, and **Reload interior mods** does the same for base interiors. This is meant for mod authors trying out changes:

- Docked, at a base screen or in a menu, **Reload ship mods** applies every change to your ship mods. Ships you meet in space change from the next time you enter a star system. If the ship you own loses its mod hull in the reload, it becomes the stock ship that hull was based on, as when a save is loaded without the mod.
- In flight, **Reload ship mods** works only while the game is paused (P). It applies the cockpit, launch pad and Repair & Upgrade pictures and the Weapons MFD at once. Flight handling and hardpoints change at your next launch. A change that adds, removes or renames a ship, or changes a ship's mounts, hold, cockpit file or dealer details, is refused in flight: land and reload again.
- **Reload interior mods** works when you are not in flight. Docked at a mod interior, the visit starts again on the scene you were on, with the new pictures, hotspots, music and sounds.
- A reload waits while a movie, a conversation or a click script is running, and runs as soon as it ends. In flight it waits until you pause.
- A reload is refused in a multiplayer session or while connected to a server. It is also refused when the reload would break something only a restart can set up again. A ship that an encounter table or a carrier base uses is one example; an interior that a nav point names is another. The overlay line under the buttons and the Mods log say why.
- If a mod no longer loads (for example after a typing mistake in its file), nothing changes and the error is shown.
- **Auto reload** (off every time the game starts) reloads by itself about a second after you save a change to a ship or interior mod file.
- Classic plays without mods, so the buttons are disabled there.

Reporting a bug with mods installed: name each mod's version and hash. The log in `logs/` lists every mod at every start, one line each, for example `mod 2. tayla, version 0.C 2026-10-21, hash 1a2b3c4d, loaded, layers ships,universe`; the launcher's Mods section shows the same version and hash. Two copies of a mod with the same hash have the same files. When a multiplayer join is refused because the mods differ, the log names your copy's version and hash and the server's version and hash for each mod that differs; the launcher's **Check Server** shows the same before you join.

Weapon mods (experimental)

For every kind of mod (weapons, items, start, mission and ship mods below, and the universe and interior mods above), a mod folder may not be named `openprivateer` or `packs`: those names belong to the game, and such a folder is skipped. The mod's file must be named in lower case exactly as shown (`weapons.json`, not `Weapons.json`); a file with another spelling is not read and the log says so. Universe mods are the exception: their files are found in any letter case. Pictures and sounds a mod names inside its file are found by the name written there. A mod's file may be up to 4 MiB; each picture or sound up to 8 MiB, and each picture up to 4096 pixels wide or high and 2048 by 2048 pixels in all. A larger file is not used, and the log names it and the limit. A picture whose name ends in `@2x.png`, `@3x.png` or `@4x.png` is a high-resolution picture (a Ship Dealer or Software Booth item, a base scene's

background, a cockpit view): in Enhanced and Modern it shows its full detail at the matching Render Scale.

A weapon mod changes the guns. OpenPrivateer comes with none: you add them yourself, and without one the guns are the original's. Each mod is its own folder with a `weapons.json` in it, for example `content/mods/weapons/rebalance/weapons.json`; mods load in alphabetical order of their folder names, and a later mod's value for the same gun and stat wins. A mod can change a gun's damage, time between shots, energy per shot, bolt speed and lifetime (and with them its range), its wear limit and its three-letter name on the Weapons display; a changed original gun changes for every ship that carries it, computer-flown ships included. Named stat sets let several guns share one baseline and adjust it. A weapon mod can also set a gun's name, prices, the ships the Ship Dealer sells it for and the bases that sell it; where a Ship Dealer mod (below) sets the same thing for that gun, the weapon mod's value is used and the log says so. The `README.txt` in `content/mods/weapons/` shows a first mod, and the modding documentation that comes with the source describes every field with worked examples.

A mod can also add new guns. Each new gun is based on an existing one, and the Ship Dealer sells it under its own name (in the selected language when the mod gives one) and price, after the original items, on the ships the mod lists; buying, selling, repairing and fitting it on a turret work as for any gun. A new gun based on the Steltek gun is kept like the Steltek gun unless the mod says otherwise: it takes no wear and cannot be sold. Ships you meet in space do not carry new guns; your hired wingmen fire them on your own ships. A mod can give a gun its own pictures (the bolt, the Weapons display icon, the dealer's icon, its picture on each ship's hardpoints, turret bays included), its own lead-pip colour and its own firing sound. Pictures are PNG files, drawn in the game's own colours; sounds are WAV or VOC files. The Laser's bolt and a beam weapon's beam (below) are drawn as lines, and a mod can give each gun's line its own colour and thickness, and the Laser's bolt its own length; only the look changes, not where the shot hits. Every ship firing that gun shows it, other pilots' ships in a multiplayer session included.

A mod gun can be a charge weapon: hold the fire button to charge it and release to fire one shot, stronger the longer you charged. Charging uses energy while you hold; releasing too early, switching views or running out of energy loses the charge. Some charge weapons fire on a quick press as well: let go before the minimum charge and the weapon finishes charging to its minimum by itself and fires a minimum shot; keep holding and it charges on as usual. A small bar above the Weapons display's `G:` line shows the charge. A weapon with a charged sound plays it the moment the charge is full, whether it then waits for you to let go or fires by itself. Ships flown by the computer fire charge weapons too: they charge while they have a target in range and fire when the charge is full and the gun points at it. A turret of your own ship that you are not sitting in does not fire a charge weapon. A charge can take up to 20 seconds, and longer when a mod allows it.

A mod weapon can be **lethal**: one hit that reaches a ship destroys it, whatever its shields and armour, and that includes your ship. A phase shield (see "Ship mods" below) stops a lethal hit unless the weapon can go through phase shields. An enemy capital ship can carry such a gun, a cannon that charges for a long time and then fires one huge shot that destroys any ship it hits. When a computer-flown ship within about 5000 units of you is charging a weapon, you hear the weapon's charging sound, if the mod gives it one; the shot sounds like any other ship's shot. Stay out of its gun's arc, or out of its range, while it charges.

A weapon mod can make your torpedoes **phase torpedoes**, the one weapon of a fighter that gets through a capital ship's phase shield. To use one:

1. Select the torpedo with `W`, target the phase-shielded ship with `T` and lock it with `L`.
2. Keep the ship in front of your nose, inside what the front view shows, within the range the mod sets (12000 unless it says otherwise), and keep the nose steady: small corrections are fine and rolling is fine, but a hard turn of more than half a second interrupts the decode. While the torpedo decodes, a red reticle (an oval ring with a cross through it) circles the target, starting well away from it and closing in as the decode builds (20 seconds unless the mod says otherwise); rolling or pitching your ship turns it around the target. An interrupt makes the reticle disappear, and the decode starts again (or, with some mods, waits where it was).
3. When the reticle reaches the middle of the target, the reticle and the target box flash and the missile-lock diamond appears (with a sound, if the mod brings one). From then on you can manoeuvre and close in, as long as the ship stays in front of your nose: turning away from it loses the decode (the reticle disappears), and so do changing or unlocking the target, selecting another weapon with `W`, taking a turret's seat (`F5` / `F6`), jumping and landing. Looking left, right or back, or through an outside camera, does not matter: only where your ship points counts.
4. Press `Enter` at the range you like (about 3000 is typical): the torpedo goes through that ship's phase shield, and with most mods it homes on the ship. Each torpedo needs its own decode. Launching a phase torpedo releases the target lock: the ship stays your target, and you lock it again with `L` for the next decode.

The typical attack: start the decode at long range, outside the capital ship's gun range, then fly in toward it without turning away and fire at close range.

A torpedo will not launch at a phase-shielded ship before it has decoded ("Torpedo not decoded" on the message line). At a fighter, or any ship without a raised phase shield, the torpedo launches and flies as an ordinary torpedo. A mod can give a decoded torpedo its own speed and flight time, often slower than an ordinary torpedo and much longer, and its own damage, so a decoded torpedo can hit far harder than an ordinary one while the torpedoes of other ships stay ordinary. A mod can also sell the phase torpedo as a round of its own, beside the ordinary Proton Torpedo (new rounds are described further down). Then only the phase torpedo decodes: select it with `W` to start a decode. With the Proton Torpedo selected nothing decodes, and a Proton Torpedo fired at a phase-shielded ship is stopped by its shield. Switching from one decoding torpedo to another starts the decode again from the beginning. With some mods one decode lets a whole salvo of phase torpedoes through (see salvos below). Phase torpedoes work in multiplayer too (see "Multiplayer"). Computer-flown ships that carry them get through a phase shield too: with most mods their torpedo goes straight through any raised phase shield, yours included if your ship has one, as Wing Commander 2's computer pilots do; with some mods they decode first, holding your ship steady in front of their nose for the decode, so a hard turn keeps their torpedo out.

A weapon mod can also give a **missile** the phase decode. You decode with the missile selected exactly as with a phase torpedo (target, `L`, hold the ship in front of your nose until the reticle closes in), and the next missile goes through that ship's phase shield and homes on it, even a Dumb Fire. Fired at a phase-shielded ship before the decode it does not launch ("Missile not decoded" on the message line); at any other ship it is an ordinary missile and keeps your target lock. Only a missile that went through on a decode releases the lock. Computer-flown ships use such missiles against phase-shielded ships as they use phase torpedoes, and they work in multiplayer too.

A weapon mod can also make a missile's lock **take a moment**, as in Wing Commander 2 (Heat Seeker, Image Recognition and Friend or Foe missiles; a Dumb Fire never locks). With such a missile selected

and its usual lock conditions met (the Heat Seeker behind the target, the target locked with `L` for the other two), keep the target in front of your nose: the same red reticle circles it and closes in, quicker than a torpedo's decode (under a second to a few seconds, as the mod sets). When it reaches the middle of the target, the reticle and the target box flash and the missile-lock diamond appears: the missile is locked. Losing the conditions or turning away from the target breaks the lock (a mod may allow a short slip, or keep the progress), and so do changing the target or the missile and taking a turret's seat; looking around, as for the torpedo, does not. Every missile you launch uses the lock up, so the next one locks again from the start. A missile fired before it is locked flies straight without homing; with some mods it does not launch at all and the message line says "Need lock". Timed missile locks work in multiplayer too; computer-flown ships lock the way the original game's do.

A weapon mod can also make **guided torpedoes**: torpedoes that home on their target like a missile. You select, buy and fire them like any torpedo. To make one home, target a ship with `T` and lock it with `L`: within the mod's lock range the missile-lock diamond appears, and a torpedo launched then follows the ship. A guided torpedo fired without a lock flies straight like an ordinary torpedo. It stays a torpedo in every other way: it leaves the launcher at full speed and the missile camera (`F7`) does not follow it. Computer-flown ships home theirs on their target too, and a countermeasure decoy can lure a guided torpedo away. Guided torpedoes work in multiplayer too.

A mod can also give a guided torpedo a **timed lock**, as the timed missile locks above: after `L` , keep the target in front of your nose while the red reticle closes in, and launch once the diamond appears. Fired earlier, the torpedo flies straight, or with some mods does not launch and the message line says "Need lock". If such a torpedo also decodes phase shields, it decodes at a ship whose phase shield is up (the decode is its lock there) and takes the timed lock at every other target.

A weapon mod can also give a missile or the torpedo **lock sounds**: a sound that repeats while a lock builds, one when the missile locks (for a missile that locks at once, when the lock diamond appears) and one when the lock is lost. Launching a missile uses its lock up without the "lost" sound. The original game has no lock sounds, so without such a mod a missile locks silently. Lock sounds also play in multiplayer.

A weapon mod can also change the missiles, the torpedo and the two launchers: their damage, how long a round flies, the range at which the missile-lock diamond appears, how quickly a launcher reloads, and how a round flies: its top speed, how fast it gets there, and how sharply a homing missile can turn. It can also rename a missile or the torpedo on the Weapons display (up to 8 characters); the name the Ship Dealer lists comes from an items mod instead. The change applies to every ship that carries them. New missiles and torpedoes of their own are below.

A weapon mod can also give a missile or the torpedo a **limited burn time**: once its fuel is out it can no longer steer or speed up and flies straight on at the speed it had until it expires, so a missile fired from too far away may coast past its target. The missile-lock warning goes out once an incoming missile's fuel is spent. A mod can also make a missile leave the launcher already moving, at a set speed along its nose, with your ship's forward speed or your ship's whole velocity added. Computer-flown ships fire such missiles only at targets they can reach under power. These missiles work the same in multiplayer.

A weapon mod can also change the course a homing missile takes: it can aim ahead of a moving target, at where the target will be (**lead**), so it meets a crossing ship from the side and sooner, even when it cannot turn sharply; or come in behind it (**lag**), swinging onto the target's tail and closing from behind, which takes longer and can miss a fast ship crossing in front of it. A missile with neither steers straight

at where its target is. Computer-flown ships fire these missiles too, and they work the same in multiplayer.

A weapon mod can also give a missile a **narrow seeker**: it sees its target only within a cone in front of it. A missile fired at a target outside that cone flies straight ahead unguided, and one whose target slips out of the cone in flight, for example when the target turns hard across its nose, stops homing and flies straight on; the missile-lock warning goes out when that happens. Some such missiles pick the same target up again when it comes back in front of them; they never switch to another ship. Computer-flown ships fire these missiles too, and they work the same in multiplayer, where every pilot sees a missile stop and start homing at the same moment.

A weapon mod can also make a missile **guided by your ship**: it homes only while you keep its target selected and locked with `L` and in front of you, within a cone around your nose. Turn away or release the lock and the missile flies straight on; bring the target back in front of you with the lock on and it homes again. Select another target, or lose your ship, and the missile flies on unguided for good. The missile-lock warning shows only while such a missile is homing. Computer-flown ships fire these missiles too and guide them at the ship they are fighting, but they keep maneuvering while the missile flies, so theirs often go astray. They work the same in multiplayer: the server guides the missile from the target you have selected and locked.

A weapon mod can also give a missile or the torpedo a **warhead**: it explodes when it passes close to its target, or where it hits, and the blast hurts every ship near it, less the farther away a ship is. Your wingmen and other bystanders close to the blast are hurt too, and so is any ship nearby that you would rather not anger: a blast counts as your shot. Your own ship is spared by most such missiles, but not by all of them. Some explode at the end of their flight even when they hit nothing. Computer-flown ships fire these missiles too, and they work the same in multiplayer, where the blast can hurt other pilots near it and every pilot sees and hears the explosion.

A weapon mod can give a gun such a warhead too: its shots explode when they pass close to an enemy ship, so a near miss still does damage, and the blast hurts the ships around it as a missile's does. A ship takes at most one blast from each shot, however many pieces of a scattered shot explode around it, and a piece that hits a ship directly counts as that ship's blast. Most such shots do not explode near friendly ships or your own, but their blasts can still catch a wingman next to an enemy. Computer-flown ships carrying such a gun open fire from a little farther away. These guns work the same in multiplayer, where the blast can hurt other pilots near it and every pilot sees the explosion.

A weapon mod can also make a gun's shot burst into a spray of smaller shots: where it explodes near a ship, where it hits one, or where its flight ends, it releases a handful of shots of the same or another gun in a cone, a ring, a fan or a straight stream, which fly on and hit whatever they meet. They are your shots, so they never hit your own ship, and the pieces a shot releases where it hits a ship never hit that ship. Computer-flown ships fire such guns too. When a fight is very crowded with shots, some of a burst's pieces may not appear. These guns work the same in multiplayer: every pilot sees the pieces fly from where the shot burst, and the pieces make no firing sound of their own.

A weapon mod can also make a gun, a missile or the torpedo hit shields and armour differently: part or all of each hit can go past the shields straight to the armour (a torpedo that ignores shields), a gun can drain shields faster or slower than its damage says, or do more or less to armour. A weapon that cannot get through a raised shield, or that never damages armour, is possible too: an ion gun that only strips

shields cannot destroy a ship on its own. Every ship that carries the weapon hits this way, computer-flown ships included.

A mod gun can also be a beam weapon: while you hold the fire button it fires a continuous beam that hits whatever it points at instantly, up to its range, and keeps doing damage as long as it stays on the target. The beam uses energy while it fires and pauses when the energy runs out. A beam can overheat: held too long, it switches off for a while and then starts again. The bar above the Weapons display's `G:` line shows its heat, and while it is overheated, the time left to cool down. The lead pip sits on the target while it is within the beam's range. Ships flown by the computer and turrets you are not sitting in do not fire beam weapons.

