ENGLISH · 18 OCTOBER 2026

How skin casinos use the Steam session

Steam session tokens exported to gambling services, centrally controlled settlement, and CSGOFast's non-gambling claim. Code, policies and regulatory records.

Prepared	18 October 2026
Attribution	Agent Leon
Live article	phishdestroy.io/steam_dossier/trade-tracking
Reading files	PDFs, article JSON and verification

A dated reading edition of the published article. Original evidence, reported claims and editorial interpretation retain the distinctions and limits stated in the text.

READING MAP / 10 SECTIONS

Follow the record.

Article overview	3
Trade Token Sync 1.2.4	3
Empire receives an authentication credential from the Steam session	3
The browser permissions explain how the extension reaches that credential	6
Renewal makes the connection persist beyond a single token	7
Steam delivery and the casino’s payout are separate events	8
Roll’s older automatic extension and its current token popup are different packages	11
Fast’s own trading interface links SIH and checks its permissions	12
What the three implementations establish	13
Package. File. Location.	15
Read the underlying records.	21
Edition, sources & verification	26

The entries are clickable. The PDF bookmarks provide another route through the document. External source links open the referenced publication or retained file.

CSGOEMPIRE / CSGOROLL / CSGOFAST · REVIEWED 10 OCTOBER 2026

Inside the *Steam session.*

INVESTIGATION BY **AGENT LEON**
How casino trade tracking turns the user's logged-in account into part of the settlement system.

A skin can move directly between two Steam accounts while the commercial transaction remains under the casino's control. Empire's instructions explain why it wants access to the user's trade history: it uses the result to decide when the seller receives coins. Its extension shows how that access is obtained and renewed. Roll and Fast use different implementations, examined separately below.

Inspect the mechanism ↓Compare all three operators
ACQUIRED FROM GOOGLE'S UPDATE SERVICE

Trade Token Sync 1.2.4

`steamLoginSecurecookies · scripting · storage · tabs<all_urls>`
The official Empire guide links to this extension. The package identity and signatures were checked.

CODE EXAMINED / NO ACCOUNT CONNECTED
WHY THIS MATTERS TO OWNERSHIP
The ability to send a skin does not give the user control over the exchange. Steam must allow the transfer, and the outside operator must recognise it and release the return. The code and policies below identify the decisions that remain with those intermediaries. Read the physical-ownership comparison ↗

Original 11 October edition · complete text →Preserved original wording, alongside the current sections →

01 / THE CLAIM AND THE IMPLEMENTATION

Empire receives an authentication credential from the Steam session

Empire offers two ways to enable tracking: copy a token from an authenticated Steam page, or install the extension linked by its guide. Both routes are presented as ways to verify completed trades and release the seller's coins. [T01] [T02]

> “The token can't be used for anything else.”

CSGOEmpire · Trade Tracking (Desktop Guide) [T01]
The downloaded extension makes the handoff explicit. Its `background/steam.js` reads the `steamLoginSecure` cookie, separates the Steam account ID from the access token, and returns that token. The upload code can then send it unchanged to a registered partner endpoint. This is authentication material from the account’s existing session, not a receipt generated for one completed trade. [CT02]

That distinction is essential when assessing the assurance above. The reviewed path contains no exchange for a new credential restricted to one trade or history method. What the exported token can actually authorise is determined by Steam’s receiving endpoints. The code establishes the export; a claim that the credential is technically incapable of any other use requires evidence of those endpoint restrictions. Describing its intended use does not demonstrate its enforced limits. [CT05]

CODE PATH / TRADE TOKEN SYNC 1.2.4

One session. *A continuing connection.*

Illustrated from code
Not live account traffic
Read the login cookie, inspect the token’s expiry and check the registered recipient. A token close to expiry takes a separate refresh branch; a later cookie change can start another sync.

STEAM BROWSER SESSION

Login cookie

Steam ID + access token

`steamLoginSecure`Read the cookie finding

EXTENSION WORKER

Extract token. Check expiry.

Cookie permission + host access

The Steam credential is not narrowed.Inspect the extraction

REGISTERED RECIPIENT

Receive the token.

Recipient, account and freshness checks

`PUT → configured endpoint`Inspect the recipient gate

CONDITIONAL / BELOW 1 HOUR

Request a Steam refresh.

Inactive tab · closed after 5 seconds

`/jwt/refresh`Inspect the refresh branch

CONDITIONAL / NEXT PASS

If the cookie changes…

Schedule another sync after 10 seconds.

The refresh pass itself sends no token.Inspect the next trigger

PLATFORM SETTLEMENT

Trade verification.

The service checks the trade for its own settlement process.

Item delivery and coin release are separate.Follow the two records

REFRESH IS CONDITIONALThe illustration follows a successful cookie update into a later sync. A later sync reads the changed cookie; recipient and freshness checks govern the upload. CT03 · CT05

The extension extracts a Steam-issued token and can ask Steam to refresh the session before sending a newer token. The partner-registration checks and timing conditions are explained in the next section. CT02–CT05 ↗

WHY USE A BROWSER EXTENSION?

