Chrome 131 Enterprise and Education release notes

For administrators who manage Chrome browser or Chrome devices for a business or school.

These release notes were published on November 6, 2024..

See the latest version of these release notes online at https://g.co/help/ChromeEnterpriseReleaseNotes

Chrome 131 release summary

Current Chrome browser changes	Security / Privacy	User productivity / Apps	Management
Search with Google Lens on iOS		✓	
Asynchronous real-time Safe Browsing check	✓		
Ad-hoc code signatures for PWA shims on macOS		✓	
Choose from Google Drive on iOS		✓	
Chrome PDF Viewer OCR		✓	
Chrome on iOS promo on Desktop NTP		✓	
Cross profile password-reuse detection	✓		
Deprecate Safe Browsing Extended reporting	✓		
Entrust certificate distrust	✓		
Insecure form warnings on iOS	✓		
PartitionAlloc with Advanced Checks (PA/AC)	✓		
Simplified sign-in and sync experience	✓		
Tab freezing on Energy saver		✓	
Update Google Play Services to fix issues with on-device passwords		✓	
X25519Kyber768 key encapsulation for TLS	✓		

Deprecation of CSS Anchor Positioning property inset-area	✓		
Improvements to styling structure of <details> and <summary> elements		✓	
Keyboard Lock and Pointer Lock permissions	✓		
Remove non-standard GPUAdapter requestAdapterInfo() method	✓		
<select> parser relaxation	✓		
Support external SVG resources for clip-path, fill, stroke, and marker-* properties	✓		
Support non-special scheme URLs	✓		
Translate for search with Google Lens		✓	
New policies in Chrome browser			✓
Removed policies in Chrome browser			✓
Current Chrome Enterprise Core changes	**Security/Privacy**	**User productivity/Apps**	**Management**
GenAI Defaults policy			✓
Chrome extension telemetry integration with SecOps	✓		
Customized Chrome Web Store for Enterprises			✓
DownloadRestrictions policy support on Android	✓		✓
Enterprise Policy to force adaptive buffering for WebAudio Rendering			✓
Generating insights for Chrome DevTools Console warnings and errors			✓

Recommended policies in the Admin console			✓
Current Chrome Enterprise Premium changes	**Security/Privacy**	**User productivity/Apps**	**Management**
Chrome Enterprise Data Controls: Clipboard	✓		
Screenshot protections	✓		
Upcoming Chrome browser changes	**Security / Privacy**	**User productivity / Apps**	**Management**
Read aloud in Reading mode in Chrome 132		✓	
Removal of old Headless from the Chrome binary		✓	
Capture all screens	✓		
Remove prefixed HTMLVideoElement fullscreen APIs		✓	
Remove ThirdPartyBlockingEnabled policy			✓
Keyboard-focusable scroll containers		✓	
Throw exception for popovers or dialogs in non-active documents		✓	
User Link capturing on PWAs		✓	✓
Network Service on Windows will be sandboxed	✓		
Remove SwiftShader fallback	✓		
Privacy & security panel in Chrome DevTools	✓		
Chrome Sync to end support for Chrome versions more than four years old		✓	

Disallow spaces in non-file:// URL hosts	✓		
SafeBrowsing API v4 to v5 migration	✓		
Blob URL partitioning: Fetching or Navigation	✓		
Deprecate mutation events		✓	
UI Automation accessibility framework provider on Windows		✓	
Upcoming Chrome Enterprise Core changes	**Security / Privacy**	**User productivity / Apps**	**Management**
Remove enterprise policy used for legacy same site behavior			✓
Upcoming Chrome Enterprise Premium changes	**Security/Privacy**	**User productivity/Apps**	**Management**
DLP file download access prevention	✓		

The enterprise release notes are available in 9 languages. You can read about Chrome's updates in English, German, French, Dutch, Spanish, Portuguese, Korean, Indonesian, and Japanese. Allow 1 to 2 weeks for translation for some languages.

Chrome Enterprise and Education release notes are published in line with the Chrome release schedule, on the Early Stable date for Chrome browser.

