SheetSpeed security

Updated 19 November 2026 · Covers the Chrome extension (Core and 0.4.6) and SheetSpeed for Excel on Mac.

SheetSpeed runs in Chrome. Keyboard shortcuts act through Google's document controls. Connected tools in 0.4.6 call Google APIs directly, only after optional host access is granted and the user consents. Workbook contents are not routed through a SheetSpeed-operated server or an AI provider in the inspected Store package.

The trust pack for IT and procurement review is at /trust/.

SheetSpeed for Excel on Mac

- **Apple-verified:** signed with a Developer ID Application certificate and notarized by Apple. Gatekeeper verifies the signature and notarization ticket on first launch.
- **One permission:** macOS Accessibility (System Settings → Privacy & Security). It is required to observe the Option-key sequences and drive Excel / PowerPoint through accessibility APIs. The app only acts while Excel or PowerPoint is the front app; keystrokes are never logged.
- **No network:** the app holds no network entitlement — it makes no network connections, ships no analytics or telemetry, and cannot transmit data off the machine (verifiable with `codesign -d --entitlements`).
- **Untrusted user keymaps:** bundled keymaps ship inside the signed app. Keymaps placed in `~/Library/Application Support/SheetSpeed/Keymaps/` are treated as untrusted: `script` (AppleScript) steps are dropped at load, so a keymap file can never execute code.
- **Updates:** newer builds are published on this site; every release is signed and notarized the same way.

Implemented controls

- **Document-page scope:** content scripts match Google document URLs rather than all websites. Core shortcuts do not call a Google API. Version 0.4.6 grants `storage`, `activeTab` and `identity` at install. There is no `host_permissions` list and no local-companion permission. Google Sheets, Slides and Drive API hosts are optional and are not granted until a Pro tool needs them. Nothing connects to Google until you approve the consent screen.
- **Request checks:** connected commands validate the extension sender and originating document before targeting a workbook or presentation. These are application checks, not a file-level restriction in Google's account-wide Sheets or Slides scopes.
- **Packaged code:** Manifest V3 uses a content security policy of `script-src 'self'; object-src 'none'` for extension pages. Version 0.4.6 does not include a local companion app.
- **Recovery:** connected mutations can keep local undo records and modeling recovery journals. Concurrent edits or an interrupted operation can require reconciliation; those records are not equivalent to a full workbook backup.
- **Transport:** connected Google requests use HTTPS. There is no additional SheetSpeed encryption layer over Chrome local storage.

Review these boundaries

Core's lack of a Google API token does not mean it lacks document access: its content scripts can read page context and operate document controls. Connected Google permissions are broader than the file selected for a command. Optional Google API hosts, once granted, are host-level permissions even when endpoint paths are listed in implementation documentation.

Refresh from Sheets in 0.4.6 is off by default. If a user opts in, it temporarily link-shares a PNG in Google Drive and attempts to remove it afterwards. Browser termination or denied cleanup can leave it behind. Organizations that prohibit that sharing should set the `slidesLinkShare` policy to `false`. See the privacy policy, the Workspace admin guide and deployment review.

Report a vulnerability privately

Email jarvis@agi-jarvis.com with “SheetSpeed security report” in the subject. Include the extension version, affected feature, reproduction steps and expected versus observed behavior using a synthetic document. Do not send live access tokens, passwords or confidential document contents.

Security reports go to the product operator. There is no published bug bounty or guaranteed response time. Personal-data breaches affecting customer data are notified within 72 hours of becoming aware, as set out in the DPA. Any organization-specific requirements must be agreed separately. Do not test against other users' data or systems without their authorization.

Stop access and remove local data

1. Disable SheetSpeed in Chrome, then reload or close its open Google document tabs to remove already-injected page scripts.
2. For a connected build, revoke SheetSpeed at Google account permissions. Settings → Disconnect can clear cached tokens before you disable the extension.
3. For routine removal, resolve pending recovery and clear undo history before disabling. If you disabled first to stop access immediately, do not re-enable it just to clear history. Uninstalling removes local extension storage, including recovery records, but cannot undo document edits or erase exported images.
4. Inspect Drive for any interrupted temporary image, and clear separately copied files and clipboard contents according to your organization's policy.

Assurance status

No SOC 2 report, ISO 27001 certificate or independent penetration-test report is represented as available. This page describes implementation controls; it is not a certification, compliance guarantee, or statement that an employer has approved SheetSpeed.

© 2026 SheetSpeed