The browser permissions explain how the extension reaches that credential

MANIFEST HOST REQUEST**<all_urls>**
DESCRIBED TRADING TASK**Steam + participating sites**
The manifest requests access across all matching URLs, beyond the Steam and partner pages needed for the described transaction.

Scope requested by the package, subject to Chrome's restricted pages and the user's site-access settings.

cookies

Reads the Steam login cookie through the extension API.

scripting + tabs

Injects a page bridge into eligible open tabs and opens an inactive renewal tab.

content scripts

A universal message bridge is declared for all matching URLs.

trade editor

A Steam-page script adds specified items to the draft trade on either side.

An ordinary page script does not receive Chrome's privileged cookies API. An extension with the required cookie and host permissions can retrieve cookies through that API, including cookie records marked HttpOnly. Installing this package gives code a route into the Steam session that an ordinary visit to the casino's website does not provide. [T32] [T33]

The package also contains a script for Steam's trade-offer page. It adds specified items to the draft and redraws the offer, so the browser component performs a write operation as well as reading session data. This particular path does not submit or confirm the offer. These browser capabilities and the operations Steam accepts with an exported token are separate parts of the access model. [CT06]

02 / FOUR DIFFERENT CLOCKS

Renewal makes the connection persist beyond a single token

The guide's daily replacement instruction describes the manual route. The extension implements a renewal process driven by the token's expiry and browser events. Four timings appear in the instructions and code, and they perform different jobs. [T01] [CT03] [CT04]

GUIDE / MANUAL TOKEN**24 h**
The replacement cadence stated in Empire's instructions.

CODE / EXPIRY THRESHOLD**< 1 h**
Remaining lifetime that triggers a session-refresh attempt.

CODE / WORKER INTERVAL**10 min**
The interval registered by the worker for another sync.

CODE / COOKIE CHANGE**10 sec**
The delay after a matching login-cookie change before syncing.

The refresh branch opens Steam's `/jwt/refresh` route in an inactive tab, waits five seconds and removes the tab. A timestamp in session storage limits these refresh attempts to one per hour. A later cookie event drives the next sync. These are renewal instructions operating inside the user's existing browser session. [CT03]

Worker startup, page-triggered requests, a registered interval and matching cookie changes can all invoke synchronisation. Together they are designed to keep a partner supplied with a current token while the browser can obtain one. Expiry therefore limits the life of a particular credential without necessarily ending the commercial connection. [CT04]

STEAM JWT

The credential being exported.

Read from the Steam cookie. Its expiry is decoded locally. Steam remains the authority that accepts or rejects it.

PARTNER-REGISTRATION JWT

The extension's recipient gate.

A separate ES256-signed document identifies permitted domains and endpoint paths. It is checked against the extension's embedded verification key, issuer and expiry.

Before upload, the code checks a registered site's Steam ID and skips transfer if that site reports an equally new or newer token. The two JWTs in this process have different roles. The partner-registration signature authorises a recipient domain and endpoint; it does not change

the Steam token being sent. Recipient approval and Steam permissions are two separate controls in the same transfer. [CT05]

The page bridge can register a signed partner and trigger synchronisation; the extension handler contains no additional confirmation dialog for that registration. Removing a stored registration stops future uploads through that route. The removal function contains no call to Steam to revoke a token already supplied. Disconnecting a destination and invalidating its copy of a credential are therefore different events.

▼ **A separate startup issue: an empty partner list still reaches the cookie reader**

`enumerateDomainConfigs()` returns an object, but the worker tests its `.length` against zero. With an empty object, that early return is not taken. An offline fixture confirmed that startup still reads the synthetic Steam cookie. The actual upload routine enumerates zero destinations in this case; this finding concerns unnecessary access and possible refresh work, not an observed transmission. [CT07]

03 / THE ITEM AND THE MONEY FOLLOW DIFFERENT RULES

Steam delivery and the casino's payout are separate events

The purpose of this monitoring becomes clear in Empire's settlement rules. Its dedicated charging guide says a bid or purchase removes coins from the buyer's spendable balance immediately. The seller's payout follows a separate process, so delivery on Steam does not itself create an immediately usable balance for the seller. [T10]

TWO RECORDS
PURCHASE / TRANSFER
PROTECTION / REVIEW
SETTLEMENT
STEAM ITEM

Seller → buyer
The skin moves directly between their Steam accounts.

Reversal window
The buyer has the item while Steam protection applies.

Transfer verified
The platform checks completion and reversal status.

EMPIRE LEDGER

Buyer debited
Coins leave the buyer's spendable platform balance.

Seller payout pending
Empire says payout can take up to eight days. Special holds and credit exceptions have separate rules.

Seller credited
Empire releases coins according to its own settlement policy.

The trade-protection notice says the platform retains the buyer's funds during the reversal period. Its skin policy also permits a hold of up to a month for suspected stolen or scammed items. The seller can therefore deliver the skin before receiving spendable coins. Early-credit exceptions also depend on platform rules. The two users supply the item and the purchase balance, but Empire determines when the corresponding ledger entries become usable: a centrally administered C2C transaction with direct Steam delivery. [T06] [T07]

PRICING AUTHORITY

Seller asking prices sit inside the platform's pricing system

Empire derives a reference USD price from outside markets, chiefly BUFF, and converts it to Empire Coins. Sellers can edit their asking price. The platform controls the reference, conversion, auction routing and relisting conditions. That is centralised C2C market design. [T04] [T05] [T18]

ACCESS TO EARLY CREDIT

Playing is tied to a higher early-credit limit

The instant-credit scheme combines a tier allowance with half of the daily reward pool. Published tier allowances run from zero to 20,000 coins. Empire says increasing the limit requires progressing the account and playing; support cannot raise it manually. Reversed credited trades can produce deductions and a negative balance. [T09]

Failed-trade fees escalate.