Current Chrome version release notes

Current Chrome browser changes

Search with Google Lens on iOS

Since Chrome 126, users can search any images or text they see on their screen with Google Lens. To use this feature, go to a website and click **Search with Google Lens** on the on-focus omnibox chip and on the right-click menus on desktop, or on the 3-dot menu on both desktop and mobile. Users can click, highlight, or drag anywhere on the screen to search its contents, and refine their search by adding keywords or questions to the search box. Admins can control the feature through a policy called LensOverlaySettings. To perform the search, a screenshot of the screen is sent to Google servers but it is not linked to any IDs or accounts, it is not viewed by any human, and data about its contents is not logged. We are starting the rollout of this feature gradually on iOS in Chrome 131 and we plan to launch fully in Chrome 132.

- Chrome 126 on ChromeOS, Linux, mac, Windows: Rollout of the feature at 1% Stable
- Chrome 127 on ChromeOS, Linux, mac, Windows: Rollout to 100% Stable
- **Chrome 131 on iOS:** Rollout of the feature starts
- Chrome 132 on iOS: Rollout to 100% Stable

Asynchronous real-time Safe Browsing check on iOS

Today Safe Browsing checks are on the blocking path of page loads, meaning that the user cannot see the page until the checks are completed. To improve Chrome's loading speed, real-time Safe Browsing checks will no longer block page loads after Chrome 122, and after Chrome 131 on iOS. We have evaluated the risk and put mitigations in place:

1. For malware and 0-day attacks, local-blocklist checks will still be conducted in synchronous manner so that malicious payloads are still blocked by Safe Browsing.
2. For phishing attacks, we've looked at data and it is unlikely the user would have interacted with the page (for example, typed a password) by the time we show the warning.

- Chrome 122 on Android, ChromeOS, Linux, macOS, Windows
- **Chrome 131 on iOS**

Ad-hoc code signatures for PWA shims on macOS

Code signatures for the application shims that are created when installing a Progressive Web App (PWA) on macOS are changing to use ad-hoc code signatures that are created when the application is installed. The code signature is used by macOS as part of the application's identity. These ad-hoc signatures result in each PWA app shim having a unique identity to macOS; currently every PWA looks like the same application to macOS.

This addresses problems when attempting to include multiple PWAs in the macOS **Open at Login** preference pane, and permits future improvements to handling of user notifications within PWAs on macOS.

Administrators should test for compatibility with any endpoint security or binary authorization tools they use (such as Santa). The feature can be switched on for testing using the `chrome://flags/#use-adhoc-signing-for-web-app-shims flag`. Admins can then install a PWA and ensure that it launches as expected.

If there is an incompatibility between the feature and their current security policies, the enterprise policy can be used to disable the feature while they deploy an updated endpoint security policy. The enterprise policy is intended to be used to disable the feature only until endpoint security policies have been updated, at which point it should be unset.

- Chrome 129 on macOS
  This feature is turned on with a flag (`chrome://flags/#use-adhoc-signing-for-web-app-shims`) so that enterprises can test for compatibility with their endpoint security tools, such as Santa. If it is not currently compatible, they can control the feature using the enterprise policy while they update their endpoint security configurations. The enterprise policy is intended to be used to disable the feature only until endpoint security policies have been updated.
- **Chrome 131 on macOS**
  Feature begins to roll out to stable, starting at 1% rollout.

Choose from Google Drive

From Chrome 131 onwards, Chrome on iOS users can upload a file from Google Drive directly to a web page, without the need to download it on the device first.

- **Chrome 131 on iOS**
  Includes core functionality for uploading a single file.

Chrome PDF Viewer OCR

Chrome Desktop now makes scanned PDFs more accessible. Using on-device Optical Character Recognition (OCR) to maintain privacy (no content is sent to Google), Chrome automatically converts scanned PDFs, allowing you to select text, Ctrl+F, copy, and paste. The feature does not bypass secure PDFs. It only uses OCR on PDFs the user has access to. The solution unlocks PDF accessibility to Chrome users without any extra steps, making PDFs as accessible as the rest of the web.

- **Chrome 131 on ChromeOS, Linux, macOS, Windows**

Chrome on iOS promo on Desktop NTP

A Chrome on iOS promo on the Desktop new tab page. This promo aims to increase awareness of Chrome on iOS and present a simple way to install.
You can control this feature using the existing policies PromotionalTabsEnabled and NTPMiddleSlotAnnouncementVisible.

- **Chrome 131 on Linux, macOS, Windows**

Cross profile password-reuse detection

Previously, password-reuse detection of corporate credentials was only detectable in the corporate profile. Now, password-reuse detection detects corporate credential reuse across all non-Incognito profiles on the managed browser.
We've updated the cross profile password-reuse detection criteria to more accurately reflect managed enterprise accounts. We've also updated the on-screen message to make it clearer to users that their organization is monitoring their corporate password reuse.

