

Section 3.1

Inside Steam, Valve's list is the only list

Investigated by: Agent June

Inside Steam, Valve's list is the only list

Inside Steam, Valve's list is the only list.

A link clicked in the Steam client, the in-game overlay or Big Picture mode does not open in the user's browser. It renders inside Steam's own embedded browser — where the warning layers a browser would apply are not present. **The Link Filter is not one safeguard among several. It is the whole of it.**

In a normal browser

Three independent layers

1. **Google Safe Browsing**A full-page interstitial for hosts on Google's phishing and malware lists, updated continuously and independent of the site being visited.
2. **The antivirus browser extension**Most consumer endpoint products ship a browser add-on that checks the destination against the vendor's own reputation list.
3. **Endpoint HTTPS inspection**Where the product performs TLS inspection, page content is scanned before it reaches the user.

In the Steam client, overlay and Big Picture

One layer

1. **Google Safe Browsing**Not present. The check requires Google API keys that are not provisioned in embedded Chromium builds; where they are absent, Chromium's own design documentation records that the Safe Browsing handler is simply never added to the request chain.
2. **The antivirus browser extension**Not present. Steam's renderer is not the user's browser and loads no extension from the user's browser profile, so no add-on runs against the page.
3. **Endpoint HTTPS inspection**Product-dependent. The traffic terminates inside a code-signed Valve process rather than a recognised browser, and a number of endpoint products exempt signed applications from TLS inspection.
4. **The Steam Link Filter**Valve's own blocklist, consulted when a link leaves the Steam ecosystem. This is the sole check that runs — and the table above records what it returns for the auth hosts documented in this section.

A test any reader can run, and the finding falls if it fails

Open one of Google's published Safe Browsing test URLs (testsafebrowsing.appspot.com) in Chrome: a full-page red interstitial appears. Open the same URL from a Steam chat message, so that it renders in the Steam client or the overlay: no interstitial appears. The only warning Steam can produce is its own Link Filter page, and only for hosts on Valve's list.

Valve's own bug-bounty record

The one layer has been got around before — on Valve's own record

If the Link Filter is the only check that runs inside the client, its own failure modes are not a side issue. Two reports on Valve's HackerOne programme, both accepted, both resolved, both since disclosed in full:

1. [#291750 archived 2025-11-16 ↗](#) Open Redirect · medium

Link filter protection bypass

Reported 19 November 2017, resolved, disclosed 9 May 2018. The report's own proof of concept: substituting the ideographic full stop 。（%E3%80%82） for the dot in a hostname — “it is possible to bypass the blocking.” The interstitial this section is about did not appear.

2. [#1079561 archived 2026-09-24 ↗](#) Information Disclosure · high

Big Picture web browser leaks login cookies and discloses sensitive information (may lead to account takeover)

Reported 15 January 2021, resolved, disclosed 21 September 2021. Valve's own severity rating is high, and the title states the consequence. The disclosed record carries the title, rating and dates; the technical write-up is not in the published payload, so nothing further is quoted from it here.

The authors' reading, not a finding of either report: the two sit on either side of the same gate. One is the warning failing to appear before the destination opens; the other is the session leaving the client once a page is open inside it. The sequence that follows a stolen session — key registered without a prompt, the real trade cancelled, a duplicate sent to a clone — is the pattern set out in [Section 5](#) to [Section 8](#). Both reports are closed; neither is offered as a live vulnerability. What they establish is narrower and enough: the single remaining layer has a documented history of being got around, on Valve's own record.

Source: the HackerOne disclosure record for each report, retrieved 2026-09-22 — number, title, team, weakness, severity, report and disclosure dates and state as published by HackerOne. Retrieval record: [hackerone-reports.json](#). No bounty figure is stated here; none appears in the disclosed record retrieved.

Field observation · PhishDestroy **Detections that normally end a campaign did not end these**

We watched phishing hosts keep pulling traffic while carrying **20 or more vendor detections**. Delivered by web or email, a campaign in that state dies in days — the browser and the antivirus both act on those verdicts. Inside Steam neither does. The verdicts existed the whole time. Nothing in the client ever asked.

What this means for a minor with an antivirus installed

He ignores no warning and disables nothing. A message arrives inside Steam — a free knife, a vote for a friend's team — and the page opens inside Steam. He never opens a browser, so the browser never protects him. His antivirus is running and may already rate that domain malicious; it is never asked. **Every warning that could have stopped him already exists. None of it reaches him.** He believes he never left Steam, and he did not.

The giveaway is not the phisher's invention — free Valve items are how the skin and gambling sites advertise, which is exactly why the message reads as ordinary. Complaints: [the enforcement archive](#). Age and children's data: [Section 17](#).

Sources: Steam renders web content through the Chromium Embedded Framework in the `steamwebhelper` process — see [DARKNAVY's technical analysis of 27 June 2024 ↗](#) and Valve's own [Steam Client Beta forum ↗ archived 2025-08-04 ↗](#). No CEF version is relied on: embedded Chromium is not Chrome and ships none of Chrome's Google services. Safe Browsing without API keys: [Chromium design document ↗ archived 2026-08-21 ↗](#), [Chromium API keys ↗ archived 2026-08-30 ↗](#) ("Many of the Google APIs used by Chrome are specific to Google and not intended for use in derived products"), and the same effect in other unkeyed builds — [Homebrew cask #193814 ↗](#). Test URLs: [testsafebrowsing.appspot.com ↗](#).

[Download PDF](#)

[Download JSON](#)

[Download PGP signature](#)

PhishDestroy · file integrity signature for this fact

[Material record ↗](#)