FAILURES / SHARE OF ITEM OR BID VALUE
1
1%
2
2%
3
4%
4
8%
5
16%
6
32%
7+
50%

Published cap: 50%. Cooldowns rise to 50 hours; penalties reset after 48 hours without a failed trade. A minimum charge and auction price-lock also apply. [T08]

A Steam reversal can trigger a separate financial penalty

The skin policy allows permanent marketplace restrictions. If the buyer reverses, the seller can recover the item and keep up to the full coin payment. A support-approved exception usually uses a 30/70 split, adjustable by the operator. The published policy does not state which side keeps which share: the split is the operator's own remedy arithmetic, adjustable by the operator — publish the formula and every applied instance. [T07]

The operator also decides disputed outcomes

Support may request a screen recording, decide a dispute and apply exceptional treatment. The platform writes the rules, controls the ledger and decides how a dispute is resolved. Its own support process also decides whether an exceptional adjustment is warranted. [T07]

Ordinary fees and failure penalties follow different rules

The published zero-fee statement concerns routine skin deposits and withdrawals. Failed-trade penalties can still reach 50% of the item or bid value, and early-credit limits depend partly on play. A reader evaluating the cost and control of the service needs all three rules together. [T15] [T16]

▼ **The instructions conflict on protection, debit timing and cancellation**

Steam protection and the payout delay are blurred.

The mobile guide describes seven-day protection; the desktop guide calls it eight days. The protection notice describes seven days on Steam and up to eight days to receive coins. Users need the Steam deadline and the platform's additional delay stated separately. [T01] [T02] [T06]

Two instructions give different debit events.

The dedicated charging article says coins are removed immediately on a bid or withdrawal. The general withdrawal guide describes deduction after receiving the item. The ledger event should be consistent across both instructions. [T10] [T11]

Old and new cancellation rules remain side by side.

An April 2024 article says buyers cannot cancel an active withdrawal. A July 2026 article permits some normal withdrawals to be cancelled after thirty minutes, with tracking enabled and before the seller sends. Both remain in the collection. The collection leaves the reader to reconcile them. [T12] [T13]

Penalty review remains inside the same platform.

The punishment guide permits users to ask support to review an unfair penalty. The skin policy allows corrections when the platform recognises an error. Both remedies are administered by the operator controlling the balance. [T08] [T07]

The business no longer needs to hold every skin in its own bot inventory. Empire describes direct user delivery alongside seller listing automation. Its privacy feature can hide item identifiers from prospective buyers. The participant's inventory now supplies the delivery route. A count of banned accounts cannot show who lost their skins or whether the operator's business was stopped without identifying those account categories. [T17] [T14] See the account-count comparison ↗

04 / CSGOROLL'S TWO EXTENSION IDENTITIES

Roll's older automatic extension and its current token popup are different packages

The June 2024 announcement describes automatic token collection, expiry handling and background transmission. Its link targets an older extension ID. Roll's current short URL points to a different package that still extracts a Steam token for the user to hand over. [T27] [T28]

2024 ANNOUNCEMENT / OLDER ID

CSGORoll extension

`cgkgfnlnpcifjnbfdbmcphcgnkeinjpd` **UPDATE RESPONSE: HTTP 204**
Google's update service returned no package in this check. The store URL resolved to an empty-title shell without an extension listing. [T29] [T46]

CURRENT REDIRECT / DIFFERENT ID

Steam WebAPI Token Extension

`bdgacfnoihldeemdcbggkgmjgdfcliah` **UPDATE RESPONSE: HTTP 200 · v1.2**
The current package was returned again with the same hash as the earlier audit. It extracts a token in a popup for copying. Its registered manifest has no background worker or daily scheduler; current help describes pasting the token into Roll. [T30] [T31] [T42]

The current help still asks the user to supply a Steam token to Roll. In version 1.2 the final handoff is manual, whereas the older announcement described background collection and upload. The distinction changes the technical mechanism, not the stated purpose: Roll obtains account information to recognise trades and administer its own balance. [T31] [T42]

05 / CSGOFAST: THE CURRENT CONNECTION AND THE IDENTITY CLAIM

Fast’s own trading interface links SIH and checks its permissions

The connection is in CSGOFast’s own shipped application. Its trading interface links to SIH, checks the extension’s connection and permission state, and asks the platform backend whether the user can trade. [T38] [T39]

1. **A named installation target**The installation component links to Steam Inventory Helper’s Chrome extension ID `cmeakgjggjdlcpncigglobpjbkabhmjl`. The modal loader imports this component through a separately fetched module. [CF01]
2. **A connection-and-permission condition**The interface reads `connections.SIH` and checks both `online` and `permission`. The check button and warning state depend on those values. That is a concrete dependency in the shipped trading interface. [CF02]
3. **A platform decision endpoint**The client requests its own backend’s `/api/market/steam/check-user` route and consumes the resulting permission state. Fast’s own frontend establishes the integration. The linked SIH audit traces the extension’s credential retrieval and server-command mechanisms. [CF03] [T42]

> “The Site does not present itself as a gambling platform.”