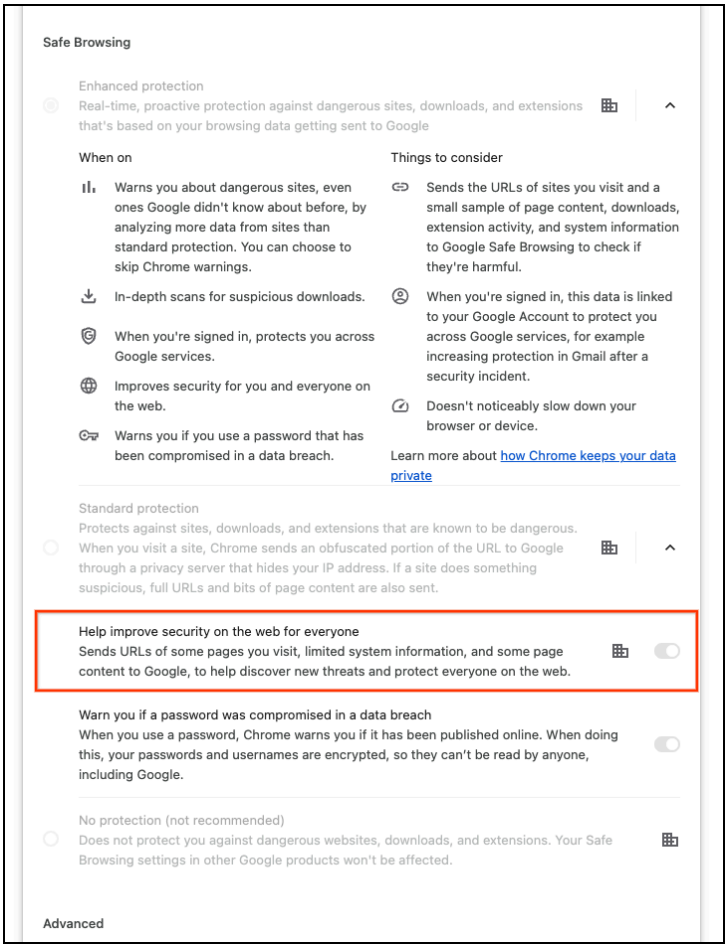
- Chrome 123 on Android,iOS,ChromeOS,Linux,macOS,Windows,Fuchsia
- **Chrome 131 on Android,iOS,ChromeOS,Linux,macOS,Windows,Fuchsia**

  We've updated the cross profile password-reuse detection criteria to more accurately reflect managed enterprise accounts, and updated the UX message to make it clearer to users that their organization is monitoring their corporate password reuse.

Deprecate Safe Browsing Extended reporting

We are deprecating the Safe Browsing Extended reporting feature, which previously enhanced the security of all users by collecting telemetry information from participating users that is used for Google Safe Browsing protections. The data collected includes URLs of visited web pages, limited system information, and some page content.
This feature is now superseded by **Enhanced protection mode**. We suggest users switch to Enhanced protection to continue providing security for all users in addition to enabling the strongest security available in Chrome. For more information, see Safe Browsing protection levels.

- Chrome 129 on Android, iOS, ChromeOS, Linux, macOS, Windows
  Deprecation of Safe Browsing Extended Reporting. Excluding real-time Client Safe Browsing Report Request
- **Chrome 131 on Android, iOS, ChromeOS, Linux, macOS, Windows**
  Deprecating SafeBrowsingExtendedReportingEnabled for real-time Client Safe Browsing Report Request



Entrust certificate distrust

In response to sustained compliance failures, Chrome is changing how publicly-trusted TLS server authentication (website) certificates issued by Entrust will be trusted by default in Chrome 131 and greater on Windows, macOS, ChromeOS, Android, and Linux. iOS policies do not allow use of the Chrome Root Store in Chrome for iOS.

Specifically, TLS certificates validating to the Entrust root CA certificates included in the Chrome Root Store and issued:

- after November 11, 2024, will no longer be trusted by default.
- on or before November 11, 2024, will be unaffected by this change.