Some beam weapons fire in timed pulses instead: one press of the fire button starts a pulse, which keeps firing for its full time even after you let go, and then the beam rests for a while before it can fire again; holding the button fires the next pulse as soon as the rest is over. A charged beam has to be charged first: hold the fire button to charge it and release it to fire the pulse (some fire by themselves once fully charged); the longer you charged, the stronger and longer the pulse. Some charged beams also fire on a quick press: they finish charging to their minimum by themselves and fire a short, weak pulse. Releasing too early, switching views or running out of energy loses the charge, and a pulse stops early when the energy or the rounds run out, when the beam overheats or when you switch views. The bar above the Weapons display's `G:` line fills while the beam charges, empties as the pulse fires, and shows the rest in the damaged colour.

A mod gun can also overheat or fire in bursts. An overheating gun gets hotter with every shot and cools once it has rested; when it gets too hot it stops firing for a while and then starts again, cold. A burst gun fires a few shots and then pauses before it fires again; some burst guns fire the whole burst from one press of the fire button. The bar above the Weapons display's `G:` line shows the hottest such gun's heat (for a burst gun, how much of the burst is used), and while a gun is held off, the time left in the damaged colour. A mod can give the gun a sound for the moment it locks. Every ship that carries such a gun is held by it the same way, computer-flown ships included, and these guns work in multiplayer too.

A mod gun can also need ammunition. All guns of that kind on your ship share one supply of rounds, and each gun you fit adds its own magazine to it; a gun comes with a full magazine. The Weapons display's `G:` line shows the rounds left after the gun's name, `ACn(300)`, and `(0)` when they are gone; when the line is too long for the display it shortens, and with four different guns on a Centurion the count can be left off. A gun out of rounds does not fire and uses no energy, and a mod can give it a click for the moment you pull the trigger on it. Buy more at a Ship Dealer: while your ship carries the gun, the Buy list has a row for its rounds, each purchase adds a pack up to what the gun's magazines hold (NO ROOM ON SHIP when they are full), and the last purchase that fills them costs only its share. Rounds cannot be sold: selling the gun gives up the rounds it held. Computer-flown ships carrying such a gun start with a full magazine and never get more; your hired wingmen use the rounds of the ship they fly, and you restock that ship when you fly it yourself. The rounds left are kept in your saves. In a multiplayer session the server keeps count of your rounds: the Weapons display shows its count, and a gun stops firing when the server says it is empty. Each launch takes the rounds your ship carries, up to what its magazines hold, so rounds bought at a Ship Dealer during the session come with you.

A save stores a new gun as the original gun it is based on, and the `.openpriv.json` file next to the save remembers which new gun it is. A save loaded without the mod has that original gun. In Classic, or with **Weapon Mods** off, the ship carries the original gun too, and the new gun comes back when the mods are in effect again.

A mod can also add new missiles and torpedoes, each based on one of the originals: a phase torpedo beside the Proton Torpedo, say. The Ship Dealer sells each one as ammunition of its own, after the original rounds, under its own name and price. It is fired from the launcher of its kind (a torpedo from a Torpedo Launcher, a missile from a Missile Launcher) and shares that launcher's room with the original rounds: a Torpedo Launcher holds 20 torpedoes, Proton Torpedoes and phase torpedoes together. `W` reaches the new rounds after the original ones, and the Weapons display shows each by its own short name and count. Selling a launcher sells the rounds the other launchers cannot carry, the cheapest first. Ships you meet in space fire the new rounds only when a ships mod fits them; a wingman flying one of your own ships fires the new rounds it carries, a torpedo at a capital ship and a missile that can lock first, the cheaper round before the dearer one. A save stores them as the original round each is based on, as far as the original game can load them, and the `.openpriv.json` file keeps their own counts. In Classic, or with **Weapon Mods** off, the rounds stay on board but cannot be selected or bought, and they come back when the mods are in effect again.

A mod can also add new launchers, each based on the Missile or the Torpedo Launcher: a heavy launcher that fires only phase torpedoes, say, or a missile rack that fires backwards. The Ship Dealer sells each one after the original launchers, under its own name and price, on the hulls and launcher positions the mod allows; it is sold and repaired like an original launcher, and trading the ship in credits it at its own price. A launcher of its own kind of round fires only those rounds, and the Ship Dealer sells those rounds only while the ship carries such a launcher, as many as its launchers hold (each new launcher holds the number the mod gives it). A new launcher can have its own reload time, its own icon on the Weapons display and fire behind, above or below the ship. A launcher can also fire a **salvo**: one press of `Enter` launches several rounds from it, a moment apart, and it reloads after the last one. Each round uses one from your hold; a homing missile salvo fired with a lock homes, every missile on its own. Selecting another round with `W`, taking a turret's seat or flying into a no-fire zone stops the rest of the salvo, and so does running out of rounds. A save stores it as the original launcher it is based on, and the `.openpriv.json` file remembers which launcher it is. In Classic, or with **Weapon Mods** off, the ship carries the original launcher in its place and its own rounds stay on board, unused, until the mods are in effect again.

A mod can also add **countermeasures**: chaff, flares or decoys, and a **dispenser** that releases them. Press `Q` in flight (Enhanced and Modern; rebindable as **Release countermeasures** on the controls screen) and the next dispenser of your ship releases a burst of decoys behind you, whatever `W` has selected and from any view. A missile that homes on you and passes near a decoy may chase the decoy instead: the missile warning goes off when it does. How likely that is depends on the decoy and the missile (a Heat Seeker is fooled most easily, an Image Recognition missile less, a Friend or Foe missile only by the decoys of its enemies); a Dumb Fire missile and a torpedo are never fooled. A missile that reaches the decoy explodes there, so release them early, before it is close. The Weapons display shows how many decoys you have left, as an icon with a number in a top corner of the page. The Ship Dealer sells a dispenser like a launcher (one inside the ship that needs no launcher position, or one on a launcher position, as the mod has it) and the decoys like ammunition; selling the dispenser sells the decoys it held. A wingman flying one of your own ships with a dispenser uses it too, and so does a ship you meet in space whose ships mod gives it one. A save keeps your dispensers and decoys in the `.openpriv.json` file; a save loaded without the mod has neither. In Classic, or with **Weapon Mods** off, they stay on board, unused.

A mod can also add **mines** and a **mine layer** to lay them. The Ship Dealer sells the mine layer like a launcher and the mines like ammunition. Select the mines with `W` like any missile (the Weapons display

reads, for example, `Mine (8)`, or `No Mines` when you have none left) and press Enter: a mine drops behind your ship. It needs no target and no lock. A mine is harmless for its first moments, so you can fly clear of it; after that an enemy ship that comes close sets it off, and its blast hurts the ships near it. Usually your own mines do not go off for you and their blast spares you, but a mod can make them dangerous to everyone, so do not linger near them. Each ship can keep only a few mines out at once: laying one more sets off your oldest in a small, harmless burst. Enemies and wingmen whose ships carry a mine layer drop mines behind them when a ship chases them. Some mods add **smart mines**: once armed, a smart mine that sees an enemy nearby flies at it like a missile, and your missile warning sounds when one comes for you. A decoy can lure it away, as it lures a missile. A save keeps your mine layer and mines in the `.openpriv.json` file (the mine layer counts as a Missile Launcher in the save itself); a save loaded without the mod has a Missile Launcher and no mines. In Classic, or with **Weapon Mods** off, the mine layer works as a Missile Launcher and the mines stay on board, unused.

A mod can also add **cluster missiles** and torpedoes that split: you buy, select and launch them like any other missile or torpedo, but when one bursts near its target, hits, or reaches the end of its flight, it scatters a spray of gun bolts that fly on and hit whatever is in their path. The bolts are yours: they never hit you. A cluster missile fooled by a decoy scatters its bolts at the decoy, and one shot down by flak scatters nothing. Enemies and wingmen whose ships carry them launch them too, so give an incoming one room. In Classic, or with **Weapon Mods** off, they fly as the plain missile or torpedo they are based on.

A mod can also add **flak**: guns whose shells burst near their target, as on the capital ships of Wing Commander 2. A capital ship's flak turret may also burst near you with no shell at all; those bursts land farther off the faster you fly, so a slow or stopped ship takes the most flak. A burst hurts every ship near it, yours included, and when the mod makes missiles and torpedoes able to be shot down, it also destroys those near it: they vanish in a small explosion without doing any damage. Ordinary gun fire shoots them down too: a bolt that meets such a missile or torpedo (or a mine or a decoy the mod makes so) damages it, and enough hits destroy it. A capital ship or station with flak turrets aims them at the missiles and torpedoes launched at it, so your torpedo may never reach it; launch from closer, or several at once. A flak gun you buy, when the mod sells one, destroys such missiles and torpedoes near its bursts too, so a burst between you and an incoming missile can stop it; if the mod gives its shells a fuse for missiles, a shell that passes close to an incoming missile or torpedo bursts by itself. Your own bursts normally spare your own missiles and torpedoes.

Weapon mods apply only in Enhanced and Modern; Classic always uses the original guns. In a multiplayer session the server's weapon mods apply instead of yours (see Multiplayer). The **Weapon Mods** box in the debug overlay's MODS section (or `WeaponMods=0` in `openprivateer.ini`) turns them off and on, and the change takes effect at once, also in flight. With no mod folder, nothing changes.

A file with a mistake is not loaded at all, and the log names the field and the reason. While any weapon mod is installed, the log also lists the guns in effect with their derived figures (shots per second, damage and energy per second, damage per energy, range). Starting the game with `openprivateer --weapons-report` writes that list to the log even with no mod installed, and to the console when there is one; the game then starts as usual. Use plain ASCII for names: the game's fonts have no other characters. The generated `content/mods/weapons/README.txt` has the format and an example.

Ship Dealer mods (experimental)

A Ship Dealer mod changes what the Ship Dealer offers. OpenPrivateer comes with none: without one the dealer is the original's. Each mod is its own folder with a `shipstuff.json` in it, for example `content/mods/items/prices/shipstuff.json`; mods load in alphabetical order of their folder names, and a later mod's value wins.

A mod changes the dealer's existing items and can add new ones: shield generators, engine upgrades, armour, ECM packages, repair droids, the Righteous Fire improvers and Cargo Expansions of its own (new guns come from weapon mods). For each item it can set:

- its name, in the selected language when the mod gives one;
- what it costs to buy, what the dealer pays when you sell it, and what a repair costs;
- its picture in the dealer's window, for Buy and Sell and for Repair;
- the ships it is offered for, which can add a ship the original does not sell it for (for example the level 5 engine for the Centurion) or remove one;
- that it is not for sale at all;
- the bases that sell it: by base name, base type (pleasure, refinery, mining, agricultural, pirate and so on), system, quadrant, or the factions that fly in the system.

A mod can also make the dealer of some bases a **repair facility** (by the same kinds of base choice): there Repair & Upgrade sells nothing and shows **REPAIR ONLY** in Buy mode, but buys back and repairs as usual, and opens on Repair (or Sell when nothing needs repairing). A service plan does not re-arm at such a base.

A mod can also change the in-flight **fuel tender** (see "Refuelling in flight"): its price, how near a hostile may be, how long the refuel takes, when the tender arrives, what counts as a full stop and how long it waits, or switch it off.

What a mod keeps off the dealer's Buy list never leaves your ship: you can always sell and repair what you carry, at any base. Trading in your ship for a new one values your equipment at the prices the dealer pays for it. An engine level bought for a ship the original does not offer it on works like any engine: it sets the ship's energy (capacity and how fast it refills), is kept in the save, and can be sold and repaired.

A mod can also change what shield generators and engines do: how much a shield holds on each side, how fast it recharges, how much energy it draws and how hard hits strike it; and how much energy an engine level holds and how fast it refills, the same for every ship or different for each type of ship. It can change how much armour each armour type gives, how often an ECM package makes a missile lose its lock, how fast a repair droid works, and how strong the Righteous Fire improvers are. These changes take effect the next time you launch, for your own ships only: ships you meet in space keep their original equipment. Damaged armour keeps its share of damage when a mod changes how much the armour holds, so repairs cost the same. Your save keeps the original equipment, so it loads normally without the mod and in Classic, where everything works as in the original.

New items. A new item appears in the dealer's Buy list with its own name, picture and prices, and you buy, sell and repair it like any item. Each one is made like a stock item (for example a "Shield Mk IV" like Shield Generator 4) and takes that item's place on your ship: a ship with the Shield Mk IV has no room for another shield generator until you sell it. It works with its own strength from your next launch, a trade-in values it at the price the dealer pays for it, and a pilot you hire flies your ship with it. Your save

records the stock item it is made like, with the new item noted beside it, so the save loads without the mod and in the original game as a ship with that stock item. In Classic, or with Item Mods off, your ship carries the stock item and the dealer shows only the original items; the new item is back when the mods are in effect again. If you load a save whose new item's mod is no longer installed (or no longer makes the item the same kind of stock item), your ship keeps the stock item and the notification center tells you which items were replaced; install the mod again and reload before saving if you want them back. In a multiplayer session the server's new items are sold, and a new item you fit works on the server as it does alone (see Multiplayer).

A Ship Dealer mod can also make the Cargo Expansion hold more. The bigger hold counts at once, everywhere you load cargo: buying at the Commodity Exchange, mission cargo, the tractor beam, buying a new ship and switching ships. Your save still records an ordinary Cargo Expansion. Without the mod (or in Classic, or with Item Mods off) the hold is the original size again: cargo already aboard stays, but you cannot load more until the hold is below the original limit.

A Ship Dealer mod can also change the afterburner (a higher afterburner speed, quicker acceleration while it burns, less or more energy drawn) and the jump drive (how many jumps a full tank holds), on the stock items or as new items of its own. A jump drive with a bigger tank comes full when you buy it, every landing fills it again, and the FUEL gauge and the Navigation Map count against its own tank. Your save records a stock afterburner and jump drive with at most 6 jumps; how many more a new jump drive holds is noted beside it, so the save loads without the mod and in the original game with up to 6. These items work in a multiplayer session too, when the server runs the mod; there the afterburner draws no energy.

A Ship Dealer mod can also change the turrets (how fast a turret turns, how far it can aim to each side and up or down, and the direction it rests in), on the stock Tail, Top and Bottom Turrets or as new turrets of its own. A new turret is bought like a stock one and goes on the first free turret position of its kind; on a ship with several turret positions of one kind each position carries its own turret, and selling a turret takes it from the last position of its kind that carries it. Your save records the stock turret of each kind; which positions carry a new turret is noted beside it, so the save loads without the mod and in the original game with the stock turrets. Turret changes work in a multiplayer session too, when the server runs the mod.

A Ship Dealer mod can also change how many missiles or torpedoes each launcher carries, for every type of ship or differently for each, and on which positions of a ship the dealer fits a launcher. It can only take positions away, so a ship may take fewer launchers, or none: the dealer then answers NO ROOM ON SHIP. A launcher you already carry on such a position stays and fires, and rounds above the new limit stay aboard, but the dealer sells no more until you are below it. These limits count at once; your save keeps your launchers and rounds as they are, and without the mod (or in Classic, or with Item Mods off) the usual limits apply again.

A Ship Dealer mod can also change the colours a scanner shows radar contacts in (hostile and friendly ships, bases, missiles, pods and ejected pilots, the nav point) and which objects its target box colours, for each of the nine scanners of the Software Booth. The cockpit radar, the turret radars and the generic HUD's radar all show the mod's colours; without the mod (or in Classic, or with Item Mods off) each scanner has its usual colours.

It can also change what a scanner does and costs: whether it can lock (**L**) and toggle the ITTS sight (**I**), how fast it identifies the contact you target, its range (which is both how far away you can target and

lock and how far the radar shows contacts), how much damage it takes before it fails, and its name, prices and pictures at the Software Booth. Your save keeps the same scanner; without the mod (or in Classic, or with Item Mods off) it works, costs and looks as usual again.