CSGOFast · Responsible Play Policy §8.1 [T35]

ITS OWN DESCRIPTION

The non-monetary label sits beside paid funding routes

The same policy denies real-world value to items and Credits. Its FAQ describes funding by skins, partner gift codes, cryptocurrency and card-mediated crypto purchases, followed by a balance credit. The user incurs a real acquisition cost even where the resulting unit has restricted redemption rights. The operator’s description of its credit does not establish that the surrounding business is free of monetary value or payment relationships. [T35] [T37]

AN EXTERNAL LEGAL RECORD

The same domain appears in a final Dutch gambling-enforcement record

In April 2025, the Dutch Ksa ordered Gamusoft LP to stop unlawful gambling offered through csgofast.com. Its published record says the order became final and violations continued. Fast’s current non-gambling label cannot rewrite that enforcement record. [T41]

THE CONSEQUENCE FOR THE USER

Fast asks users to explain their identity and finances while its own account of the service remains inconsistent. The non-gambling presentation, funding instructions and AML demands must be read together. They describe an operator that controls access to the user’s balance

and requests evidence about the real assets used to fund it. [T36]

Fast's AML and privacy pages display a 2024 revision date while naming Lumigrid OÜ, registered in September 2026. The AML text pairs that Estonian company with UK law. It even treats unusual attention to its AML policy as a suspicion indicator. The user is expected to explain their finances to a service whose own disclosures contain these unresolved defects.
Read the documentary findings ↗

NEW CASE / VALVE'S 2016 ENFORCEMENT CLAIM

CSGOFast, SIH and the commercial control that survived

Five interface exhibits, 33 code locations, authenticator custody and the record before the commission.

↗

▼ **Read the original 11 October wording**
Preserved verbatim from the supplied 11 October 2026 edition. "Today" and "currently" refer to that edition; its original estimates and formulations remain available alongside the dated text above.

Four interface exhibits, 33 code locations, authenticator custody and the record before the commission.

Open this passage in the complete original edition →

06 / THREE OPERATORS, SPECIFIC IMPLEMENTATIONS

What the three implementations establish

API means an interface. A publisher key, a user API key and a session JWT are different credentials. None of these browser workflows requires the user to be a game developer.
Original three-column credential comparison ↗

01 / ACCOUNT ACCESS

CSGOEmpire

IDENTIFIED PACKAGE
Trade Token Sync 1.2.4, linked by Empire's guide. [T24]

ACQUISITION
Reads the Steam login cookie and extracts its JWT. Manual help uses Steam's token page.

RENEWAL
Expiry threshold, cookie event, worker startup and ten-minute interval; inactive Steam refresh tab.

CONTROL ESTABLISHED

A session credential leaves the browser through a developer-approved recipient route. The platform controls payout, holds and sanctions.

02 / ACCOUNT ACCESS

CSGORoll

IDENTIFIED PACKAGE
Current token-popup extension 1.2; older automatic extension has a different ID. [T27] [T28]

ACQUISITION
Current popup obtains the WebAPI token from authenticated Steam content, then offers a copy action.

RENEWAL
Automatic renewal is described in the older announcement. It is absent from the current package's registered entry path.

CONTROL ESTABLISHED
The current workflow still hands a Steam token to Roll. Manual copying changes the handoff, not who receives the credential.

03 / ACCOUNT ACCESS

CSGOFast / SIH integration

IDENTIFIED CONNECTION
Current site code links the SIH ID and checks its connection and permission. [CF01]

ACQUISITION
SIH's reviewed worker can supply a Steam WebAPI token in response to a server request; Fast's frontend establishes the connection flow.

RENEWAL
The earlier SIH audit found retry-on-403 and cached token retrieval, not a fixed daily scheduler.

CONTROL ESTABLISHED
The trading interface checks SIH connection and permission; Fast's backend supplies the trading-permission state.

The technical evidence explains the dependence that the collectibles analogy omits

These implementations differ in extraction, renewal and transfer. Empire's code exports a Steam session token through an approved-partner mechanism; Roll's current package prepares a token for manual submission; Fast's own interface links an SIH permission flow. They show concrete ways in which a user's Steam account becomes part of an outside service's verification process.