Should a Chrome user or enterprise explicitly trust any of the affected Entrust certificates on a platform and version of Chrome relying on the Chrome Root Store, for example, explicit trust is

conveyed through a Windows Group Policy Object, the SCT-based constraints described above will be overridden and certificates will function as they do today.
Additional information and testing resources are the Google Security blog.
To learn more, see this FAQ about the Chrome Root Store.

- **Chrome 131 on Android, ChromeOS, Linux, macOS, Windows**
  All versions of Chrome 131 and higher that rely on the Chrome Root Store will honor the blocking action, but the blocking action will only begin for certificates issued after November 11, 2024.

Insecure form warnings on iOS

Since Chrome 125, Chrome browser blocks form submissions from secure pages to insecure pages on iOS. When Chrome detects an insecure form submission, it displays a warning asking the user to confirm the submission. The goal is to prevent leaking form data over plain text without the user's explicit approval. A policy InsecureFormsWarningsEnabled is available to control this feature.

- Chrome 125 on iOS: Feature rolls out
- **Chrome 131 on iOS**: InsecureFormsWarningsEnabled policy will be removed

PartitionAlloc with Advanced Checks (PA/AC)

PartitionAlloc (PA) and its associated memory security projects have an array of advanced safeguards that are deactivated by default (or exclusively in debug builds) due to their potential impact on performance. While enabling the feature for all users might not be immediately possible, there is still an opportunity to partially enable it under specific, limited conditions.
This project seeks to achieve advanced safeguards for the enterprise customers. Enterprise administrators have the option to apply enhanced security measures through Enterprise Policies. Security tends to be prioritized over performance in Enterprise. There's a likelihood that they desire advanced checks, even if it comes at a cost to performance.
PA with Advanced Checks is advanced memory security. The feature is OFF by default due to expected performance regression. Enterprise customers have an option to enable it to achieve advanced security via enterprise policy.

- **Chrome 131 on Android, iOS, ChromeOS, Linux, macOS, Windows, Fuchsia**

Simplified sign-in and sync experience

Starting in Chrome 131, existing users with Chrome sync turned on now experience a simplified and consolidated version of sign-in and sync in Chrome. Chrome sync is no longer shown as a separate feature in settings or elsewhere. Instead, users can sign in to Chrome to use and save information like passwords, bookmarks and more in their Google Account, subject to the relevant enterprise policies.
As before, the functionality previously part of Chrome sync that saves and accesses Chrome data in the Google Account can be controlled by SyncTypesListDisabled. Sign-in to Chrome can be switched off via BrowserSignin as before.
Note that the changes do not affect users' ability to sign in to Google services on the web (like Gmail) without signing in to Chrome, their ability to stay signed out of Chrome, or their ability to control what information is synced with their Google Account.

- **Chrome 131 on Android**

Tab freezing on Energy saver

When Energy saver is active, Chrome freezes a tab that has been hidden and silent for >5 minutes and uses a lot of CPU, unless:

- The tab provides audio- or video- conferencing functionality, detected via microphone, camera or screen, window, or tab capture, or an RTCPeerConnection with an open RTCDataChannel or a live MediaStreamTrack.
- The tab controls an external device, detected via usage of Web USB, Web Bluetooth, Web HID or Web Serial.

This extends battery life and speeds up Chrome through reduced CPU usage.

- Chrome 130 on ChromeOS, Linux, macOS, Windows
  The feature can be tested in Chrome 130 using the `#freezing-on-energy-saver` entry in about:flags. Alternatively, it can be tested with the `#freezing-on-energy-saver-testing` flag, which simulates that Energy saver is

active and that all tabs use a lot of CPU; this allows verifying whether a tab is eligible for freezing and would be frozen if it used a lot of CPU. Energy saver availability can be controlled using the BatterySaverModeAvailability policy. This change has no effect when Energy save is inactive.

- **Chrome 131 on ChromeOS, Linux, macOS, Windows**
  The feature will start rolling out to 1% of stable in Chrome 131. It will gradually be ramped up to 100% of Stable. Energy saver availability can be controlled via the BatterySaverModeAvailability policy. This change has no effect when Energy saver is inactive.

Update Google Play Services to fix issues with on-device passwords

Users with old versions of Google Play Services will experience reduced functionality with their on-device passwords, and Password Manager might soon stop working for them altogether. These users will need to update Google Play Services, or will be guided through other troubleshooting methods depending on their state. This is part of an ongoing migration that only affects Android users of Google Password Manager.