A mod can also add new scanners. The Software Booth then has more than one page: its first page is the usual grid of nine scanners, and the two small buttons on the panel's top rim, either side of **REPAIR** (**<<** and **>>**), turn to the pages of new scanners, nine to a page in the same grid. Click a scanner to select it, or step through every item, new scanners included, with the panel's **<<** and **>>** below the display; the grid turns to the page of the scanner selected. A new scanner is bought, sold and repaired like any other, at its own prices, and you still carry one scanner at a time. It shows the radar contacts in its own colours and has its own range, lock and identification. Each new scanner stands in for one of the nine usual scanners: in Classic, with Item Mods off or without the mod your ship has that usual scanner, and your save loads with it in the original game; when the mod is removed for good, a notification names the new scanner and the usual one you now have. Buying a ship with a trade-in keeps your scanner.

Ship Dealer mods apply only in Enhanced and Modern; Classic always shows the original dealer. In a multiplayer session the server's Ship Dealer mods apply instead of yours (see Multiplayer). The **Item Mods** box in the debug overlay's MODS section (or `ItemMods=0` in `openprivateer.ini`) turns them off and on, and the change takes effect at once, also with the dealer open. Nothing a Ship Dealer mod changes is stored in your save except which new items your ships carry (a new turret with its turret position, a new scanner) and the jumps left in a new jump drive.

The same mods can also change the Commodity Exchange: a good's name, its prices and how much of it a kind of base, a system, a region or a single base buys, sells or keeps in stock, which market a base added by a universe mod trades (the base it borrows its look from, another base's or a kind of base's), and the cargo runs on the mission boards follow what a kind of base now produces and buys. They follow the same **Item Mods** box and apply from your next landing (the names at once); Classic always trades the original prices. Prices and stock are never stored in your save.

A mod can also add new trade goods: they are listed after the usual 40, bought and sold like any other good, shown in the cargo manifest under their own name, and carried on the mission boards' cargo runs. With the mods switched off (Classic, or **Item Mods** off) the Exchange neither buys nor sells them, but they stay in your hold. Your save keeps them; the save itself holds each one as the usual good it is based on, so a game without the mod (also the original DOS game) loads that good instead, and the game tells you so in a message. Load the save with the mod installed again before saving over it to keep the new goods.

A mod can also decide which goods are contraband: make a legal good illegal, make Tobacco, Ultimate, Brilliance or Slaves legal, add a new contraband good (sold at pirate bases and bought dear elsewhere, like Tobacco), or make a good illegal or legal only in some systems, a quadrant or the systems of one faction. Patrols search for exactly those goods in the system you are flying in, your secret compartment takes them first, a wingman refuses them, and smuggling jobs carry them. With the mods switched off (Classic, or **Item Mods** off) only the usual four goods are contraband. Check the log at start, or start with `--commodity-report`, to see each changed good's legality.

In a multiplayer session the server's mods decide the Commodity Exchange, new trade goods and contraband included: every pilot sees the same goods and names and is searched for the same

contraband, while each pilot's prices and stock are still their own. Joining with new trade goods in your hold that the server's mods do not have gives you the usual good each is based on instead.

A file with a mistake is not loaded at all, and the log names the field and the reason. While any Ship Dealer mod is installed, the log also lists every item the dealer offers with its name, prices, ships and bases, and marks what each mod changed. Starting the game with `openprivateer --dealer-report` writes that list to the log even with no mod installed, and to the console when there is one; the game then starts as usual. Adding `--dealer-base=<base name>` also shows whether that base sells each item. The generated `content/mods/items/README.txt` has the format and an example.

Start mods (experimental)

A start mod changes how a NEW game starts: the base you start at (an original base or an Open Universe base such as Gemini), your credits, your standing with each faction and your kill record, guild memberships, the quadrant maps you own, the name and callsign the registration screen suggests, and your whole ship: hull, guns, launchers, missiles, equipment, scanner and cargo. Each start mod is a folder holding an `init.json`, for example `content/mods/init/mystart/init.json`:

```
{
  privateer: {
    start: "New Detroit",
    credits: 50000,
    ship: { hull: "centurion", launchers: { "2": "missile_launcher" } },
  },
}
```

The file has one part for Privateer (`privateer`) and one for Righteous Fire (`righteous_fire`); only the game you play uses its part, and anything the file leaves out stays as in the original start. Equipment uses the names the Ship Dealer shows (`"Tungsten Armor"`), guns can also use the short weapon names (`laser`), and guns and launchers are placed on numbered hardpoints. The hull can also be a ship mod's ship you fly (`hull: "capital/captain"`, see "Ship mods"); that ship mod must be installed, otherwise the start mod is not used and the log says why. Equipment can also be a Ship Dealer mod's new item by its name in the mod (`shield: "shipyard/mk4_shield"`, or `gun_cooler: "shipyard/chiller"` in place of `true`); that Ship Dealer mod must be installed in the same way.

Only one start mod can be used. With a single folder it is used automatically; with several, pick one in the launcher's Mods section (**New Game Mod**), in the debug overlay's MODS section (**Init Mod**) or with `InitMod=<folder>` in `openprivateer.ini`, otherwise none is used and the log lists them. The **Init Mods** box (or `InitMods=0`) disables start mods. They apply only in Enhanced and Modern and never in a multiplayer session; Classic always starts as the original does, and Righteous Fire's "Load an old Privateer ship?" ignores them. A start at an Open Universe base needs Open Universe on; otherwise you start at the original starting base.

The main story may not work when a start mod changes the start: the story expects the original start.

A new game from a start mod saves like any other game, so the save loads in Classic too. A file with a mistake is not used at all, and the log names the field and the reason. The generated `content/mods/init/README.txt` has the format and an example.

Mission mods (experimental)

A mission mod adds missions to the mission boards: the Mission Computer terminal and the Mercenaries' and Merchants' Guild computers offer them beside their usual random missions. Each mod is a folder holding a `missions.json`, for example `content/mods/missions/mymod/missions.json`:

```
{
  missions: {
    troy_sweep: {
      kind: 'patrol',
      board_text: 'Sweep $NN nav points in $DS for $EN raiders. $PY credits.',
      summary: 'Patrol $NN nav points in the $DS system.\n\n$CL pays $PY credits.',
      destination: { systems: ['Troy'] },
      offered: { boards: ['terminal'], weight: 100 },
    },
  },
}
```

A mod mission is one of the usual kinds: patrol, scout, defend, attack, bounty, cargo, or one of the bar strangers' sabotage, spy and smuggling jobs. You take it, fly it and get paid exactly like any other board mission: it fills one of your three mission slots, shows its objectives on the Navigation Map and in the cockpit, pays when you land after completing it, and appears in the notification center. What the mod chooses is the text the board shows and the text your Mission Computer keeps, who pays, where the mission sends you, which enemy you meet and how many, the cargo, the pay, and at which bases and on which boards it is offered. A mod can also choose the ships you meet: a flagship with its escort, a mix of ship types, pilots of a given skill and temper, friendly wingmen at your side, up to 64 ships in one mission, in several versions of which one is picked for each offer (a harder one for a more experienced pilot, perhaps paying more). Such a mission can also set its own goals, listed on the Navigation Map as usual: destroy a named ship ("Destroy Kilrathi") or a number of the enemy, keep a friendly ship alive until every attacker is destroyed ("Defend Merchant"), destroy every enemy ship there ("Clear navpoint"), or reach a given nav point. A mission whose ship to defend is destroyed first cannot be completed any more: reject it on the Mission Computer. A mod mission can also send in reinforcements: more ships that arrive once during the fight, after you have destroyed some of the enemy, after a while at the nav point (the time counts only while you are there), when you reach another nav point, or when you have met one of the goals. They appear at the nav point, at another nav point or near a jump point, and a goal to destroy every enemy ship or to keep a friendly ship alive is

met only once the reinforcements have come and been destroyed too. In Enhanced and Modern no more reinforcements come once the mission's goals are met: destroy the target of a mission first and the escorts you shoot down afterwards call in nobody (ships already there stay and fight on). Reinforcements that answer the very goal you just met, a militia patrol coming in once the nav point is clear, still arrive. In Classic they come whenever they are called. A mod mission can also fail outright: when a friendly ship you were to protect is destroyed, when its time limit runs out (counted while you are at the nav point), or when you land without all of its cargo. The cockpit message line says "Mission failed" with the reason at once; on your next landing the mission leaves your list unpaid, its cargo is taken back, and the notification center gives the reason. A failure after you have already met the mission's goals does not count. Mod missions may also give their ships jobs: an enemy flagship's escorts stay with it, attack any ship it is hostile to (you included) that comes within about 8000 units of it and then fall back to it, a picket holds a nav point, raiders go after a freighter before they turn on you, and a convoy of freighters flies its own route (it does not wait for you) and leaves space at its end, at a base or a jump point. A mission to escort such a convoy is met when every freighter has reached the end of its route. Some ships meet you elsewhere than at the mission's nav point: they appear when you get there. An enemy flagship may run for the jump point or a base once its escort thins out; if it gets there, it is gone, and such a mission may fail with "The target escaped". A mod mission may also offer bonus objectives: extra credits for destroying more of the enemy, clearing every enemy ship, keeping a friendly ship alive or finishing quickly. They never stop the mission from completing. The cockpit message line says "Bonus" when you earn one and "Bonus missed" when you lose one; the bonuses you earned are paid together with the mission when it pays, the notification center lists each one, and the Mission Computer's MISSIONS page tells you how much the bonuses can pay. Classic pays no bonus. A mod mission's ships may also be ships from a ships mod, and its pilots may speak with the callsigns, faces and lines of an NPC mod; a saved mission whose ships mod is no longer installed flies with the standard ship it is based on. Words starting with `$` in the texts are filled in when the board shows the mission (`$DS` the system, `$PY` the pay and so on; the generated README lists them all). A mission without `destination.systems` goes where the usual missions go: most often the system you are in, otherwise a system picked the original game's way, which favours the Clarke quadrant wherever you are; the generated README explains the rule.

Each board slot picks between the usual missions and the mod missions offered there, so a board shows a mix. A mod can also change the usual missions of the Mission Machine and the two guilds: post some kinds more or less often or not at all (a Mercenaries' Guild with more bounties, a Mission Machine without scouting jobs), post only its own missions there, or send other ships to the usual missions (pirates waiting at every Merchants' Guild delivery). Such a usual mission keeps its text and pay.

A mod can also offer its job through a stranger in the bar (see **Fixers in the bars** in section 4): the stranger then speaks the mod's own lines, perhaps with a face of the mod's that talks and blinks and with recorded voices (heard with Sound F/X on, a woman's voice where the stranger is a woman), and Accept takes the mod's mission. When a stranger stands in a bar, the mod's jobs offered at that base share that place with the usual strangers, so it is sometimes the mod's stranger and sometimes a usual one. This needs both **Fixers** and **Mission Mods** on. If you turn **Mission Mods** off while the mod's stranger is there, Accept says the job is no longer offered. At an Open Universe base such as Gemini the boards offer the usual missions of the original base it borrows its rooms from (when it names one) and the mod missions offered there; without mission mods they stay empty. Mod missions can also send you into an Open Universe system such as Gemini itself (a patrol of its nav points, an attack at one of them) and deliver cargo to, or defend, one of its bases or stations: for a delivery, land at that very base.

Mission mods apply only in Enhanced and Modern; Classic always shows the original boards. In a multiplayer session the server's mission mods apply instead of your own, whatever the box below says (see **Missions in multiplayer** in section 22). The **Mission Mods** box in the debug overlay's MODS section (or `MissionMods=0` in `openprivateer.ini`) turns them off and on; the change applies the next time a board shows its missions, and a mod mission still on a board can no longer be taken once they are off. Missions you already carry keep going: your save stores the whole mission, so it flies on without the mod and in Classic too. The one exception is a mission into an Open Universe system or to an Open Universe base: it is kept in the save's `.openpriv.json` file and is there only when you load the save in Enhanced or Modern with Open Universe on and the same Open Universe content installed. Otherwise the mission is gone from your list, and the notification center tells you which one and why.

A file with a mistake is not loaded at all, and the log names the field and the reason. Starting the game with `openprivateer --missions-report` writes the list of mod missions to the log and the console; the game then starts as usual. The generated `content/mods/missions/README.txt` has the format and an example.

Ship mods (experimental)

A ship mod adds variants of the computer-flown ships: a ship based on a stock one, such as the Paradigm, with its own name on the target display, its own shields and armour, other guns, missiles and turrets, its own speed, turning and energy, its own loot, and its own effect on your standing with the factions when you destroy it. A ship mod can also bring a new ship with a look of its own: its own pictures in flight and on the target display, its own size, and its own places where its guns sit and where hits land. A mod ship can carry up to 24 turrets, each covering its own side of the ship, and a mod ship can be a **station**: it never moves or turns, but its turrets and guns fire at you as any ship's do, and you destroy it as any ship. A mod ship's turret can also be a **missile turret**: it turns toward you and launches missiles or torpedoes at you. An Open Universe mod places these ships in its encounters, and can make a station a carrier you land on (see the universe mods above). Each mod is a folder holding a `ships.json`, for example `content/mods/ships/capital/ships.json`. A variant can have a **phase shield**: while that side of its shield is up, your guns, missiles and torpedoes do nothing to it, the hits land and the shield takes them without draining. Only a weapon a weapon mod makes able to go through a phase shield, such as an anti-matter gun, reaches it; once such hits drain that side or wreck its shield generator, every weapon gets through there. On the Target MFD, a side whose phase shield is up has a thin light blue layer just outside its shield bars; the bars themselves show and drop away as on any ship, and a side without the blue layer is an ordinary shield. Some variants play their own sound when their phase shield stops a hit. You never buy a variant, and a ship of it you meet is never in your saves.

A ship mod can also bring a ship **you fly**, a mod ship with its own speed, turning, size, armour, damage areas, gun positions and pictures in the outside views (F7 to F9); your ship looks like itself in the autopilot flyby and the jump sequence too. Each such ship is based on one of the four ships you can own, its "alias": it uses that ship's turrets (unless the mod gives it turret seats of its own, up to eight), Weapons display and death pictures, and your save records it as that ship. The takeoff and landing movies show the mod ship itself, flying the alias's path past the base (a mod can keep the alias's ship there instead). It can take a moment to appear when the movie starts right after you first fly the ship. A mod can give it a cockpit of its own, a repaint of the alias's: the views, gauges and damage marks then have the mod's pictures (a view the mod leaves empty shows open space with only the HUD), its windows, gauges and displays where the mod puts them, and its energy and fuel gauges go dark the

way the mod says (a cockpit taken from another Wing Commander game drains them as that game does). The drained part of a gauge shows black, or a picture the mod gives it: a cockpit taken from Wing Commander 1 or 2 shows the game's dim "empty" cells there (the Sabre's fuel tank empties into dark blue cells). Such a cockpit can also have its own MFDs, for example two displays on a ship whose alias has one: each shows the pages the mod gives it, `[` cycles the left one and `]` the right one, and the page keys bring each page to its display (a cockpit taken from Wing Commander 1 or 2 shows the weapons and damage pages on the left display and the destination, target and comm pages on the right, as that game does). A cockpit can also show your shield strength as numbers in the front view, which count down as that side of the shield takes hits and climb back as it recharges (a cockpit taken from Wing Commander 1 or 2 prints the front and rear shield beside its shield display, as that game does). The number is a percent of that side's full shield: 100 at full, 1 to 99 as it takes hits, and 0 only when that side is empty or the shields are switched off or none is fitted; a damaged Shield Generator or a lowered shield power level keeps it below 100. A mod can show the Shield Generator's strength itself instead. Such a cockpit can also have its own LOCK light, blinking in the front view while a missile is homing on you, in place of the LOCK mark on the radar (a cockpit taken from Wing Commander 2 blinks its red LOCK label as that game does: a fast flicker while the missile is far, slower as it closes in; one from Wing Commander 1 flickers its missile warning lamp). A cockpit's turret view (F5) can have its own radar, centred on where the turret points, and can leave out the turret MFD. Such a radar shows the turret's aim with two short white lines: one on the side the turret is turned to (left or right) and one above or below for its elevation. Each appears at the radar's edge as the turret turns away from the middle of its travel and moves in towards the centre, which it reaches when the turret can turn no further that way. A cockpit taken from Wing Commander 2 shows its gunner's seat in the turret view, with the gun energy column as the turret's energy gauge and the gunner's scanner as the radar; that game's seat is its rear view, so F4 there looks out behind the ship with no cockpit around it. A mod ship without a cockpit of its own starts each flight in the cockpitless HUD (see section 14), and `,` shows its alias's cockpit. You buy one at a Ship Dealer that sells it (click the salesman or press `B` in the showroom, see section 11), at the price the mod gives it, and it trades in at that price or at the share of it the mod sets; or you start a new pilot in one with a start mod (see "Start mods": `ship: { hull: 'capital/captain' }`). Fitted, parked and flown by a hired wingman, it works like any ship you own. Its own gun and launcher positions and its hold (at most its alias's) are what you fit and carry; the Ship Dealer offers it the equipment the mod allows, and Repair & Upgrade and the Launch Pad show the mod's picture of it, or its alias's; a mod can give some launch pads a picture of their own, seen from that pad's camera.