The settlement rules explain why that access matters. Steam determines whether the item can move, while the casino determines how the recognised transfer changes its own balance. Physical ownership is a misleading model for this dependence: neither possession in a Steam inventory nor delivery to another user removes the need for those continuing permissions. Valve’s response and the ownership comparison ↗

▼ **Trade Token Sync’s linked privacy notice does not explain this credential lifecycle**
The Chrome listing declares handling authentication information and links to a guest Pastebin privacy notice dated 5 August 2025. That notice discusses broad website data, profiling, partners and session replay, and still refers to EU–US Privacy Shield. It does not set out a token-specific retention period, the signed partner registry, the background renewal mechanism or deletion of uploaded credentials. The missing explanation concerns this extension’s actual data flow. [T24] [T26] Why the transfer-safeguard reference matters ↗

What the operators still owe their users

1. Publish the Steam methods and account data accessible with the exported token, and demonstrate any claimed history-only restriction.
2. Explain why all-URL host access and a universal page bridge are necessary for the stated trading task.
3. Identify the partner-registration signer, current recipients, approval process and revocation mechanism.
4. Disclose token retention, employee access, server request logs and deletion after a site is disconnected.
5. Provide a consistent timeline for buyer debit, Steam protection, platform holds, early credit and final payout.
6. For CSGOFast, identify the SIH permissions used and reconcile the non-gambling policy with the payment route, AML demands and named enforcement record.

07 / REPRODUCIBLE CODE RECORD

Package. File. Location.

The findings below pin original filenames, hashes and search anchors. The downloadable record includes acquisition responses and the review of all 22 articles in Empire’s linked collection.

Download evidence record ↗Manifest ↗

▼ **CT01Browser authority extends beyond Steam history**
The manifest requests cookies, scripting, storage and tabs, with all-URL host access. It registers a universal content script and a Steam trade-offer script. The worker also injects the universal bridge into eligible already-open HTTP(S) tabs. The universal bridge extends the extension’s reach beyond a single casino tab.

empire / manifest.json

Original line 12 · UTF-8 byte 294

Find: `"permissions"`

SHA-256 367cebed484e3007511fefed990fffbffa6509f6b2ca1354696a75fb3953931f

empire / background/service-worker.js
Original line 39 · UTF-8 byte 1426

Find: `injectContentScriptIntoExistingTabs`

SHA-256 ad5caa9c9974f56bc28de449ad7a8b8351810c2c7a7c22b491f243acadb6316f

▼ CT02The exported credential is extracted from the Steam login cookie
The cookie reader requests steamLoginSecure using chrome.cookies.get, URL-decodes its value and separates the Steam ID and access token. It decodes expiry and returns the token without a narrowing exchange. With a synthetic two-hour token, the original function returned the identical token.

empire / background/steam.js
Original line 33 · UTF-8 byte 910

Find: `async function getSteamLoginSecureCookie`

SHA-256 6136673df2e6bdc4b3fc8a7a2f8f67fa4b84810da704e4a39577454c0e248490

▼ CT03Session renewal uses an inactive Steam tab
If expiry is less than one hour away, getSteamAccessToken calls refreshSteamSession. The function throttles attempts to once an hour through session storage, opens an inactive Steam /jwt/refresh tab, waits five seconds and removes it. The read returns null on this pass; the intended next sync follows a cookie update. This behavior was exercised with invented cookies and mocked browser APIs.

empire / background/steam.js
Original line 19 · UTF-8 byte 510

Find: `jwtExpiresAt.getTime()`

SHA-256 6136673df2e6bdc4b3fc8a7a2f8f67fa4b84810da704e4a39577454c0e248490

empire / background/steam.js
Original line 45 · UTF-8 byte 1334

Find: `async function refreshSteamSession`

SHA-256 6136673df2e6bdc4b3fc8a7a2f8f67fa4b84810da704e4a39577454c0e248490

▼ CT04Daily expiry, periodic checks and cookie events are distinct
The worker calls doSteamSync at startup and registers a ten-minute interval. A matching Steam login-cookie change schedules another sync after ten seconds. An authorised page may also request a sync; that page-triggered path is throttled to once a minute. Chrome worker suspension was not simulated.

empire / background/service-worker.js
Original line 36 · UTF-8 byte 1242

Find: `setInterval(doSteamSync`

SHA-256 ad5caa9c9974f56bc28de449ad7a8b8351810c2c7a7c22b491f243acadb6316f

empire / background/service-worker.js
Original line 42 · UTF-8 byte 1526

Find: `chrome.cookies.onChanged`

SHA-256 ad5caa9c9974f56bc28de449ad7a8b8351810c2c7a7c22b491f243acadb6316f

empire / background/injected-api.js
Original line 120 · UTF-8 byte 3247

Find: `async function triggerSync`

SHA-256 0095ac14d4175b783f2160980ca881e3c5627a8f36004b59cadf6feacf5eeab4

▼ **CT05Signed partner configuration governs where the credential goes**
A distinct ES256 registration JWT is checked using an embedded key, issuer, time claims, domain patterns and endpoint paths. The extension stores valid configurations in sync storage. Upload checks a reported Steam ID and token expiry, then PUTs the access token to the configured endpoint. setDomainConfig has no extension-owned confirmation dialog. clearDomainConfig removes the registration without calling a Steam revocation route. These checks control recipients; they do not narrow the Steam credential.