- **Chrome 131 on Android**

X25519Kyber768 key encapsulation for TLS

Starting in Chrome 124, Chrome enables by default on all desktop platforms a new post-quantum secure TLS key encapsulation mechanism X25519Kyber768, based on a NIST standard (ML-KEM). This protects network traffic from Chrome with servers that also support ML-KEM from decryption by a future quantum computer. This change should be transparent to server operators. This cipher will be used for both TLS 1.3 and QUIC connections.
However, some TLS middleboxes might be unprepared for the size of a Kyber (ML-KEM) key encapsulation, or a new TLS ClientHello cipher code point, leading to dropped or hanging connections. This can be resolved by updating your middlebox, or disabling the key encapsulation mechanism via the temporary PostQuantumKeyAgreementEnabled enterprise policy, which will be available through the end of 2024. However, long term, post-quantum secure ciphers will be required in TLS and the enterprise policy will be removed. Post-quantum cryptography is required for CSNA 2.0.

For more detail, see this Chromium blog post and this Google Security blog post.

- Chrome 124 on Windows, macOS, Linux: new post-quantum secure TLS key encapsulation mechanism X25519Kyber768 is enabled
- **Chrome 131 on Linux, macOS, Windows**: Chrome will switch the key encapsulation mechanism to the final standard version of ML-KEM
- Chrome 141 on Windows, macOS, Linux: Remove enterprise policy

Deprecation of CSS Anchor Positioning property *inset-area*

The CSS working group (CSSWG) resolved to rename the `inset-area` property to `position-area`. For more details, see the CSSWG discussion on github. The new property name, `position-area`, as a synonym for `inset-area` shipped via this feature update described on Chrome Platform Status, describing the deprecation and removal of the `inset-area` property.

- **Chrome 131 on Windows, macOS, Linux, Android**

Improvements to styling structure of <details> and <summary> elements

Support more CSS styling for the structure of `<details>` and `<summary>` elements to allow these elements to be used in more cases where disclosure widgets or accordion widgets are built on the web. In particular, this change removes restrictions that prevented setting the display property on these elements, and adds a `::details-content` pseudo-element to style the container for the part that expands and collapses.

- **Chrome 131 on Windows, macOS, Linux, Android**

Keyboard Lock and Pointer Lock permissions

May show a permission prompt to the user when Keyboard Lock or Pointer Lock is requested by a website, and saves the user preferences as content settings. The settings can be queried for via the Permissions API. This helps mitigate the abusive use of the APIs.

- **Chrome 131 on Windows, macOS, Linux**

Remove non-standard GPUAdapter requestAdapterInfo() method

The WebGPU WG decided it was impractical for `requestAdapterInfo()` to trigger a permission prompt so they've removed that option and replaced it with the GPUAdapter info attribute so that web developers can get the same GPUAdapterInfo value synchronously this time. To read more, see the previous Intent to Ship: WebGPU: GPUAdapter info attribute .

- **Chrome 131 on Windows, macOS, Linux, Android**

<select> parser relaxation

This change makes the HTML parser allow additional tags in `<select>` besides `<option>`, `<optgroup>`, and `<hr>`.
This change is in support of the customizable `<select>` feature but is being shipped first because it can be done separately and has some compact risks.
This feature is gated by the temporary policy, **SelectParserRelaxationEnabled**. This is a temporary transition period, and the policy will stop working in milestone Chrome 136.
For more details, see the Open UI Customizable <select> explainer and the What Working Group HTML parser changes for customizable <select> article.

- **Chrome 131 on Windows, macOS, Linux, Android**

Support external SVG resources for clip-path, fill, stroke and marker-* properties

Allow external references for clip paths, markers, and paint servers (for the `fill` and `stroke` properties). For example, `clip-path: url("resources.svg#myPath"`.

- **Chrome 131 on Windows, macOS, Linux, Android**

Support non-special scheme URLs

Chrome 130 supports non-special scheme URLs, for example, git://example.com/path. Previously, the Chromium URL parser didn't support non-special URLs. The parser parses non-special URLs as if they had an opaque path, which is not aligned with the URL standard. Now, the Chromium URL parser parses non-special URLs correctly, following the URL standard. For more details, see http://bit.ly/url-non-special.

- Chrome 130 on Windows, macOS, Linux, Android
- **Chrome 131 on Windows, macOS, Linux, Android**
- Chrome 134 on Windows, macOS, Linux, Android: Feature flag being removed

Translate for Search with Google Lens

Augmented reality (AR) Translation capabilities are being implemented to the **Search with Google Lens** feature. An enterprise policy is already in place enabling enterprises to turn the feature on or off using LensOverlaySettings.