Your saves keep the mod ship as long as the mod is installed. Loaded without the mod, the ship is its alias: guns and launchers that had no place on the alias are not on it, the log says so and the notification center shows a message; save over it only when you no longer want the mod ship back. In Classic and with **Ship Mods** off the ship flies as its alias and stays the mod ship in your save. In a multiplayer session you fly a mod ship when the server runs its mod (see "Flying a mod ship in multiplayer" in the multiplayer section); otherwise you do not join, and the game continues in single player.

A ship mod can also change a stock ship itself: then every computer-flown ship of that kind, for example every Centurion you meet in the encounters and in the missions, has the mod's armour, guns, speed or name, and can have new pictures, a new size and new places where its guns sit and where hits land. Your own ship stays as it is and keeps its own look in the outside views, and so does an owned ship a hired pilot flies for you or that comes along as a wingman. Without the mod, every ship is the stock one again.

A ship mod can also change the Tarsus, Orion, Centurion and Galaxy you fly: their speed, acceleration and turn rates, their size and where hits land, where their guns sit and which positions you can fit, their cargo hold, their armour when you have none fitted, and what the Ship Dealer asks for them, what they trade in for and what they come with. Clicking the Centurion in the showroom then buys the changed Centurion, and it flies that way from your next launch; the equipment you buy is still the Ship Dealer's, and your ship keeps its own look in the outside views. Your saves record the stock hull, so without the mod, in Classic and with **Ship Mods** off the ship flies as the stock one. In a multiplayer session the server's ship mods decide: when they change a hull you fly, it flies changed for you and every other pilot, otherwise it flies as the stock one.

Ship mods apply only in Enhanced and Modern; Classic never shows a variant. In a multiplayer session the server's ship mods apply instead of yours (see Multiplayer): the changed stock ships are changed for everyone, and new mod ships appear where the server's universe mods place them in the original Gemini systems. The **Ship Mods** box in the debug overlay's MODS section (or `ShipMods=0` in `openprivateer.ini`) turns them off and on; the change applies when you next enter a star system, and with them off the encounters that would bring a variant bring none and every ship is the stock one. Starting the game with `openprivateer --ships-report` writes the list of variants and changed stock ships to the log and the console; `openprivateer --ships-preview <folder>` writes each mod ship's and each changed stock ship's pictures, in the game's own colours, into that folder.

While the ship mods load, the loading screen counts them ("Loading ship mods: wc3ships 12 of 36"). The game reads a mod ship's pictures the first time that ship appears in flight, so starting the game stays quick however many mod ships are installed. If one of its pictures is damaged, the log says so then and the ship shows the pictures of the ship it is based on; `--ships-report` checks every picture at once. The generated `content/mods/ships/README.txt` has the format and an example.

To make a new ship from a stock one, start from the asset export (section 25): every ship that flies in the game's encounters has a `ship_mod.json5` in its folder there, the ship with all its values as a ship mod entry. Copy that folder into your mod (for example `content/mods/ships/capital/GOTHRI/`), name it in the mod's `ships.json` as `{ ships: [ { file: 'GOTHRI/ship_mod.json5' } ] }`, and the mod has a ship that is exactly the stock Gothri, with its own pictures from that folder. Give it a new `key`, change the values and repaint the pictures you want to change, and place it with an encounters row. To change every Gothri you meet instead, name the same folder as a changed stock ship, `{ override: { GOTHTYPE: { file: 'GOTHRI/ship_mod.json5' } } }`: as exported it changes nothing, and what you change or repaint there changes every computer-flown Gothri. The Talon's two folders can be used this way only without their pictures (delete their `art`).

The folders of the four ships you can fly (`tarsus/`, `orion/`, `centurion/`, `galaxy/`) make a ship you can fly as well: copied into a mod as they are, the Ship Dealer sells the copy at the stock ship's price, and its Weapons MFD shows the stock page, drawn from `mfd_weapons.png` in the folder with each gun and launcher icon at its `mfd` point in `ship_mod.json5`. Repaint that picture and move the points for a ship of your own. To use one of these four folders for a computer-flown ship only, or to change every computer-flown ship of that kind, delete `player`, `art.mfd_weapons` and every `mfd` from its `ship_mod.json5` first.

OpenPrivateer also ships built-in Open Universe content. In Enhanced and Modern with Open Universe enabled, **Gemini** is a flyable system in Humboldt, the metropolis of the sector. Its nav points:

Nav	What is there
1	**Gemini**, the metropolis planet (landing base)
2	**Heliopolis Array**, a solar array station (no weapons fire)
3	**Novaya Riga**, a run-down refinery station
4, 5	Asteroid fields
6, 7	Empty space
8, 9, 10	Jump points to Troy, Junction and Aquila

Return jumps connect those systems back to Gemini: Troy at Nav 9, Junction at Nav 10 and Aquila at Nav 1. You can fly through Gemini with autopilot and leave through any of its three jump points. On the Navigation Map, QUAD and `Home` show the quadrant of the system you are in (Humboldt for Gemini, Petrov for Aquila). Selecting one of Gemini's nav points shows "Jump to <system>" for a jump point, the station's name and type for a station, and "Asteroid Field" or "Empty Space" for the plain nav points. Gemini's sky shows no sun or planets yet.

Gemini has traffic like an original system. Near each nav point you may meet militia patrols in Gladius fighters, merchants in Tarsus and Drayman transports, or pirates in Talons and in Demon fighters; around the planet a militia Paradigm sometimes patrols. Pirates are most common around Novaya Riga and the asteroid fields, the militia around the planet and the stations. Which group you meet at a nav point is decided each time you enter the system, and its ships appear when you come near the point. Autopilot stops when its route crosses another nav point's area, as in an original system.

Weapons cannot be fired near the **Heliopolis Array**: within about 12000 meters of it your guns, turrets and missiles stay silent and the message line reads "No weapons fire zone". Other ships are not bound by it. The tractor beam still works.

A visit to a built-in mod base uses the interior of a similar original base, but the mod base keeps its own name and location and only its configured services are available. Its Mission Computer and guilds have no missions yet, and missions cannot be delivered there. You launch back into Gemini. Saving while docked at a mod base stores the extra location in the matching `.openpriv.json` sidecar while the original save stays in the original format. If you turn the built-in content or Open Universe off before loading, you load at the similar original base instead.

The asteroid fields at Nav 4 and Nav 5 work like original asteroid fields: rocks can hit the ship (a rock bursts on impact, so each one hits only once), and autopilot stops when it reaches a field.

A `navpoints.json`, `zones.json` or `encounters.json` that reuses a built-in key with the same `mod` and `key`, or adds a second jump between the same systems, is not loaded. An encounter table on a built-in nav point or zone is added beside the built-in one.

To disable the built-in content, uncheck **Built-in Content** under **Open Universe** in the debug overlay, or set `UniverseBuiltinContent=0` in `openprivateer.ini`, then restart.

Translations live under `content/i18n/` with names such as:

```
openprivateer_de.i18n
```

The translation format uses `key=value`, `#` comments and `\n` for embedded line breaks.

Save the file as UTF-8. With the original fonts, letters with accents are shown without them (`é` as `e`); with the Spanish translation's fonts the Spanish letters show with their accents (see **Font Charset** in section 3); with a custom font of your own, choose **latin1**.

21. Optional 1993 Alpha Content

The `alpha/` folder is for optional material from the August 1993 Privateer alpha.

It is not distributed with OpenPrivateer and is not required to play.

See `alpha/README.txt` for the exact files expected by your build.

22. Multiplayer

OpenPrivateer includes dedicated client/server multiplayer. It is still an Alpha feature.

Server

Basic example:

```
openprivateer_server --tre /path/to/PRIV.TRE
```

The default UDP port is `24601`.

Useful server options include `--port`, `--max-players`, `--verbose` and `--experience-mode`. `--stop-on-stdin-close` stops the server when its standard input closes, for a program that starts the server and wants it to end with itself (the launcher's **Host a session** uses it); do not use it for a server that runs without a console. To let only the pilots you choose join, give the server a password (see "Private servers (a password)" below). To list the server on a master server, see "Listing your server on a master server" below.

`--all-equipment /path/to/RF.TRE` makes a Privateer server's sessions sell and fly Righteous Fire's equipment (see section 11). It needs `--tre` and is refused in Classic and with Righteous Fire data. Every player must set **All Equipment** the same way as the server: a client that differs is refused with "All Equipment mismatch". On a server without it, equipment from Righteous Fire on a pilot's ship (the Fusion Cannon, Shield Generator 6 and 7, Isometal Armor, the Gun Cooler) is not used in the session; the save keeps it.

`--kill-bounties` pays kill bounties to every pilot in the server's sessions (see **Kill bounties** in section 16). It is refused in Classic. Without it nobody is paid a bounty in a session.

A server can run weapon, Ship Dealer, ship, universe, interior and mission mods: `--mods DIR` takes a folder laid out like your own `content/mods` (for example `DIR/weapons/<mod>/weapons.json`, `DIR/items/<mod>/shipstuff.json`, `DIR/ships/<mod>/ships.json`, `DIR/universe/<mod>/encounters.json`, `DIR/interiors/<mod>/interiors.json`, `DIR/missions/<mod>/missions.json`, or a pack in `DIR/packs/<mod>/`), with an optional

`DIR/mods.json` for the order (`--mods-config FILE` reads the order and the disabled mods from FILE instead, for example the `mods.json` in your own data folder). Only weapon, Ship Dealer, ship, universe, interior and mission mods run in multiplayer. A server's mission mods offer their missions to every pilot on the boards and from the strangers in the bars, the server flies them, and their bounty rules decide the session's kill bounties; a mission mod that sends pilots into Open Universe systems or bases needs a server started with `--open-universe`, and a server without it does not start with one. An interior mod that gives a real Gemini base a new interior (section 20) shows it to every pilot who lands there, on any server; its other interiors, those of Open Universe bases, show only on a server started with `--open-universe` (below). A server with Ship Dealer mods gives every pilot the modded shields, armour, engines, improvers and new items, each ship's own where a mod sets them per ship, and a ship mod that changes a stock ship changes every computer-flown ship of that kind in the session, a ship mod that changes a stock ship you fly changes it for every pilot in the session, and the mod ships a ship mod lets you fly can be flown there (see "Flying a mod ship in multiplayer" below). Of a universe mod, a session runs only the encounter tables in the original Gemini systems: they add, take away or change the original's groups of ships at a nav point, exactly as in single-player, and can bring a ship mod's new ships into the session; its new systems, nav points and zones, and its tables in those systems, are used only by a server started with `--open-universe` (below). The server does not start with `--mods` in Classic, with a mod file that has a mistake, or with a weapon a session cannot run yet: a beam (hitscan) gun, or, on a server without `--open-universe`, a weapon that destroys planets; the message names the mod and its file. `--mods-report` lists each mod with its content hash.

A session runs a weapons mod's new missiles and torpedoes, its new launchers (salvo launchers among them), countermeasure dispensers and their decoys (the countermeasure key works in a session), mines and smart mines, cluster missiles and flak, and rounds that can be shot down: a gun's bolt that meets such a missile, torpedo, mine or decoy hits it, and enough hits or a nearby blast destroy it with a small explosion. Every player sees the same decoys, mines and rounds.

A session also runs a weapons mod's capital weapons: charge weapons (yours charge and fire the moment you let go, as in single-player, and you hear another ship charging near you), lethal weapons, the way computer-flown ships fire their capital guns (barrage volleys, a cannon that charges and then fires at its target, guns that fire only at capital ships or at the ship an encounter sends them after: in a session, the nearest pilot), weapons that go through phase shields, phase torpedoes and missiles whose lock takes a moment, and missile turrets on mod ships. A phase torpedo's decode and a missile's timed lock are kept by the server: your reticle, the lock diamond and the lock sounds show the server's decode or lock, a moment after it, and a press the server would not launch says "Torpedo not decoded", "Missile not decoded" or "Need lock" on the message line. Every player needs a build of protocol 45 or newer; an older client or server is refused with a protocol mismatch.

`--open-universe` (Enhanced and Modern, needs `--tre`) makes the server run the Open Universe in every session: the built-in Gemini and Aquila and every new system, nav point, asteroid zone and encounter table of its universe mods. Pilots jump into them from Troy and Junction as in single-player, fly their encounters, use autopilot there, and cannot fire inside a no-weapons-fire zone such as the Heliopolis Array's. Add `--no-builtin-universe` to leave the built-in Gemini and Aquila out and run only the universe mods' own systems. When you join such a server your game loads the server's universe mods and its built-in setting in place of your own **Open Universe**, **Built-in Content** and universe mods for that run. A client whose game builds another universe (another build or other game files than the server's) is refused with "the server's Open Universe is another". You land at an Open Universe base such as the Heliopolis Array as in single-player (Esc, the automatic landing, or a docking

request on its comm channel), visit it with its own interior when the server's interior mods give it one (with its landing and takeoff movies and its music; else the scenes of the original base it borrows), launch from it and save there; loading a save docked there and joining puts you at that base again. A save docked at a base the server does not have joins at the original base it borrows. The carriers and stations that host mod bases are flown by the server, the same ship for every player: you can land there while the ship stands and is not hostile to you, and a base that will not take you does not announce its landing zone. When such a base can be destroyed and its ship is shot down, it is gone for every player until enough landings anywhere in the session bring it back; it shows "(destroyed)" on the Navigation Map, its wreck (if the mod gives one) stands in its place for every player, and a save docked there joins at the original base it borrows. A planet a mod makes destructible is destroyed the same way when a round of a planet-killing weapon reaches it, whoever fired it: gone for every player, its landing refused, until enough landings bring it back. The session forgets destroyed bases and planets when the server stops, and they never reach your own save.

Client

To join from the launcher, open its **Multiplayer** section, right below **Experience**:

- **Server**: the server's address, `HOST` or `HOST:PORT` (for example `192.168.1.20` or `play.example.org:7000`). Without a port, `24601` is used. IPv6 addresses are not supported.
- **Callsign**: your name in multiplayer, up to 14 characters (letters, digits, spaces and common punctuation; see "Your callsign in a session" above). Empty means `Pilot`.
- **Password**: appears for a server that asks for a password; see "Private servers (a password)" below. It is never saved.
- **Check Server** asks the server which mods it runs, without joining, and compares them with yours. It shows whether the server answered (or why not), its version and mods, and whether all of its mods are installed on this computer, or which are missing or different.
- **Servers** (at the end of the section) lists servers to pick from instead of typing an address; see "Finding a server" below.
- **Host a session** (under Servers) starts a server on this computer for you and your friends; see "Hosting a session" below.

With a server entered, the section title reads "PLAY joins" and the address. **PLAY** then checks the server first ("Checking the server…"; it takes at most a few seconds) and starts the game only when the server answers and all of its mods are installed on this computer (and, for a server that asks for a password, once one is typed in **Password**); the session is joined once you load a game or start a new one. A successful **Check Server** in the last 30 seconds counts, so PLAY starts at once after it. If the server cannot be reached, answers with another version, or runs mods this computer lacks or has in a different version, the game does not start: the Multiplayer section opens with the reason and "Invalid server address. Correct it or leave the Server field empty to play single-player." Start the server, fix the address or install the server's mods and press PLAY again, or clear **Server** to play single-player. Both fields are remembered for the next start. Multiplayer uses the server's weapon, item (Ship Dealer), ship and universe mods instead of your local mods.