empire / background/jwt-verifier.js
Original line 8 · UTF-8 byte 313

Find: `export async function verifyDomainJwt`

SHA-256 4b34276f5510aa4c72939b6eafe8e4021d943ff199e425048903effa8930392c

empire / background/partner-sites.js
Original line 36 · UTF-8 byte 999

Find: `export async function uploadNewAccessToken`

SHA-256 96581ffe9c2dcad46d4c1090bb5cbb5790891df0d0bbf162b82bfb402cadf53d

empire / background/injected-api.js
Original line 67 · UTF-8 byte 1607

Find: `async function setDomainConfig`

SHA-256 0095ac14d4175b783f2160980ca881e3c5627a8f36004b59cadf6feacf5eeab4

empire / background/injected-api.js
Original line 113 · UTF-8 byte 3037

Find: `async function clearDomainConfig`

SHA-256 0095ac14d4175b783f2160980ca881e3c5627a8f36004b59cadf6feacf5eeab4

▼ **CT06The Steam-page script modifies the draft offer**
The injected script reads item identifiers from URL parameters, adds missing entries to either side of g_rgCurrentTradeStatus and redraws the draft. No submit or confirmation call appears in this file. This is a separate interface-writing capability from trade-history access.

empire / content-scripts/tradeoffer.js
Original line 12 · UTF-8 byte 246

Find: `const theirItems`

SHA-256 32481abc873f43965b41cebfff93ebdc9d0dee82ac6027fbd8f5d544e23de1af

empire / content-scripts/tradeoffer.js
Original line 62 · UTF-8 byte 1368

Find: `window.RedrawCurrentTradeStatus`

SHA-256 32481abc873f43965b41cebfff93ebdc9d0dee82ac6027fbd8f5d544e23de1af

▼ **CT07The empty-registration early return uses the wrong data shape**
enumerateDomainConfigs returns an object. doSteamSync checks its length as though it were an array. In an offline empty-object fixture, the worker still invoked the cookie reader. The real upload implementation enumerates zero domains in that situation. The recorded defect is the worker's object-versus-array test and the resulting cookie-read invocation.

empire / background/service-worker.js
Original line 14 · UTF-8 byte 540

Find: `domainConfigs.length === 0`

SHA-256 ad5caa9c9974f56bc28de449ad7a8b8351810c2c7a7c22b491f243acadb6316f

empire / background/partner-sites.js
Original line 96 · UTF-8 byte 3080

Find: `const output = {}`

SHA-256 96581ffe9c2dcad46d4c1090bb5cbb5790891df0d0bbf162b82bfb402cadf53d

▼ **CF01The current Fast frontend contains an SIH installation path**
The SIH component points to Chrome extension ID cmeakgjggjdlcpncigglobpjbkabhmjl. The main application references the public component chunks; its modal registry imports chunk-WXK6SRSN.js, which imports this component. This places the SIH connection flow in CSGOFast's own frontend. The dedicated case follows that integration into SIH's registered account-operation handlers.

fast / chunk-4J5LYWSH.js
Original line 3 · UTF-8 byte 1338

Find: `//chrome.google.com/webstore/detail/steam-inventory-helper`

SHA-256 4ed0af788a6ef319c3f8bfb55f06252f16ff705d50bb10da2eae05da54c057bf

fast / chunk-CJLFSPQI.js
Original line 10 · UTF-8 byte 74042

Find: `SIH_INSTALL`

SHA-256 6fa061bbe3343c763ab641c1a71126558a40c1add657788b33d6f10164eb0f56

▼ CF02Fast checks SIH online state and permission
The component reads p2pPermissions.connections.SIH. Its disabled and warning state depends on whether the connection exists, is online and has permission. The dialog closes when both online and permission become true. The general P2P component also reads this connection state.

fast / chunk-4J5LYWSH.js
Original line 3 · UTF-8 byte 1743

Find: `checkBtnDisabled$`

SHA-256 4ed0af788a6ef319c3f8bfb55f06252f16ff705d50bb10da2eae05da54c057bf

fast / chunk-4OFQKOVU.js
Original line 6 · UTF-8 byte 13931

Find: `connections?.SIH`

SHA-256 6ab7ee0bbecb2fc844689f73131343e588bd3cf4850156ee99d96eb6469a57da

▼ CF03The Fast client asks its backend for Steam-trading permission state
The client service has getP2pPermissions requesting /api/market/steam/check-user with credentials. Another component consumes canSteamAPI and a steamApiKey field. These names establish the frontend contract, not the actual format of a user's secret or the backend's Steam access. No authenticated endpoint was called in this review.

fast / chunk-2FZZMIT7.js
Original line 3 · UTF-8 byte 182042

Find: `getP2pPermissions=t⇒`

SHA-256 0eee2c9e863b98d6aed571d198a78c76cf4d4100b7219208b6530b1cbf2a3309

fast / chunk-ONY24YIU.js
Original line 3 · UTF-8 byte 8322

Find: `apiKey$=`

SHA-256 f6a72f25785c3772f47f7e2557251c89e54683358d96a4b89ac7ee24566d9606

▼ All 22 Empire trading articles reviewed

- T01 · Trade Tracking — Desktop ↗
  Manual token handoff, daily replacement, automatic checks and extension referral.