- **Chrome 131 on ChromeOS, Linux, macOS, Windows**

New policies in Chrome browser

Policy	Description
DownloadRestrictions	Allow download restrictions
CAPlatformIntegrationEnabled	Use user-added TLS certificates from platform trust stores for server authentication
SelectParserRelaxationEnabled	Controls whether the new HTML parser behavior for the <select> element is enabled
EnterpriseProfileBadgeToolbarSettings	Controls visibility of enterprise profile badge in the toolbar
WebAudioOutputBufferingEnabled	Enable adaptive buffering for Web Audio

Removed policies in Chrome browser

Policy	Description
ProfileLabel	This policy controls a label used to identify a signed in profile. This label will be shown in various locations to help users identify the profile such as next to the toolbar profile icon.
ToolbarAvatarLabelSettings	Managed toolbar avatar label setting
BeforeunloadEventCancelByPreventDefaultEnabled	Control new behavior for the cancel dialog produced by the beforeunload event

Current Chrome Enterprise Core changes

GenAI Defaults policy

Starting in 131, Chrome Enterprise Core introduces a policy, GenAiDefaultSettings, to control the default behavior of multiple GenAI policies as part of our Trusted Tester program. You can sign up for our Trusted Tester program here. This policy does not impact any manually-set policy values for generative AI features. This policy controls the default settings for the following policies:

- CreateThemesSettings
- DevToolsGenAiSettings
- HelpMeWriteSettings
- HistorySearchSettings
- TabOrganizerSettings
- TabCompareSettings
- GenAIVcBackgroundSettings
- GenAIWallpaperSettings
- HelpMeReadSettings

For more details about the default settings, see Chrome—Generative AI features and policies.

- **Only available to Trusted Testers.** You can sign up for our Trusted Tester program here.

Chrome extension telemetry integration with SecOps

We begin to collect relevant Chronicle extension telemetry data from within Chrome, for managed profiles and devices, and send it to Google SecOps. Google SecOps analyzes the data to provide instant analysis and context on risky activity; this data is further enriched to provide additional context and is searchable for a year.

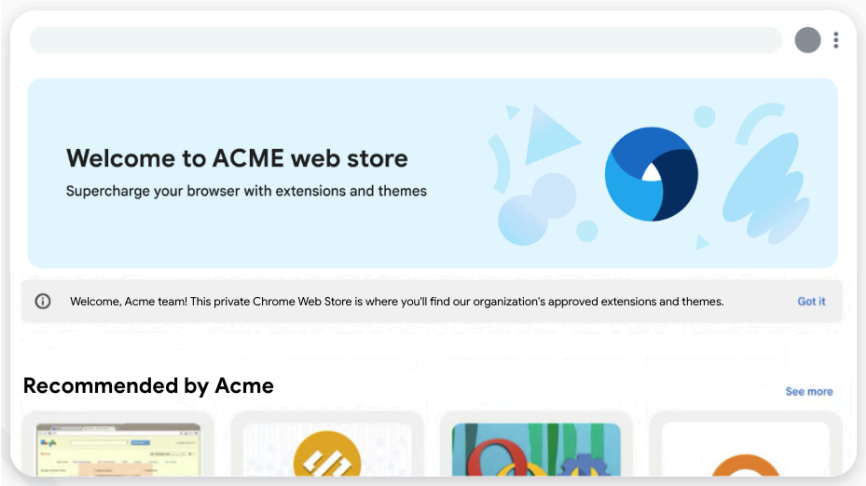
- **Chrome 131 on ChromeOS, Linux, macOS, Windows**

Customized Chrome Web Store for Enterprises

IT admins will be able to customize the Chrome Web Store for their managed end-users using company-specific branding, custom messaging and tailored navigation. Admins can personalize the store with logos, banners, and recommended extensions, while also hiding irrelevant categories and improving extension discovery.
This feature is configurable via the Admin console and this milestone 1 custom configurations will be available to all signed-in managed users (users signed-in to the Chrome Web Store with a managed Google Account). Milestone #2 will support this feature for CEC enrolled browsers (without the need to sign in) and will only be available later in 2025.