From the command line:

```
openprivateer --connect HOST:24601 /path/to/PRIV.TRE
```

`--connect` skips the launcher, and `--callsign NAME` sets the callsign; both take precedence over the launcher's fields.

Finding a server

Open **Servers** at the end of the launcher's Multiplayer section. It has four tabs, each with the number of servers in it:

- **Favourites**: the servers you added.
- **Recent**: the last 10 servers you started the game for with **PLAY**, newest first.
- **LAN**: servers on your local network. **Search** (also done when you first open the tab) asks the network; your favourite and recent servers at local addresses are asked too, so one on another port is found as well.
- **Internet**: the servers of the server list fetched from openprivateer.org (see "A server list" below).

Every server in a tab is asked who it is when you open the page or the tab and when you press **Refresh**. The table shows its name (a `*` marks a favourite), players (`3/8`), ping, game and Experience mode, how many mods it runs (with "(yours)" when they are the same as yours) and its version. Click a column's title to sort by it, again to sort the other way; at first the servers with the most players come first, then the one with the lowest ping. **Filter** shows only the servers whose name or address contains what you type; **Hide full**, **Hide empty** and **Hide incompatible** hide those servers.

Click a server to select it: its address goes into the **Server** field and **PLAY** joins it as usual (it checks the server first). **Join**, or a double click, does the same and presses PLAY. Under the table the selected server's description, whether you can join it, its mods, the pilots on it, its rules and how long it has been running are shown.

A server you cannot join with this game is greyed, with the reason under the table and when you point at it: it runs another version of OpenPrivateer, the other game (Privateer or Righteous Fire), another Experience mode than the one chosen under **Experience** (change it there and the server is no longer greyed), or it is full. A greyed server cannot be joined from the list. A server that does not answer shows "no answer": it may be down, hidden from lists or behind a firewall; you can still try to join it. A server that asks for a password shows "[password]" after its name; selecting it brings up the **Password** field (see "Private servers (a password)").

To add a server, type its address (`HOST` or `HOST:PORT`) into **Address** under the table and press **Add to Favourites** or Enter. The button above it adds the selected server to your favourites, or removes it when it is one.

Asking a server shows it your internet address, as joining does. Nothing is asked while the Servers page is closed.

Your favourites and recent servers are kept in `servers.json` in your data folder (see "Where your files live"), with the name each server last answered with and nothing else.

A server list

The **Internet** tab shows the servers of the OpenPrivateer server list, published at `https://openprivateer.org/servers.json`. The list names servers by address, with an optional name,

region, contact and note; everything else (name, players, mode) comes from the servers themselves, so a server that has gone away only shows “no answer”.

When you open the Internet tab, and when you press **Refresh** there, the launcher fetches the list from openprivateer.org in the background. Meanwhile, and whenever you are offline, it shows the copy it kept from the last time, with its date. The line above the table says what is shown: that the list is being fetched, when it was last updated, or that it could not be fetched (with the reason) and which copy is shown instead. A copy older than a week is marked as old. The list is fetched at most once a minute, however often you press Refresh, and never while the Servers page is closed or when the launcher starts.

Fetching the list shows your internet address to the website’s host (GitHub Pages), as the update check shows it to GitHub; nothing else is sent. To turn it off, untick **Fetch the list from openprivateer.org** on the Internet tab (or set `ServerListFetch=Off` in `openprivateer.ini`). The tab then shows only the copy you already have. This switch is separate from the update check’s.

The copy is kept as `servers_cache.json` in your data folder (see “Where your files live”), with its date in `servers_cache_info.txt`. A list that cannot be read is named on the tab with the reason, entries with a mistake are left out and counted there, and a fetch that fails never replaces the copy you have. With fetching off you can place a list there yourself (for example one a community shares); its format is described for list authors in `doc/formats/servers_json_format.md`.

To have your server listed, ask the project: a pull request to the website’s repository that adds your server, or a message to the project owner. Only a server’s operator should ask, and the owner decides.

Master servers and room codes

A master server is a small program that keeps a list of the servers whose operators asked it to list them (see “Listing your server on a master server” below). **The project runs no master server**: there is none unless someone runs one and tells you its address. A community may run one.

The Internet tab has a **Master servers** box. It is empty, and while it is empty no master server is ever asked. Type a master server’s address there (`HOST` or `HOST:PORT`; the port is `24600` when left out; up to four, separated by commas). From then on, when you open the Internet tab or press **Refresh** there, the launcher also asks those master servers for their lists, at most once every ten seconds. Their servers join the tab’s rows, each asked for its name, players and ping like every other row, and a line under the box says how many servers each master listed or that it did not answer. The box is saved in `openprivateer.ini` as `MasterServers=`.

Room codes. A server can stay out of the lists and be found only by a room code its operator shares, like `K7M-Q2X`. Type the code into **Room code** on the Internet tab (in any case, with or without the hyphen) and press **Find** or Enter: the launcher asks your master servers, and the server the code belongs to is added to the tab and selected; press **Join** (or **PLAY**) as for any server. A code works only while its server runs, and only with the master server that gave it.

Joining through a master server. A server found through a master server (its list or a room code) may run on someone’s home computer without a forwarded port. When you join such a server, your game first asks that master server to introduce it to the server: the master server tells the server your internet address, and the server sends your game a few small messages, which opens the server’s home router for your game. This works with most home routers, but not with every one: some routers (and many

mobile and some fibre providers that share one address among many customers) do not allow it, and then the join fails and the server needs its port forwarded (see “Port forwarding”). There is no relay: your game always talks to the server directly. This has not yet been tried across real home routers.

Asking a master server shows it your internet address, as fetching the list shows it to the website’s host. It sees nothing else: not your callsign, not your game.

The Servers page in the game

In Enhanced and Modern the title screen shows **Multiplayer** at its top right. Click it to open the same Servers page inside the game: the same tabs, list, favourites, recent servers, LAN search and server list as in the launcher. Above the list is your **Callsign**. Press **CLOSE** or `Esc` to close the page. Classic’s title screen has no Multiplayer entry: in Classic, join from the launcher.

Select a server and press **Join**, or double-click it. The game then restarts and joins that server: a server’s mods load when the game starts, so joining another server needs a fresh start. Your settings are kept, and the launcher’s **Server** and **Callsign** fields show the server and callsign you joined with. After the restart, start or load your game with **NEW** or **LOAD** as usual to enter the session. If you host a server from the launcher, it stops when the game restarts (the page says so).

For a server that asks for a password, a **Password** box appears under the list when you select it; type the password there (it shows only stars) and press Join. The password goes only to the restarted game and is never saved. If a join is refused for its password, the title screen says so; open **Multiplayer** again, and the page shows that server with the Password box ready.

The page takes the mouse and the keyboard. With a controller, move the pointer with the stick and click with its pointer click button, as on the title screen; typing a callsign, a filter or a password needs a keyboard.

Hosting a session

You can run a server on your own computer straight from the launcher and play on it with your friends. Open **Host a session** at the end of the launcher’s Multiplayer section (under **Servers**) and fill in:

- **Server name**: the name other players see in their server lists (at most 48 letters, digits and common punctuation; an accented letter counts two).
- **Description**: an optional line about your server (at most 120).
- **Port**: the UDP port the server listens on, `24601` unless you change it (1024 to 65535).
- **Max players**: how many pilots can join at once, you included (1 to 255, 8 at first).
- **Hide pilots’ names**: server lists then show how many pilots are on your server but not their callsigns.
- **Chat**: untick it to run the server without chat.
- **Message of the day**: an optional chat line every pilot gets when they join (at most 120 characters).
- **Password**: optional; with one, only pilots who give it can join (see “Private servers (a password)”). It shows only stars. Leave it empty for an open server.

The launcher remembers these fields, except **Password**, which is never saved: type it again each time you host. A field it cannot use is named in red under the form, and **Start** waits until you correct it.

The server plays the game and the Experience mode chosen in the launcher, with your own game files, the mods you have switched on in the **Mods** section and your own **Open Universe**, **Built-in Content**, **Kill Bounties** and **All Equipment** settings (in Classic: no mods and none of these). The lines under the form say what it will run with. Your friends need the same game, Experience mode and mods, as for any server.

Press **Start**. The line under the buttons says "Starting the server on UDP port 24601…" and, once it is ready, "Running: listening on UDP port 24601, 0/8 pilots." The server's address on your own computer (`127.0.0.1`) is then put into **Server**: press **PLAY** to join your own server, or press **Join my own server**, which does both; with a password, the launcher uses it for your own join. While the server runs, the form's fields cannot be changed. **Copy invite** never copies the password: tell it to your friends yourself.

The server keeps running while you play and stops when you press **Stop**, when you close the launcher without pressing PLAY, and when the game ends, also if the game closes unexpectedly. The server writes its log to the `logs` folder of your data folder (see "Where your files live"), named `server_` with the date and time; the form shows its name. If the server cannot start, the line under the buttons says why: for example that the port is already in use (another program, or a server you started yourself, already listens there: pick another port), or the server's own reason. Bans you make on your hosted server (see "Moderation") are kept in `host_bans.txt` in your data folder.

How your friends join

- **On your own network** (the same home network, a LAN party): the form shows your computer's network address with the port, for example `192.168.1.20:24601`. Your friends type it into **Server**, or find your server under **Servers**, on the **LAN** tab. **Copy invite** puts the address on the clipboard, ready to paste into a message.
- **Over the internet**: your friends need your public IP address and the port, and your router must pass that UDP port on to your computer (below). OpenPrivateer does not look up your public address; your router's status page shows it, as do many websites that tell you your IP address. In the text **Copy invite** puts on the clipboard, replace `MY-PUBLIC-IP` with it.

The first time the server starts, your system's firewall may ask whether `openprivateer_server` may use the network. Allow it, or your friends cannot reach the server (on Windows, allow private networks for friends at home, and public networks too if you are on a network Windows calls public).

Port forwarding (for friends on the internet)

A home router lets other computers on the internet reach a game on your computer only when you tell it to. This is called port forwarding, and OpenPrivateer does not change your router's settings for you. In plain steps:

1. Find your computer's network address: the form shows it (for example `192.168.1.20`).
2. Open your router's settings page in a web browser. Its address is usually printed on the router, often `192.168.1.1` or `192.168.0.1`; sign in with the password printed there or the one you set.
3. Find the page named "Port forwarding", "Virtual server", "NAT" or "Applications and gaming".
4. Add a rule: protocol **UDP**, external port **24601** (or the port you chose), internal port the same, to your computer's network address from step 1. Save it.
5. Give your friends your public IP address and the port, for example `203.0.113.7:24601`.

Your computer's network address can change when it restarts; many routers let you keep the same address for it ("address reservation" or "static lease"). Some internet connections share one public address among many customers (called CGNAT, common on mobile and some fibre connections); there port forwarding does not work, and one of your friends with a forwarded port has to host instead. Only the one UDP port is needed; no other port has to be opened.

Hosting shows your public IP address to everyone who joins, as joining any server shows yours to it. Your server also answers server lists and LAN searches with its name, players and settings, as every server does.

Your callsign in a session

The other pilots know you by your multiplayer callsign: when they target your ship, their Target MFD shows your callsign instead of the ship's name, at once. You see theirs the same way.

A callsign holds up to 14 characters: the letters A to Z and a to z, the digits, spaces and the marks `! " # $ % & ' ( ) + , - . / : ; ? _`. These are the characters every font of the game draws the same way, so every pilot reads your callsign as you typed it. The launcher's Callsign field takes only these characters; anything else, from the field or from `--callsign`, is left out, spaces at the start and the end are dropped, and a longer callsign is cut to 14 characters. A callsign with nothing left becomes `Pilot`. The server applies the same rule, so the callsign the Rename Pilot box names is the one the other pilots see.

Callsigns are unique on a server, without regard to capitals. When another pilot on the server already uses your callsign, the server gives you the next free one of `Bob (2)`, `Bob (3)` and so on (shortened at the end when needed so it still fits 14 characters) for as long as you stay; the server tells you in a chat line, and the other pilots see the new name. When a pilot leaves, their callsign is free again. `Server` is the server's own name and is never given to a pilot.

Chat

Servers carry chat between the pilots of a session, in every Experience mode. A line goes to everyone on the server, to the pilots in your star system (docked or flying), or as a whisper to one pilot by callsign. In single player the chat keys do nothing.

Typing. In flight or in a docked scene, press `Y` to talk to everyone on the server or `U` to talk to the pilots in your star system. A one-line box opens at the bottom of the screen, labelled `ALL:` or `SYSTEM:`; type your line and press `Enter` to send it, or `Esc` to close the box without sending. In the box:

Key	What it does
`Enter`	send the line
`Esc`	close the box; what you typed is dropped
`Backspace`, `Delete`	delete before or after the cursor
Left, Right, `Home`, `End`	move the cursor
Up, Down	bring back one of the last 10 lines you sent
`Page Up`, `Page Down`	scroll back through the chat history

Key	What it does
`Tab`	complete a pilot's callsign after `/w`, `/ignore`, `/unignore`, `/mute`, `/unmute`, `/kick` or `/ban`
`Ctrl+V`	paste

While the box is open every key of the keyboard types into it, so your keyboard does not fly the ship, but a joystick, a gamepad and the mouse keep flying it: the session goes on around you and the ship keeps its set speed. Characters that not every pilot's font can show never appear in the box, and it stops at 120 characters. The key that closes the box does nothing in the cockpit until you let go of it, so the `Enter` that sends a line never fires a missile.

The box does not open during a film or a cutscene, a conversation, on the Mission Computer, the Navigation Map, Options, or while the control wheel or another window is open, and it closes by itself (keeping what you typed) when one of these comes up. On a server without chat the key shows "Chat is off on this server."

Commands. A line that starts with `/` is a command:

Command	What it does
`/all TEXT`	to everyone on the server
`/sys TEXT`	to the pilots in your star system
`/w NAME TEXT`	a whisper to one pilot (`/w Bob (2) see you` works; quotes around the name work too)
`/r TEXT`	a whisper back to the last pilot who whispered to you
`/who`	the pilots on the server
`/ignore NAME`	hide that pilot's lines until you quit the game
`/unignore NAME`	show them again
`/help`	the commands and the box's keys

Server admins have more commands; see "Moderation" below.

Reading. Chat lines show at the top left of the screen in flight and in docked scenes, the newest four, each for 10 seconds; with the box open the last eight show and stay. Lines to everyone are white, lines to your star system green, whispers yellow ("Bob whispers: …" and, for yours, "To Bob: …"), the server's own lines orange, and lines the server refused red, with the reason. Lines that come during a film, a conversation or on the Mission Computer wait and show when you are back in flight or in the base. In Enhanced and Modern a whisper to you clicks. With a font that has no accents, accented letters show as the plain letter.

From the Communications MFD. In Enhanced and Modern, when you target another pilot's ship, the Communications MFD (`C`) lists that pilot as "Pilot NAME" after the other contacts (before "Fuel Service" when that is offered). Pick it with its number key, then pick "Send a message": the chat box opens addressed to that pilot (`TO NAME:`), so you only type the message and press `Enter`. The control

wheel's Comms ring offers the same rows. The row is there only while chat is on for the server; in Classic the Communications MFD keeps the original's rows, and you whisper with `/w` in the chat box.

The chat keys are in the "Chat (multiplayer sessions)" group of Remap Controls, together with "Chat: reply to the last whisper", which opens a box addressed to the last pilot who whispered to you and has no key until you give it one. None of them has a controller button until you bind one.

What the server does:

- When you join it sends you its message of the day, if the operator set one, and tells everyone that you joined; when a pilot leaves, everyone is told.
- A line holds at most 120 characters. Letters, digits, spaces and `! " # $ % & ' ( ) + , - . / : ; ? _` are kept, and accented letters (with a font that has no accents they show as the plain letter); curly quotes, dashes and `...` become plain ones; anything else is left out, and line breaks and runs of spaces become one space. Every pilot sees the same line, yours included.
- Four lines may go at once, then one every 1.5 seconds. A line sent faster is not passed on, and the server says how long to wait. Three such lines within 10 seconds mute you for 60 seconds, and the server tells you.
- A whisper to a callsign nobody on the server uses, or to yourself, is not sent, and the server says why.
- The server's console prints every chat line.