- T02 · Trade Tracking — Mobile ↗
  Manual Steam-token page and seven-day protection wording.
- T03 · How the P2P system works ↗
  User-to-user delivery, seller prices, deadlines, pending payouts and disputes.
- T04 · How prices are calculated ↗
  External market inputs, USD reference and conversion to Empire Coins.
- T05 · Editing a listing price ↗
  The seller can change the asking price and percentage.
- T06 · Trade protection update ↗
  Seven-day Steam protection, up-to-eight-day coin payout and held funds.
- T07 · Skin Trading Policy ↗
  Reversals, coin allocation, exceptional holds and support discretion.
- T08 · Trade punishments ↗
  Escalating fees, cooldowns, price-locks and reversal restrictions.
- T09 · Instant Credit Limit ↗
  Tier allowances, reward-pool component, play-based progression and recovery.
- T10 · When an item is charged ↗
  Immediate removal of spendable coins on a bid or withdrawal.
- T11 · Withdrawing Counter-Strike items ↗
  Withdrawal workflow and its different debit wording; seller-listing bots.
- T12 · Cancelling a withdrawal ↗
  July 2026 conditional cancellation with tracking enabled.
- T13 · Cancelling an active trade ↗
  April 2024 no-cancellation wording still linked from the collection.
- T14 · Item privacy toggle ↗
  Hiding item identifiers and simplifying seller ratings.
- T15 · Trading fees ↗
  Zero routine skin-deposit and withdrawal fee statement.
- T16 · Wagering to withdraw skins ↗
  No wagering requirement stated for skin withdrawals.
- T17 · Withdrawal duration ↗
  Auction and regular deadlines; no site-owned bots and seller automation.
- T18 · Auction listings and limits ↗
  Routing thresholds and relisting-price rules.
- T19 · Thirty-minute auction update ↗
  Dated change to buyer and seller auction deadlines.

- T20 · Depositing Counter-Strike items ↗
  Listing inventory and completing the transfer when another user withdraws.
- T21 · Telegram and Discord notifications ↗
  Linked notification accounts, event categories and unlinking.
- T22 · Updating a trade URL ↗
  Trade URL settings; distinct from a session credential.

▼ **Method and technical boundaries of this review**

We reviewed public documents and downloadable client code, verified the Empire CRX signatures and extension-ID key match, and ran six isolated fixture checks. No extension was installed, no account connected and no live token obtained or replayed. Fixture results ↗

The code establishes extraction, renewal instructions, transmission paths and client-side integration. Server retention, actual production commands and Steam's accepted operation set were not measured. Browser permissions and Steam-token permissions are separate; an unchanged exported token does not, by itself, establish every operation Steam will accept.

The ten-minute interval is registered in code. Chrome can terminate an idle service worker, so it is not a measured uptime guarantee. Startup, page requests and cookie events provide further sync triggers. [T34] The trade-offer script edits the draft but does not submit or confirm it.

The older Roll package returned HTTP 204, and its store link returned an empty listing shell. Neither response supplies Google's reason. The current Roll popup package, the older announced automatic workflow and Empire's renewal code are identified separately throughout the record.

The Dutch order concerns Gamusoft LP and csgofast.com in 2025. Its named operator, date and jurisdiction remain attached to the finding; the separate identity investigation examines the companies named in Fast's current documents.

08 / SOURCE REGISTER

Read the underlying records.

Operator instructions, published policies, Chrome documentation, official package delivery and identified frontend files. Retrieval timestamps and SHA-256 values are in the evidence download.

▼ **Open the full source register**

1. T01
   CSGOEmpire — Trade Tracking — Desktop ↗
   Manual token handoff, daily replacement, automatic checks and extension referral.
2. T02
   CSGOEmpire — Trade Tracking — Mobile ↗
   Manual Steam-token page and seven-day protection wording.

3. T03
   CSGOEmpire — How the P2P system works ↗
   User-to-user delivery, seller prices, deadlines, pending payouts and disputes.
4. T04
   CSGOEmpire — How prices are calculated ↗
   External market inputs, USD reference and conversion to Empire Coins.
5. T05
   CSGOEmpire — Editing a listing price ↗
   The seller can change the asking price and percentage.
6. T06
   CSGOEmpire — Trade protection update ↗
   Seven-day Steam protection, up-to-eight-day coin payout and held funds.
7. T07
   CSGOEmpire — Skin Trading Policy ↗
   Reversals, coin allocation, exceptional holds and support discretion.
8. T08
   CSGOEmpire — Trade punishments ↗
   Escalating fees, cooldowns, price-locks and reversal restrictions.
9. T09
   CSGOEmpire — Instant Credit Limit ↗
   Tier allowances, reward-pool component, play-based progression and recovery.
10. T10
    CSGOEmpire — When an item is charged ↗
    Immediate removal of spendable coins on a bid or withdrawal.
11. T11
    CSGOEmpire — Withdrawing Counter-Strike items ↗
    Withdrawal workflow and its different debit wording; seller-listing bots.
12. T12
    CSGOEmpire — Cancelling a withdrawal ↗
    July 2026 conditional cancellation with tracking enabled.
13. T13
    CSGOEmpire — Cancelling an active trade ↗
    April 2024 no-cancellation wording still linked from the collection.
14. T14
    CSGOEmpire — Item privacy toggle ↗
    Hiding item identifiers and simplifying seller ratings.
15. T15
    CSGOEmpire — Trading fees ↗

Zero routine skin-deposit and withdrawal fee statement.