As early as Chrome 131, on Linux, mac, Windows and ChromeOS : Milestone #1 is available via the Trusted Tester program
As early as Chrome 132 on Linux, mac, Windows and ChromeOS: Milestone #1 rolls out

- **Chrome 131 on ChromeOS, macOS, Windows**



DownloadRestrictions policy support on Android

DownloadRestrictions is a universal policy available to Chrome Enterprise Core users on Desktop. DownloadRestrictions policy is now supported on Android. This policy allows admins to block all downloads on mobile Chrome on Android.

- **Chrome 131 on Android**

Enterprise policy to force adaptive buffering for WebAudio rendering

Chromium's WebAudio implementation includes an adaptive buffering mechanism, which was added to resolve numerous glitching issues especially on Android with the AAudio backend. While this mechanism reduced glitches significantly, it also increased audio latency. Chrome is running an experiment that will disable the adaptive buffering mechanism and run the rendering synchronously on all platforms besides Android.
Starting Chrome 131, an enterprise policy WebAudioOutputBufferingEnabled is available that will force Chrome to default to the previous behavior of using adaptive buffering for WebAudio rendering.

- **Chrome 131 on ChromeOS, Linux, macOS, Windows**

Generating insights for Chrome DevTools Console warnings and errors

A new Generative AI (GenAI) feature is now available for unmanaged users: Generating insights for Chrome DevTools Console warnings and errors.

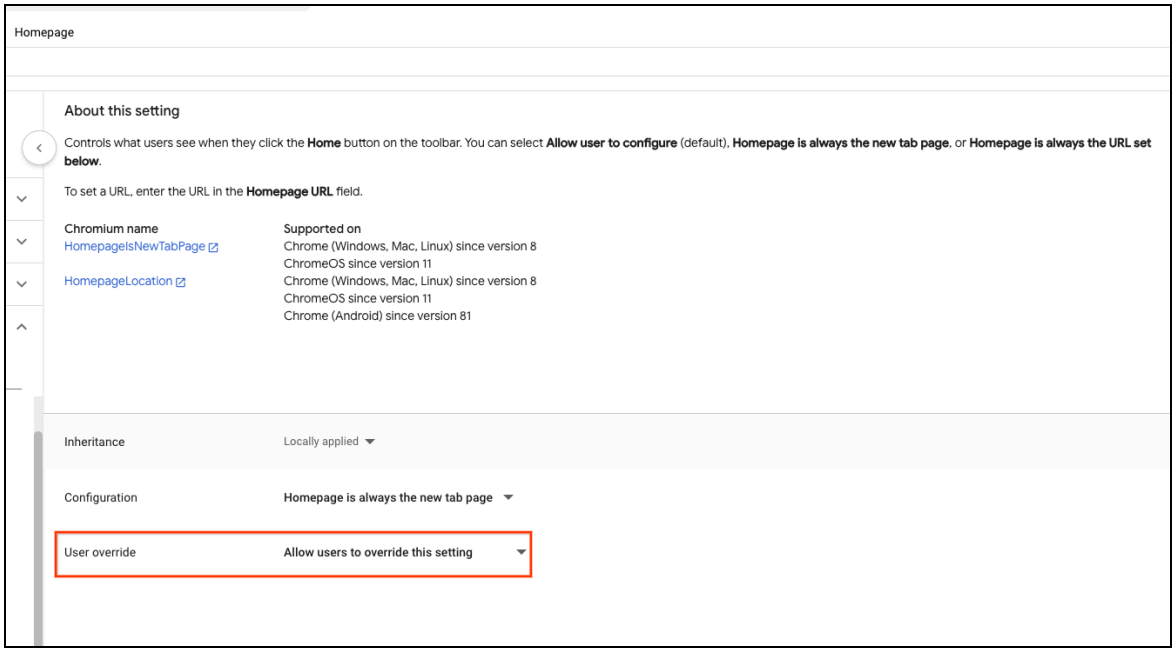
These insights provide a personalized description and suggested fixes for the selected errors and warnings. Initially, this feature is only available to users (18+) in English. Admins can control this feature by using the DevToolsGenAiSettings policy.