For server operators: chat is on unless the server is started with `--no-chat` (then no pilot can open the chat box: the keys show "Chat is off on this server", and the server sends no lines of its own). `--motd TEXT` sets the message of the day (up to 120 characters, the characters above).

Moderation

A server's operator, and pilots the operator trusts with the server's admin password, can mute, kick and ban pilots. This works in every Experience mode.

As a pilot. If the server is muting you, your lines are not passed on, and the strip says so and for how long ("The server muted you for 10 min."); when the mute ends, or the admin lifts it, you may chat again. A mute stays with you when you leave and join the server again before it ends. If you are kicked or banned, your session ends: the game returns to the title screen, which shows "Kicked from server." or "Banned from server." and the reason the admin gave; click it to dismiss it. After a kick you may join again; after a ban the server refuses you when you join, with the same reason, until its operator lifts the ban. The other pilots see a line saying who was kicked or banned.

As an admin. Type `/login` and the server's admin password in the chat box (`/admin` works too). The password is shown as `#` while you type it, is never kept for Up to bring back, and is never written to any log. From then on, until you leave or type `/logout`, the other pilots see "(admin)" after your callsign in your lines and in `/who`, and you have these commands:

Command	What it does
`/mute NAME TIME`	the pilot's lines are not passed on for TIME: `30` is minutes, or `90s`, `5m`, `2h`; 10 minutes when you give no time, at most 24 hours

Command	What it does
`/unmute NAME`	lift a mute
`/kick NAME REASON`	end the pilot's session; the reason (optional) is shown to them
`/ban NAME REASON`	kick the pilot and refuse them from now on (by their internet address and callsign)
`/unban ENTRY`	lift a ban: a banned callsign or address
`/say TEXT`	a line from the server to everyone
`/logout`	stop being an admin

NAME is a callsign as `/who` shows it, without regard to capitals: `/kick Bob (2) spamming` works without quotes, quotes around the name work too, and the start of a callsign is enough when only one pilot's callsign starts that way. You cannot mute, kick or ban yourself. A wrong password is refused; after 5 wrong ones within a minute (or 20 within 10 minutes from one internet connection) the server refuses every login from there for a while and says how long. A pilot who is not an admin who types one of these commands gets "Only a server admin may do that."

For server operators. While the server runs, type a command into its window and press Enter:

Command	What it does
`help`	the commands
`who`	every pilot: slot, callsign, star system, docked or flying, admin, muted, and their internet address
`say TEXT`	a line from the server to everyone
`mute NAME TIME`, `unmute NAME`	as above (TIME optional)
`kick NAME REASON`	as above (the reason optional)
`ban NAME REASON`	as above; `ban ADDRESS REASON` bans an internet address nobody is using now
`unban ENTRY`	the number `bans` shows, an address or a callsign
`bans`	the ban list, numbered
`chat on`, `chat off`	switch chat on or off; pilots who joined while it was off join again to type
`quit`	stop the server (as Ctrl+C)

NAME may also be `#` and a slot number from `who`. A server without a window (run as a service or in the background) runs on without these commands and says so once in its log.

- `--admin-password TEXT` sets the admin password: 1 to 64 letters, digits and `! " # $ % & ' ( ) + , - . / : ; ? _`, no spaces. Anyone who can see the server's command line can read it, so you may set the environment variable `OPENPRIVATEER_ADMIN_PASSWORD` instead. Without either, nobody can log in from the game and moderation is at the server's window only. The server never prints the password, and logs every login, every wrong password and every moderation command.

- Bans are kept in `bans.txt` in the folder the server is started in (`--bans FILE` chooses another file), read when the server starts and written at every ban and unban, so they last across restarts. Each line is one ban, `ADDRESS | CALLSIGN | BANNED AT | REASON`, with `-` for a field left empty: an address bans everyone connecting from it, and a line with `-` for the address bans that callsign from anywhere. You may edit the file by hand while the server is stopped; a line the server cannot read is skipped, and its log names the line. Bans do not expire.
- Internet addresses are kept in the ban file and shown in your server window only; the server never sends a pilot's address to another pilot.
- This keeps unwelcome pilots out of an ordinary server; a determined player can come back from another internet address under another callsign.

Server name and what a server tells those who ask

A server answers a game that asks it, on its UDP port, who it is, without that game joining and without using a player slot. This is what lets a list of servers show each one's name, players and ping. The answer holds:

- the server's name and description,
- Privateer or Righteous Fire, the Experience mode and the multiplayer version and build,
- how many pilots are on it out of how many it takes, and their callsigns with their star system and whether they are docked or flying,
- how many mods it runs (and a fingerprint of the set), whether chat, kill bounties, the refuel service, All Equipment and the Open Universe are on, and how long it has been running.

It never holds a pilot's internet address. A game that asks must first answer a short challenge from the server, and each internet address is answered at most 10 times a second, so the server cannot be used to flood someone else, and asking costs it almost nothing.

Server options:

- `--name TEXT` gives the server's name (otherwise "OpenPrivateer server"): at most 48 letters, digits, spaces and `! " # $ % & ' ( ) + , - . / : ; ? _`; an accented letter counts as two. Runs of spaces become one space.
- `--description TEXT` gives a line about the server, at most 120 of the same characters.
- `--hide-players` shows how many pilots are on the server but not their callsigns.
- `--no-query` makes the server answer no one: lists and the launcher's LAN search do not see it. Joining it by its address works as always.

The server does not start with a name or description that is too long or holds no character the game can draw, and says which option is wrong.

Private servers (a password)

A server can ask for a password, so that only the pilots you give it to can join.

Running one. Set the password in the environment variable `OPENPRIVATEER_SERVER_PASSWORD` before you start the server, for example:

```
set OPENPRIVATEER_SERVER_PASSWORD=Open-Sesame_42

openprivateer_server --tre C:\Games\Privateer\PRIV.TRE
```

(on Linux and macOS: `OPENPRIVATEER_SERVER_PASSWORD=Open-Sesame_42 openprivateer_server --tre /path/to/PRIV.TRE`). `--password TEXT` works too, but anyone on the same computer can see a program's options in the process list, so the variable is the better way. The password is 1 to 64 letters, digits and `! " # $ % & ' ( ) + , - . / : ; ? _`, without spaces; the server does not start with another one and says so. It is never printed. In the launcher's **Host a session**, type it into **Password** (see "Hosting a session").

Joining one. In server lists a server with a password shows "[password]" after its name. When the server in **Server** asks for a password (the Servers page or **Check Server** found out), a **Password** field appears under **Callsign**; type the password there (it shows only stars) and press **PLAY**. PLAY does not start the game for such a server while the field is empty. The launcher keeps the password only until the game ends: it is never saved, not with your favourites and not in `openprivateer.ini`. If the join is refused, the title screen says "Password required." or "Wrong password."; in Enhanced and Modern click **Multiplayer** on the title screen, type the password and press Join (see "The Servers page in the game"), or close the game, start it again and type the password in the launcher. Starting the game with `--connect` and no launcher, it takes the password from the environment variable `OPENPRIVATEER_JOIN_PASSWORD`.

Too many wrong passwords. Five wrong passwords within a minute from one internet address lock that address out for five minutes: every try from it, even with the right password, is then refused as a wrong password. Pilots who share one internet address (one home network) share this count.

What a password does and does not do. It keeps strangers out of your server. The password itself never goes over the network: the game sends a key made from it and a random value the server picks for that one connection, so a key someone overhears cannot be used again. But it is a gate, not encryption: the game's traffic is not encrypted, so someone on the network path can read and change what a session sends, and someone who watched you join could try to guess the password, which a long one makes hard. Do not use a password you use anywhere else.

Listing your server on a master server

A server contacts no master server unless you ask it to. To have it listed, start it with `--register` and the master server's address:

```
openprivateer_server --tre /path/to/PRIV.TRE --name "Friday Night" --register
master.example.org
```

The port is `24600` when left out (`--register master.example.org:24650` otherwise). Repeat `--register`, or give a comma list, for up to four master servers. The server then sends each master server a small message from its game port every minute; the master server checks where it came from, asks your server the same "who are you" question a game asks, and lists your server's address while it answers. So `--register` does not work together with `--no-query`, and the server does not start with both. The console says when a master server lists the server, at which internet address it sees it, and if a master server could not reach it or its list is full. A master server forgets a server three minutes after its last message.

Unlisted, with a room code. Add `--unlisted` to stay out of the master servers' lists. Each master server then gives the server a room code, which the console prints:

```
master master.example.org: room code K7M-Q2X (not in the public list; ...)
```

Tell your friends the code and the master server's address; they type both into the launcher (see "Master servers and room codes"). The code stays the same while the server keeps running. A room code is not a password: anyone who has it can find the server, so use a password too if you want to keep strangers out.

What a master server sees. Your server's internet address and its answer to the "who are you" question (its name, description, modes, players and their callsigns unless `--hide-players`). Never your password, and never the game's traffic.

Without port forwarding. A server listed on a master server can be reached by games that join through that master server even behind a home router without a forwarded port, through the introduction described in "Master servers and room codes". This works with most home routers but not with all of them (not with routers that use a different outside port for every destination, and often not when your provider shares one internet address among many customers); it has not yet been tried across real home routers. Forwarding the port always works.

Running a master server

Anyone can run a master server; the project does not run one. `openprivateer_master` is a small program in the same download folder as the server when it is shipped there, or built from the source with the rest. It needs no game files.

```
openprivateer_master --port 24600
```

- `--port N`: the UDP port it listens on (default `24600`).
- `--max-servers N`: the most servers it keeps (default 1024; at most 4 from one internet address).
- `--ban-file FILE`: its ban list (default `master_bans.txt` in the folder it is started in), one internet address a line with an optional reason.
- `--stop-on-stdin-close`: stop when its standard input closes; not for a master server run as a service.

While it runs, type `status`, `servers` (each server with its address, name, players and room code), `ban ADDRESS [REASON]` (which also drops that address's servers at once), `unban ADDRESS`, `bans` or `quit`. `Ctrl+C` stops it too. It writes what happens to its console: each server listed or forgotten, and a summary of what it dropped.

Where to run it. A master server needs an internet address that does not change and its UDP port open. The smallest rented Linux server (a "VPS", about 4 to 6 euros or dollars a month) is plenty: it uses almost no processor time, memory or traffic (each server sends one small message a minute; a game asks for one to eight small pages of the list). Open the UDP port in the server's firewall, for example with `ufw allow 24600/udp`.

Starting it with the computer (Linux with systemd), as `/etc/systemd/system/openprivateer-master.service`:

```
[Unit]

Description=OpenPrivateer master server

After=network-online.target

Wants=network-online.target



[Service]

User=openprivateer

WorkingDirectory=/opt/openprivateer

ExecStart=/opt/openprivateer/openprivateer_master --port 24600 --ban-file
/opt/openprivateer/master_bans.txt

Restart=on-failure



[Install]

WantedBy=multi-user.target
```

then `systemctl enable --now openprivateer-master`. Its console lines go to the system log (`journalctl -u openprivateer-master`). Without a console, edit `master_bans.txt` and restart it to change the bans.

Looking after it. Everything a server shows in the list (its name and description) is the server's own; the master server only shows it. When a server's name is abusive or someone floods the master server, ban that address. The master server already limits how often each address may ask, never sends an unproven address more than it received, and never relays game traffic.

Server limits

A server takes a steady stream of messages from every pilot, many times what the game sends, and leaves out what goes past that. A connection that keeps sending far more is dropped, and the server's console says so. Ordinary play never comes near these limits.

The server can reject clients when important session properties differ, including build, Privateer vs. Righteous Fire data, Experience mode or All Equipment. A server and a client must run builds of the same multiplayer version; otherwise they refuse each other, and the launcher's server check shows "Different version" with the server's version and this build's. This build cannot join a server running an earlier release, and an earlier release cannot join a server of this build: run the same release on both ends. When the game is started to join a server, the All Equipment box is locked for that run: set it before starting.

When joining, the game first asks the server which mods it runs, during the loading screen. If this computer has all of them (the same mod folders with the same files, a ship mod's pictures included; your own load order and on/off choices do not matter), the game loads exactly the server's weapon, Ship Dealer, ship and mission mods for the session, so everyone flies the same weapons and equipment, sees the same dealer, meets the same changed ships and is offered the same kinds of missions; the server's universe mods decide which ships meet you at the original nav points. A universe mod counts with all of its files, also the ones a session does not use. Your other mods (new game, NPC, and your own universe content) stay off in a session.

Flying a mod ship in multiplayer. A mod ship you fly (section 11) joins a session only when the server runs its ship mod. It then flies exactly as in single-player: its own speed, turning, size, armour and gun positions, and the other pilots see it with the mod's pictures. If your ship comes from a ship mod the server does not run, the game does not join (the log says "join refused: the server's session does not run the ships mod …"), and you keep playing single-player; fly one of your stock ships or one of the server's mod ships to join. A ship of a mod the server does not run, chosen after joining, does not launch. At a Ship Dealer, the salesman or `B` in the showroom lists the stock ships and the server's mod ships that dealer sells, bought as in single-player. If you buy one whose mod is not among your own installed and enabled mods, the showroom's status line says so: your save keeps the ship, but outside the session it loads as its alias until you install and enable that mod. A mod ship with up to eight forward guns or turret seats of its own flies in a session with all of them: keys 5-8 flip guns 5-8, and `F5` / `F6` move between the turret seats. Only the seat you sit in fires.

You fly a session with at most as many missiles and torpedoes as the server's dealer lets your launchers carry; your save keeps your own count. Your afterburner, Speed Enhancer and Thrust Enhancer work as in single-player: without an afterburner the afterburner key gives no boost. A Ship Dealer mod's afterburner, jump drive and turrets work too, but in a session the afterburner draws no energy.

If the server runs a mod this computer does not have, or a different version of it, the game does not join: the title screen shows "Join refused. Server mods differ." (click to dismiss), the log names each missing or different mod with its size, and you can play single-player with your own mods. Install the server's mods (for example by copying its pack folder into `content/mods/packs/`) and start again to join; **Check Server** in the launcher shows the same list before you start. Mods are never downloaded from a server.

If you have the same mod files but your copy of the game cannot load one of them (usually because the server runs a newer build that knows fields yours does not), the game does not join either: the title screen shows "Join refused, weapons mod" (or "items mod", "ships mod", "universe mod", "missions mod") with the mod's name and "did not load.", and the log names the mod and the exact error. You can play single-player, but your weapon, dealer-item, ship and mission mods stay off until you restart the game. Use the same build of the game as the server (or one the server's mods were made for) and start again to join. A mod that loads with warnings joins normally.

When a session ends (you leave it or your ship is destroyed), your own weapon, dealer-item, ship and mission mods stay off until you restart the game; the title screen says "Restart to use your own mods.".

Files in the `content/priv` or `content/rf` override folders only change your own game: the server uses its own copy of the game data. The log warns about them when you join a server.

Multiplayer covers the shared flight/combat environment. The normal single-player campaign progression is not the multiplayer campaign model.