16. T16
    CSGOEmpire — Wagering to withdraw skins ↗
    No wagering requirement stated for skin withdrawals.

17. T17
    CSGOEmpire — Withdrawal duration ↗
    Auction and regular deadlines; no site-owned bots and seller automation.

18. T18
    CSGOEmpire — Auction listings and limits ↗
    Routing thresholds and relisting-price rules.

19. T19
    CSGOEmpire — Thirty-minute auction update ↗
    Dated change to buyer and seller auction deadlines.

20. T20
    CSGOEmpire — Depositing Counter-Strike items ↗
    Listing inventory and completing the transfer when another user withdraws.

21. T21
    CSGOEmpire — Telegram and Discord notifications ↗
    Linked notification accounts, event categories and unlinking.

22. T22
    CSGOEmpire — Updating a trade URL ↗
    Trade URL settings; distinct from a session credential.

23. T23
    CSGOEmpire — Counter-Strike deposits and withdrawals ↗
    All 22 linked articles acquired and read.

24. T24
    Trade Token Sync — Chrome Web Store ↗
    Direct response on 10 October: version 1.2.4, updated 28 September 2026; a search-engine rendering still showed 1.2.3.

25. T25
    Trade Token Sync — Google update-service package ↗
    Version 1.2.4; CRX hash and signature checks pinned below.

26. T26
    Privacy notice linked by Trade Token Sync ↗
    Guest Pastebin notice dated 5 August 2025; credential-specific disclosure examined.

27. T27
    CSGORoll — WebAPI Trading Extension ↗

25 June 2024: automatic collection, renewal and background transmission; older extension ID.

28. T28
    CSGORoll — current extension redirect ↗
    Resolved to Steam WebAPI Token Extension, a different ID.

29. T29
    Older CSGORoll extension — Google update response ↗
    HTTP 204 and zero bytes in this acquisition; no reason supplied.

30. T30
    Current CSGORoll extension — Google package ↗
    HTTP 200; version 1.2; unchanged package hash from the earlier audit.

31. T31
    CSGORoll — Steam WebAPI Token help ↗
    Current help instructs copying and pasting a token.

32. T32
    Chrome — Declare permissions ↗
    Host permissions, browser API capabilities and optional access.

33. T33
    Chrome — Cookies API ↗
    Cookie and host permission requirements; cookie fields including HttpOnly.

34. T34
    Chrome — Extension service worker lifecycle ↗
    Idle termination and event-driven worker revival; a timer is not guaranteed uptime.

35. T35
    CSGOFast — Responsible Play Policy ↗
    No-gambling representation and no-real-world-value wording.

36. T36
    CSGOFast — AML Policy ↗
    User verification, document demands and asserted authority.

37. T37
    CSGOFast — FAQ ↗
    Skin deposits, partner vouchers, cryptocurrency and card-mediated crypto funding.

38. T38
    CSGOFast — P2P frontend component ↗
    Reads SIH connection state in the trading interface.

39. T39
    CSGOFast — SIH installation and permission component ↗
    Official SIH extension link and online/permission checks.

40. T40
    CSGOFast — client service bundle ↗
    Requests the platform's /api/market/steam/check-user endpoint.

41. T41
    Dutch Ksa — Gamusoft LP / csgofast.com ↗
    April 2025 unlawful-gambling order; final status and continued violations recorded.

42. T42
    PhishDestroy — earlier Roll and SIH code audit ↗
    Prior acquired packages, code findings and hashes; this is our audit record.

43. T43
    CSGOFast — public application entry bundle ↗
    Directly references the reviewed frontend chunks.

44. T44
    CSGOFast — SIH modal import module ↗
    Connects the public modal loader to the SIH component.

45. T45
    Chrome — Content scripts ↗
    Page access, script injection and messaging with extension components.

46. T46
    Older CSGORoll extension — store URL ↗
    Direct request resolved to an empty-title shell without a listing.

EDITION NOTES / TRACEABLE FILES

Read the claim. Check its source.

This reading edition was prepared on 18 October 2026 from the current English article. It retains the authored discussion and analytical tables in the source HTML. Navigation and interactive controls are removed. Interactive datasets remain linked rather than expanded into thousands of printed rows.

Selected source images remain linked to their publications or retained copies. The PDF does not independently authenticate the participants in a screenshot, verify every report in a collection, or determine the merits of a legal allegation.

Check this edition

The detached OpenPGP signature covers the exact bytes of the article JSON. That JSON includes the exported text, source identity and SHA-256 of this PDF. It is not a PDF document signature, and it does not certify the truth of underlying third-party material.

Article JSON · OpenPGP signature · JSON SHA-256
Published verification key · Verification instructions

```
Signing-key fingerprint
129FC7F39C40C69D2EF3C329BFCF9E8DD295EE01

Source HTML SHA-256
fc1c459577573c11aaf06d8b3dfee3e92048c7900dee2587209083e5dbb1092b
```

Reuse and attribution

Authored research is released under CC0 1.0 Universal. Third-party statements, screenshots and other source material retain their own rights and attribution. The original rights statement is a separate signed object.

```
Build 2026-10-18T11:58:30.589Z
Source: https://phishdestroy.io/steam_dossier/trade-tracking
```