Missions in multiplayer. In a session the Mission Computer and the guild boards offer every kind of mission they offer in single-player (patrol, scout, cargo, attack, defend and bounty), in every experience mode, and in Enhanced and Modern the strangers in the bars offer their jobs too. When the server runs mission mods, the boards and the strangers also offer the mods' missions, exactly as the server's mods describe them, whatever mission mods you have installed yourself; to join such a server you need the same mission mods (see above). When you launch, the missions you carry are handed to the server, which flies them for you:

- Your missions' ships appear for every pilot in the system, and anyone can shoot them. If another pilot (or anyone else) destroys a ship your mission needs destroyed, it counts for your mission.
- Only your own ship reaches your missions' nav points: another pilot flying there starts nothing of yours, and each pilot flying the same kind of mission meets their own ships.
- Only you are paid for your missions, at your own landing, into your own credits, exactly as in single-player (the same pay, cargo check and landing fee). Helping another pilot pays you nothing from their mission.
- Your standing with the factions comes from your own save: pilots who are friendly with the pirates are left alone by them in a session too.
- Your kills change your standing and your kill record in a session as in single-player, by the same amounts, and the Mission Computer's FIN page shows them at once; your save keeps them. A kill counts for the pilot whose shot or ram hit the ship last, and only for that pilot: two pilots shooting at the same pirate never both get it, and a ship that a wingman or another ship finishes off without a recent hit from a pilot counts for nobody, as in single-player. Helping destroy another pilot's mission ships counts for your own standing and kill record, though not for your pay. Ships react to your new standing from then on; a ship that has already dealt with you keeps its view of you.
- The Nav Map ticks off objectives as the server reports them, a short moment after you reach them. In Enhanced and Modern, with **Show Mission Feedback** ticked, the objective plaques show on the message line as in single-player.
- Dropping a mission in flight (Enhanced and Modern, the Mission Computer's reject button) removes its ships from space.
- If you leave the session or your ship is destroyed during a flight, nothing of that flight's missions is kept: they stay on your list as your save has them.
- A mission the server cannot fly stays on your list unflown and unpaid; the log says why. A mission from a mission mod flies with the original ships in place of a mod ship the server does not run.
- A mission mod's reinforcements, escorts, convoys, ships that run for the jump point, failure rules and bonuses work in a session as in single-player, judged on the server: a mission fails there the moment its rule is broken, and the plaque says so in Enhanced and Modern.
- Ships a mission mod marks as robbable are ordinary ships in a session: there is no piracy in multiplayer.
- On a server started with `--open-universe`, the missions into Open Universe systems and bases that your save carries stay on your list and fly there; on any other server they are dropped when your save loads, and a save you make there no longer holds them.

The story in multiplayer. You can play the campaign in a session, in every experience mode. Each pilot has their own story, carried in their own save:

- The story's characters, conversations and cutscenes at the bases are your own, as in single-player. The story missions they give you fly on the server like any other mission: their ships appear for every

pilot, anyone can help you destroy them, and only your own landing completes them. When you land after a story mission, the next character of the story reacts to how it went, as in single-player.
- Two pilots on the same part of the story each meet their own ships, so you may see another pilot's story ships alongside yours.
- The Kilrathi route's jump points open for you as your own story opens them. Another pilot's progress opens nothing for you, and yours opens nothing for them.
- The Steltek drone that hunts a pilot carrying the unboosted Steltek gun comes for that pilot alone, ahead of their ship, though every pilot can see it and fight it. Once your drone is gone, the Derelict Base disappears from your space (from your next arrival in its system on), while other pilots still see it and can land there.
- When a story mission's ships ask you something on the Communications menu, the answers are offered to you alone.
- When the story's victory movie plays in flight, your ship waits in space where it is, untouchable, until the movie ends (at most three minutes). Other pilots see your ship stand still meanwhile.

23. Update Check

Release builds can check for a newer OpenPrivateer build at most once every 24 hours.

Nothing is downloaded or installed automatically.

When a newer build exists, the title screen shows:

```
New build available

Click to download
```

Left-click opens the download page. Right-click dismisses that particular build notice.

The launcher shows the same news as an **Update available** line with a **Download** button. On Windows and Linux, Download opens your browser on the matching archive of the newest build; on other systems it opens the release page, which lists every archive. Unpack the archive yourself to update.

The check can be disabled in settings or for one launch with:

```
--no-update-check
```

The launcher's server list is fetched separately, only when you open the Internet tab of its Servers page, and has its own switch there (see "A server list" in section 22); `--no-update-check` does not change it.

24. Original Developer Switches

OpenPrivateer supports several switches from the original game:

```
-K
```

Skip the intro.

```
-B<n>
```

Start at original base ID `n`.

```
-D<n>
```

Set starting credits to `n`.

Example:

```
openprivateer.exe -K -B39 -D100000
```

These are mainly useful for testing and development.

The launcher's **Original Developer Switches** section sets the same three as checkboxes that are remembered and apply on every start. Switches given on the command line win. The `-B` and `-D` number fields can only be edited while their box is ticked.

Other command-line options

```
openprivateer --help
```

lists every option. The ones for files and the launcher:

Option	Effect
`--home DIR`	Use `DIR` as the user-data folder for this run
`--portable`	Use the program's folder as the user-data folder
`--data DIR`	Look for the game files in `DIR` first
`--launcher`	Show the launcher
`--no-launcher`	Never show it; exit with an error if files are missing

`--lang-data CODE=DIR` (repeatable) looks for the game files in `DIR` first while the **Language** setting is `CODE`, for example `--lang-data es=D:\PrivEsp --lang-data en=D:\GOG\Privateer`.

`OPENPRIV_HOME` and `OPENPRIV_DATA` (a list separated like `PATH`) are the environment-variable versions of `--home` and `--data`; `OPENPRIV_LANG_DATA` (`CODE=DIR` items separated like `PATH`) is the one of `--lang-data`.

`--gpu-debug` (or `OPENPRIV_GPU_DEBUG=1`) runs the graphics device in debug mode for troubleshooting; see "The game closes or crashes while loading" in Troubleshooting.

25. Exporting the Game's Assets

```
openprivateer --export-assets C:\PrivateerExport
```

writes the pictures, sounds, speech, palettes, text and data of your own `PRIV.TRE` and `RF.TRE` into the folder you name, and exits. No window opens; the progress is printed in the terminal. The game files are found the same way the game finds them (the launcher's Game Data folders, `--data`, `OPENPRIV_DATA`, a GOG `GAME.GOG` or a Steam `GAME.bin`); `--tre FILE` and `--rf FILE` name an archive, or a `GAME.GOG` or `GAME.bin`, directly. The export always reads the archives themselves: files in `content/` do not change it.

The folder holds `priv/` and `rf/`, laid out like the archives, with each archive file as a folder of its parts. A container's parts are numbered (`000`, `001`, …), nested containers as subfolders:

File	Content
`.png`	a picture, in the colours the game shows it
`_sheet.png`	every frame of a set of pictures side by side
`_sheet.json`	where each frame of the sheet sits
`.mask.png`	a shape the game fills with a colour (white)
`hud.json`	a cockpit's gauge and screen positions
`faces.json`	which portrait parts make up each character
`.wav`	sound effects and speech, the original samples
`.swatch.png`	a palette, 16 colours per row, index 0 top left
`.gpl`	the same palette for GIMP and other paint programs
`.palette.json`	the same palette's colours as text
`chunks.json`	a data file's structure, each part's bytes as hex
`.txt`	captions and other text
`.json5`	a mod file to copy and edit (see below)
`.bin`	everything else, unchanged (pictures among them)

The pictures written are: every ship from all 37 directions with its afterburner flames and its targeting and weapons display art, the planets, bases, suns, explosions, missiles and other space objects, the stars, the four cockpits (each view as a 320x200 screen, the gauges, crosshair and damage marks), the radar icons, the pause plaques and their lettering, the in-flight comm faces, the conversation portraits and their backgrounds, the title, Options, Mission Computer, joystick calibration and mouse cursor art, the jump, takeoff and landing movie art, every picture of the base screens (`OPTSHPS.PAK`, and Righteous Fire's `OPTSHPS2.PAK`: the backdrops, the people, the animations, the shop and guild widgets), the Ship Dealer's and Software Booth's item icons (`SHIPSTUF.IFF`, `gunt/` and `soft/`, a broken item's Repair picture as `_broken`), the Software Booth's overlays and the Commodity Exchange's icons, named after the goods. The fonts and the other movies are still `.bin`.

A base picture is coloured with the palette of the screen that shows it. A few are shown on more than one screen in different colours (a figure that stands in several bars): they are written in the first screen's colours, and `--all-palettes` also writes them in each of the others, as `000.palt22.png` for palette 22. A few pictures appear on no screen at all; they get a palette that is a guess, marked `guessed` in the manifest, and the export prints a note naming them.

`DATA/OPTIONS/OPTIONS.IFF/scenes/` holds one JSON file per base screen: its palette, every picture it is built from with its place on the screen (the picture's hotspot goes there), its animated figures and their click areas, and its click areas with their labels ("Bar", "Launch", …). Each entry names the exported picture files, so a screen can be put back together from them.

A picture's frames that belong together (a ship's views, an animation, a portrait's eye and mouth frames) are all the same size, so they line up when stacked; a ship's views are named by the direction they show it from (`f15_e+00_a000.png`: level with the ship, from straight ahead). The game draws the views from 30 to 150 degrees mirrored for the other side of the ship. Pictures are indexed PNGs that keep the game's own colour numbers, which paint programs show like any other PNG; a few translucent ones (the jump point, the tractor beam's glow) are see-through as the game blends them, which only approximates how they look over a starfield.

`priv/ships/` and `rf/ships/` hold one folder per ship, everything a mod maker needs about it in one place. The four ships you can fly are `tarsus`, `orion`, `centurion` and `galaxy`; every other ship is named after its picture file (`GOTHRI`, `TALPIR`, `SALTHI`, …). Each folder has the ship's 37 views and afterburner flames (`hull/`, `burn/`), its targeting display pictures (`mfd_target/`) and `ship.json`: the ship's size, collision boxes, gun and launcher mounts, flight values and armour, and for every ship type the game flies it as, its guns, launchers, missiles, turrets, shields, energy and cargo, who flies it and the values the game works out from them, named as the mod files name them (`laser`, `missile_launcher`, `heat_seeker`, `Tail Turret`). The game does not read `ship.json`; it describes the ship.

Every ship that flies in the game's encounters also has `ship_mod.json5`: the same ship as a ship mod entry, every value as the game has it and its pictures taken from the folders beside it. Copied with its folder into a ship mod and named in that mod's `ships.json` (`{ ships: [ { file: 'GOTHRI/ship_mod.json5' } ] }`), it makes a ship exactly like the stock one, to change from there (see "Ship mods"); the four ships you can fly are flyable copies too, sold at the Ship Dealer. `ships.json` in `priv/ships/` and `rf/ships/` names all of them, so the whole folder copied as one ship mod brings a copy of every ship.

The folders of the four ships you can fly also hold:

- `death/`, `takeoff/` and `landing/`: the ship's explosion and its takeoff and landing animation;
- `cockpit/`: the cockpit views, gauges, `hud.json` and the damage marks (`decals/`);
- `repair/`: the Repair & Upgrade picture of the ship (`ship.png`, on the 320x200 screen) and every gun, launcher, engine and scanner picture of that screen cut out on its own (`hardpoint_2_laser.png` is a Laser on hardpoint 2), with `hardpoints.json` giving where each one sits on the screen and where the Ship Dealer's click areas for each mount and turret are. A gun picture with its `x` and `y` is exactly what a weapons mod's hardpoint image takes, and the `weapons_json` part of `hardpoints.json` is a `weapons.json` that puts every gun picture back where it was: copy it and the `repair/` pictures into a weapons mod and repaint the ones you want;
- `dealer.json`: the ship's price, the Ship Dealer items offered for it and those left out, its engine and shield levels, turrets and mount positions; its `shipstuff_json` part is a dealer-items mod that keeps all of that as it is, to start a mod that changes it.

`priv/weapons/` and `rf/weapons/` hold the guns and missiles for weapon mods (`weapons.json`, section 20): one folder per gun with its bolt animation (`bolt/`), its Weapons display icons (`mfd_forward.png`, `mfd_turret.png`), its Ship Dealer pictures (`shop.png`, `shop_broken.png`) and its firing sounds (`fire.wav`, `fire_far.wav` for the same shot from another ship); `guns.json` and

`ordnance.json` give every gun's, missile's and launcher's values in the mod file's own names and units; `stock_weapons.json5` is a `weapons.json` that states every one of those values as the game has it. Copied unchanged to `content/mods/weapons/<name>/weapons.json` it changes nothing: change the numbers you want and delete the rest.

`priv/dealer/` and `rf/dealer/` do the same for the Ship Dealer: `icons/` holds every item's picture and Repair picture, `rows.json` every item's prices, the ships it is sold for and what it does, and `stock_shipstuff.json5` is a dealer-items mod (`shipstuff.json`) stating all of it, the pictures included. Copy it with the `icons/` folder to `content/mods/items/<name>/` as `shipstuff.json` and it changes nothing until you edit it.

`priv/interiors/` and `rf/interiors/` hold every base's interior for interior mods (section 20): `interiors.json5` has one interior per base (`stock-<number>`) with every screen of the base (the landing pad, the concourse, the bar, the Ship Dealer and the rest) as its own picture, with its animations, its figures, its bartender, the Ship Dealer's purchases and its doors; the guild offices, the Mission Computer, the Commodity Exchange, Repair & Upgrade and the Software Booth are the game's own screens behind its doors. `art/` holds the pictures, `stock_roles.json` every base's screens and their click labels. Copied as `interiors.json` into an interior mod with its `art/` folder, a base or station naming one with `interior` and its original base with `like` gets that base's screens, without its story characters (as at every added base); change the pictures and doors you want from there. `overrides.json5` has one entry per base (`override-<number>`) that gives that real base a copy of its own screens: the same pictures, with the rooms where the story happens left as the original rooms. Copied into an interior mod with the pictures it uses, it shows at that base itself, ready to change.

`palettes/index.json` lists every palette of a game. `INITCFG.PAK` and `INITRF.PAK` are copied unchanged when they are found next to their archive. `manifest.csv` (and `manifest.json`) lists every file: the archive file and part it came from, how it was made, and where a replacement goes. A picture's row gives its size, its hotspot (the pixel the game places on the object's position) and the palette that coloured it; `palette_confidence` `engine-rule` means that is the palette the game draws it with, `guessed` that no screen shows the picture and `palette_rule` says how its palette was chosen, `none` marks a mask; `palette_alternatives` lists the other palettes of a picture several screens show. A sound effect's row names the weapons-mod field that takes a `.wav` like it, a Weapons display icon's or a Repair & Upgrade gun picture's row the field that takes its picture, and `dealer.json`'s row the dealer-items mod fields it restates.

Options:

- `--only LIST` exports only these kinds, comma separated: `images`, `cockpits` (the cockpit pictures and `hud.json`), `ships` (the `ships/` folders), `weapons` and `dealer` (the `weapons/` and `dealer/` folders), `interiors` (the `interiors/` folder: every original base's interior as an interior mod, with its pictures), `scenes` (the base screens' JSON), `data` (here `faces.json`), `sounds`, `speech`, `palettes`, `iff` (the data files' structure), `text` and `raw` (the `.bin` files); `all` (the default) and `modkit` (what mod makers need, the `ships/`, `weapons/`, `dealer/` and `interiors/` folders among it) are presets.
- `--game priv` or `--game rf` exports one game only (default: both).
- `--raw` also writes every archive file unchanged into `priv-raw/` and `rf-raw/`.
- `--rgba` writes every PNG as RGBA instead of indexed.

- `--all-palettes` also writes a base picture several screens show in different colours in each of them.
- `--overwrite` replaces an earlier export in the folder.
- `--verbose` prints one line per archive file.
- `--home`, `--data` and `--portable` work as for a normal start.

`--only` also accepts `music`, `fonts` and `movies`; this version writes no files for them.

The files in `priv-raw/` and `rf-raw/` sit at the same paths `content/priv/` and `content/rf/` use (section 20), so a file you change there can be copied straight into `content/` to replace the original.

The folder must be new or empty. Running the export again into the same folder needs `--overwrite`; the earlier export is replaced only once the new one is complete, and other files you put in the folder are left alone. The export refuses the folder of the game files, the program folder, the user-data folder itself (a folder inside it is fine) and anything inside `content/`. Ctrl+C stops the export; what was written stays in `priv.partial/` or `rf.partial/`, and the next run with `--overwrite` removes it.

Each game is written into `priv.partial/` (or `rf.partial/`) and renamed to `priv/` when it is complete. Another program that has the folder open (a virus scanner, an Explorer window, an editor watching the folder, a terminal inside it) can stop that rename; the export keeps trying for up to 30 seconds and says it is waiting. If the folder still cannot be renamed, it stays `priv.partial/`, which is complete, and the export goes on with the next game. The summary at the end lists such folders: close what has them open and rename them by hand (`priv.partial` to `priv`), or run the export again with `--overwrite`.

Tip: export into a plain folder of your own, not into a source code or build folder or the program folder. Editors, indexers and virus scanners often watch those and hold the new files; the export prints a note when the folder lies inside one.

`openprivateer --export-assets --help` lists the options. The exit code is 0 when the export is done, 1 for a mistake in the options, 2 when a game file is missing or cannot be read, 3 when the folder is refused, 4 when a file cannot be written, 5 when everything was written but a folder had to stay `.partial`, and 130 after Ctrl+C.

The exported files are made from your own game and are copyrighted: keep them for your own use and do not share or upload them.

26. Debug Tools

The Alpha contains debugging tools used during reconstruction and testing.

Open the settings/debug overlay with:

```
~
```

or the key chosen with `DebugOverlayKey=` (see section 5).

The settings on the left are grouped into sections, top to bottom: GENERAL, GAME, MODS, GRAPHICS, VIEW, COMMAND WHEEL, VOLUME, SUBTITLES, EXPERIMENTAL, PERFORMANCE, CONVERSATION

PLAYER and UPDATES. COMMAND WHEEL holds the control wheel's **Wheel Size** and **Wheel Transparency** sliders (see section 6). GENERAL starts with **Experience**. They start folded: click a section's header to open it, and again to fold it away; the game remembers this until you quit. MODS holds everything about mods: what this run did with each mod (folded under its own **This run** header, which opens the same way), the reload buttons, Auto reload, the mod manager button, the switches for each kind of mod, Init Mod and Hull Atlas. EXPERIMENTAL holds the testing switches, among them **Mouse Relative Warp** for the F8/F9 mouse look (see "Look around your ship in F8 and F9" in section 4); Reset graphics leaves it as it is. GRAPHICS also holds **Ship Meshes**, Off by default, a development switch for Enhanced and Modern only (Vulkan and Metal): **On** draws ships as 3D models from a model pack instead of their pictures, **Box** draws the other ships as a plain lit stand-in shape (a box with an orange nose). Leave it Off for normal play. A model pack holds models you supply from your own copy, converted with the tool in `tools/model_pack` (see its README); it goes in `content/models` in your OpenPrivateer folder, one file per ship type (and per base, missile or asteroid type) plus its `models.json`. A ship type without a model in the pack keeps its picture, and a ship shows its picture for a moment while its model loads. With On, your own ship is a model too when your pack (or its ships mod) has one for it: in the outside views, in the autopilot flyby and the jump's view from outside, and in the takeoff and landing movies, where it flies the path of the stock ship over the movie's starfield and base. Without a model these keep their pictures. A ships mod can bring models of its own ships (its `art.model`, in the mod's folder); a model in your own pack for the same ship is drawn instead. The same tool also builds models for most ships of the Wing Commander II and Academy ship packs from your own copy of a Wing Commander 3D model collection; they go in the same folder. Below Ship Meshes, **Mesh Lighting** lights the models from fixed directions of their own (**Ship**, like the pictures) or from the system's sun (**Sun**), and **Mesh Look** draws them in their own colours (**Textured**) or in the game's 256 space colours like the pictures (**Palette**). **Mesh Glow** sets the engine flames of the models: **Original** (the default) shows the afterburner flame of the pictures while the afterburner burns, a streak of white and blue (orange on the Kilrathi fighters) seen from the side and a bright ring seen from behind, and at normal throttle a faint short glow at the nozzles that grows with the throttle; **Afterburner** shows the afterburner flame only, as the pictures do; **Off** shows none. Ships whose pictures never show a flame (the capital ships, the Drayman, the Broadsword and the Steltek craft) show none as models either. **Mesh Objects** sets what else is drawn as a model: **All** (the default) also draws the bases and stations, the missiles in flight, the asteroids, the tumbling wreckage of destroyed ships and the cargo canisters floating in space as models when your pack has models for them, **Ships** only the ships. An object without a model keeps its picture (an ejected pilot always does). Up close a station fills the view as its picture did, and ships, shots and explosions behind a station or an asteroid are hidden by it. **Mesh Filter** (`MeshFilter=Trilinear`, `Bilinear`, or `Aniso2x` to `Aniso16x` in `openprivateer.ini`) sets how the models' own textures are smoothed as they get smaller with distance; it only matters with a model pack and Ship Meshes On, and it leaves the pictures alone (those follow Texture Filter). **Trilinear** (the default) blends smoothly between the textures' levels of detail; **Bilinear** switches between them, which can show a visible line where a surface changes level; **Aniso 2x** to **Aniso 16x** add anisotropic filtering on top of Trilinear, which keeps a surface you see at a flat angle, such as a long hull seen from behind, sharp instead of blurred, at a small cost on older graphics cards. Reset graphics sets Trilinear. Under them the overlay lists how many ships were drawn as models, the triangles, the flames, how many bases, missiles, asteroids, debris pieces and pods, the models loaded with their memory and the model pack's folder. PERFORMANCE holds **Performance log** (Off by default, `PerfLog=Off` or `On` in `openprivateer.ini`; starting the game with `--perf-log` turns it on for that run only). While it is on, the game writes how long each part of every frame takes to `logs/perf_<date>_<time>.csv` in your OpenPrivateer folder, once a second, and the section shows the

last second: the frame times, the parts that took the most time and counts such as draw calls and ships in flight. **Capture 600 frames** also writes the next 600 frames one by one to `logs/perf_frames_<date>_<time>.csv`. The log costs a little time itself; turn it off for normal play. The right-hand pane is the Mods log (see section 20).

Toggle the base click-region overlay with:

```
R
```

The region overlay draws clickable areas in green and is useful for screenshots when reporting a base UI problem.

27. Troubleshooting

NEW is greyed out

The matching INIT file is missing. Privateer needs `INITCFG.PAK`; Righteous Fire needs `INITRF.PAK`. Copy it in with the launcher (hold Shift while starting if it is turned off), or put it next to its `.TRE` file.

The game data cannot be found

The launcher lists every folder it searched and which files it found. Use **Import to Data Folder...** to copy files in, or **Add Folder...** to use them where they are. On Linux, a file named `priv.tre` is reported as "wrong letter case": rename it to `PRIV.TRE`.

With the GOG or Steam release, `PRIV.TRE` and `RF.TRE` are inside `GAME.GOG` (GOG) or `GAME.bin` (Steam): use **Extract from GOG/Steam...** under Game Data (section 1, "GOG or Steam installation"). If the launcher says the file is not a disc image it can read, is incomplete or damaged, check that you selected `GAME.GOG` or `GAME.bin` from the installation folder, and reinstall or re-download it (on Steam: verify the game's files) if the message stays.

The asset export stops with "no PRIV.TRE or RF.TRE found"

The export lists the folders it searched. Start the game once and add your game folder in the launcher's Game Data section, or pass the archive with `--tre FILE` (and `--rf FILE`) (section 25).

Where are my settings and saves?

Open the launcher's **Files and Diagnostics** section. It shows the user-data folder currently in use and provides buttons for the folder, savegames, logs, music, wallpaper and content.

An old save does not appear

Privateer saves must be `.SAV`, Righteous Fire saves `.PRS`. Put them in the `savegame/` folder of the user-data folder.

“libSDL3.so.0: cannot open shared object file”

On Linux this means the program found neither the bundled SDL3 nor a system SDL3. Usually the archive was not extracted completely: extract it again so the `libSDL3.so.0` file sits next to `openprivateer`, or install SDL3 3.2.16 or newer from your distribution.

The controller is detected but does not fly the ship

Open **Controls…**, check the device and calibration, then verify the axis bindings. A raw joystick needs at least two axes and one button.

The pointer does not reach part of the picture

Start the game once with `--input-diagnostics` (or put `InputDiagnostics=On` in `openprivateer.ini`, and `Off` again afterwards), move the pointer into every corner of the picture, quit, and send the newest session log. Its `[pointer]` lines record the window, its size in pixels, the picture’s place and every pointer position the game saw. Without the switch the log keeps the window lines and the first 20 pointer lines of a run.

The on-screen keyboard does not appear

The game asks for an on-screen keyboard (Steam’s, on a Steam Deck or Steam Frame) when you click or tap into a text box, or open the chat box of a multiplayer session, and closes it when you leave the box. If it does not come up, or it closed before you were done, click or tap into the box again. On a Steam Deck, Steam + X opens Steam’s keyboard yourself. If it still does not appear, quit and send the newest session log: its `[input]` lines record what the system reported about an on-screen keyboard and each time a text box asked for one.

MIDI is silent

Check the selected music backend. If using SoundFont MIDI, make sure a valid `.sf2` file is selected.

The game closes or crashes while loading

Start it once with `--gpu-debug` (or set the environment variable `OPENPRIV_GPU_DEBUG=1`):

```
openprivateer --gpu-debug
```

The graphics device then checks every graphics call and writes the graphics driver’s own error messages to the session log in the `logs` folder. Loading and play are slower in this mode, so use it only to investigate. Send that log with your report; its first lines also name the graphics driver in use (for example `GPU device driver: vulkan`).

When the game has to stop because of an error, it shows a message with the error and the path of the session log before it closes. If it crashes, the last line of the session log starts with `CRASH` and says what happened; send that log with your report.

The game does not start

If a message appears, it names what failed and where the session log is. The log folder is `logs` in your OpenPrivateer user-data folder (section 2, "Where your files live"); if that folder cannot be written, the log goes to an `OpenPrivateer/logs` folder in the system's temporary folder instead, and the message says so.

If nothing appears at all, start the game from a terminal (on Linux, in the folder that holds `openprivateer`):

```
./openprivateer --version

./openprivateer
```

and send what it prints along with the newest file in the log folder, if there is one. The first command only prints the version; if even it fails, the system cannot load the program (on Linux, the message names the missing library). OpenPrivateer needs Vulkan on Windows and Linux and Metal on macOS; without them it stops with a message saying so. Without a working sound device the game still starts, without sound.

The game does not start: "no usable graphics device was found"

On Windows and Linux the game tries more than one way to start the graphics card before it gives up, each written to the session log:

1. the normal way;
2. with the AMD switchable graphics Vulkan layer turned off;
3. with every implicit Vulkan layer turned off (these are add-ons other programs install into every Vulkan game: overlays, recorders, old driver parts);
4. with the power-saving graphics chip preferred;
5. with both of the last two.

The way that works is remembered, so the next start goes straight to it. If none works, the message lists what was tried and the graphics adapters it found.

A typical cause, on a laptop with an AMD processor graphics chip and an NVIDIA card, is an old AMD Vulkan layer that stops working after the NVIDIA driver is installed; every Vulkan game on that PC then fails the same way. The game turns that layer off by itself. To turn it off for every program, add a system variable (Windows: Advanced System Settings, Environment Variables, under System variables) named `DISABLE_LAYER_AMD_SWITCHABLE_GRAPHICS_1` with the value `1`, or start the game from a command prompt after typing:

```
set DISABLE_LAYER_AMD_SWITCHABLE_GRAPHICS_1=1
```

To turn every implicit Vulkan layer off instead:

```
set VK_LOADER_LAYERS_DISABLE=~implicit~
```

(On Linux: `DISABLE_LAYER_AMD_SWITCHABLE_GRAPHICS_1=1 ./openprivateer`.) Vulkan ignores `VK_LOADER_LAYERS_DISABLE` when the game runs as administrator, so start it normally. Updating the graphics driver (NVIDIA, AMD or Intel) also helps.

You can pick the way yourself in `openprivateer.ini`:

```
GpuStartup=Auto
```

`Auto` (the default) works as described above. `Default`, `NoAmdLayer`, `NoImplicitLayers`, `LowPower` and `Both` use only that one way, with no other attempt. `GpuStartupRemembered=` is the way Auto found working; the game writes it. For one start you can also set the environment variable `OPENPRIV_GPU_STARTUP` to one of the same words, for example `set OPENPRIV_GPU_STARTUP=NoImplicitLayers`.

If the game still does not start, send the newest file from the log folder: its `GPU:` lines show the adapters, the Vulkan layers and every attempt.

Missing pictures or graphics errors on Linux and macOS

The NVIDIA graphics driver on Linux keeps one open file per piece of graphics memory, and the game's pictures alone come close to the usual default limit of 1024 open files. So on Linux and macOS the game raises its own open-file limit at every start, as far as the system allows. The first lines of the session log show the result:

```
open file limit: soft 1024 hard 1048576 -> raised to 1048576
```

`already at` means the limit was high enough already. `could not raise` means the system refused; then raise the limit before starting the game, for example with `ulimit -n 65536` in the terminal you start it from (`ulimit -Hn` shows the highest value allowed), or with a `nofile` entry in `/etc/security/limits.conf`. On Linux the log also gives the number of open files at several points (`open file descriptors ...`).

A picture the graphics card cannot take is left blank and the game goes on; the log then has a warning with `SDL_CreateGPUTexture failed` or `SDL_CreateGPUTransferBuffer failed`. Send that log with your report.

Something behaves differently from DOS Privateer

Check the Experience mode first. Use **Classic** for direct comparisons with the original.

If the difference remains in Classic, report it.

Windows Defender reports the executable

New unsigned software can occasionally trigger heuristic antivirus detections. Do not blindly whitelist a file.

Make sure the binary came from the official OpenPrivateer release, then update Windows Security's security intelligence and scan it again. A signature update can clear such a detection.

If a current official build is still detected, report the build ID and exact detection name.

macOS says the program cannot be opened

The macOS packages are not signed by Apple, so macOS blocks the first start of a downloaded copy with a warning. Make sure the archive came from the official downloads page, then either:

- try to start `openprivateer` once, then open the security section of macOS's system settings and allow the blocked program there (the names of the settings, the section and the button differ between macOS versions); do the same for the `libSDL3` library if macOS names it next, or
- in Terminal, clear the download mark from the whole program folder:

  ```
  xattr -dr com.apple.quarantine /path/to/openprivateer-folder
  ```

macOS remembers the choice for that copy.

28. Reporting Bugs

A useful Alpha bug report includes:

- OpenPrivateer build ID
- Privateer or Righteous Fire
- Classic, Enhanced or Modern
- what you expected
- what happened
- exact reproduction steps
- whether it happens every time
- the savegame when relevant
- a screenshot or video for visual problems
- for startup or file problems, the text from the launcher's **Copy Diagnostics** button

Print the build ID with:

```
openprivateer --version
```

For base-screen problems, a screenshot with the `R` region overlay can help. For a slow or stuttering game, turn on **Performance log** in the debug overlay's PERFORMANCE section (section 26), play the part that is slow for a minute, turn it off again and send the newest `logs/perf_*.csv`.

You do not need to decide whether something is an OpenPrivateer bug or an original Privateer quirk before reporting it. Describe what happened and it can be checked against the original.

29. What Alpha Means

OpenPrivateer is playable, but it is still an Alpha.

The normal single-player Privateer path is implemented, and Righteous Fire support is under active reconstruction and testing. Obscure cases are still being found as more people play the game.

Expect bugs, unfinished polish and occasional behavior changes as the original game becomes better understood.

Save before doing anything adventurous.

30. Help and Downloads

Project website:

https://openprivateer.org/

Downloads:

https://openprivateer.org/#downloads

Discord:

https://discord.gg/5H3vW2STqW

When asking for help, include the build ID and whether you are playing Privateer or Righteous Fire.

31. Legal Note

OpenPrivateer is a reimplementation of the game engine.

It does not ship the copyrighted game data from *Wing Commander: Privateer* or *Righteous Fire*. You must provide the original game files yourself.

Wing Commander, *Privateer*, *Righteous Fire* and related names and assets belong to their respective rights holders.

32. One-Minute Checklist

For Privateer:

```
[ ] Extract OpenPrivateer

[ ] Start OpenPrivateer

[ ] Launcher: copy in PRIV.TRE and INITCFG.PAK

[ ] Leave Experience on Classic for original behavior

[ ] Click PLAY, then NEW
```

For Righteous Fire:

```
[ ] Launcher: copy in RF.TRE and INITRF.PAK

[ ] Select Righteous Fire
```

For an old pilot:

```
[ ] Launcher: Files and Diagnostics, Savegames

[ ] Copy .SAV or .PRS there

[ ] Click LOAD
```

For OpenPrivateer's extra features:

```
[ ] Switch to Enhanced or Modern

[ ] Open the settings overlay with ~

[ ] Enable the features you want
```

Have fun.