- Chrome 125 on ChromeOS, Linux, macOS, Windows
  Feature becomes available to unmanaged users globally, except Europe, Russia, and China.
- Chrome 127 on ChromeOS, Linux, macOS, Windows
  Feature becomes available to managed Chrome Enterprise & Education users in supported regions.
- **Chrome 131 on ChromeOS, Linux, macOS, Windows**
  In Chrome 131, a new Generative AI (GenAI) feature becomes available for managed users: a dedicated **AI assistance** panel in Chrome DevTools which assists the human operator investigating & fixing styling challenges and helps debugging the CSS.
- Chrome 132 on ChromeOS, Linux, macOS, Windows
  The AI assistance panel can now explain resources in the Performance panel, Sources panel, and Network panel, in addition to the previous support for style debugging

Recommended policies in the Admin console

As early as November 1st, admins will be able to choose whether some settings are recommended or mandatory using the **User override** control. This control will gradually rollout for policies that can be recommended, starting with the following policies:

- Warn before quitting
- System Default Printer
- Battery Saver Mode
- Homepage
- Safe Browsing protection
- Download Restrictions



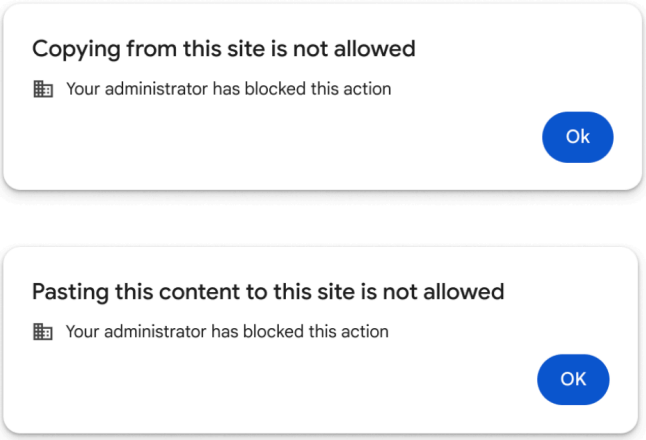
- **Chrome 131 on Android, iOS, ChromeOS, Linux, macOS, Windows**

Current Chrome Enterprise Premium changes

Chrome Enterprise Data Controls: Clipboard

Admins can set data control rules in the Google Admin console to protect end users from data leakage on Chrome browser. Data Controls are lightweight rules set in the Google Admin console that allow admins to set a Chrome policy to control sensitive user actions, such as, copying and pasting sensitive data and taking screenshots or screen sharing.
This feature can be controlled using the DataControlsRules policy.
This feature is available to test for the members of the Chrome Enterprise Trusted Tester program. You can sign up for our Trusted Tester program here.

- Chrome 128 on ChromeOS, Linux, macOS, Windows: Trusted Tester program
- **Chrome 131 on ChromeOS, Linux, macOS, Windows**: Feature rolls out



Screenshot protections

Admins can prevent users from taking screenshots or screen sharing specific web pages considered to contain sensitive data. Admins create a DLP URL filtering rule to block users taking screenshots or screen sharing specific URLs or categories of URLs. This feature can be controlled using the same EnterpriseRealTimeUrlCheckMode policy that enables all real-time URL lookups.
This feature is available to test for the members of the Chrome Enterprise Trusted Tester program. You can sign up for our Trusted Tester program here.

- Chrome 129 on ChromeOS, Linux, macOS, Windows: Trusted Tester program
- **Chrome 131 on ChromeOS, Linux, macOS, Windows:** Feature rolls out.

Coming soon

Note: The items listed below are experimental or planned updates. They might change, be delayed, or canceled before launching to the Stable channel.

Upcoming Chrome browser updates

Read aloud in Reading mode in Chrome 132

Reading mode is a side-panel feature that provides a simplified view of text-dense web pages. Reading mode will include a Read aloud feature which allows users to hear the text they are reading spoken out loud. Users can choose different natural voices and speeds, and see visual highlights.

- **Chrome 132 on ChromeOS, Linux, macOS, Windows**

Removal of old Headless from the Chrome binary

Running Chrome with `--headless=old` no longer launches the old Headless mode, and instead prints the following log message:
The old Headless mode has been removed from the Chrome binary. You can use the new Headless mode or the chrome-headless-shell, which is a standalone implementation of the old Headless mode.

- **Chrome 132 on Linux, macOS, Windows**

Capture all screens

This feature captures all the screens currently connected to the device using `getAllScreensMedia()`. Calling `getDisplayMedia()` multiple times requires multiple user gestures, burdens the user with choosing the next screen each time, and does not guarantee to the app that all the screens were selected. `getAllScreensMedia()` improves on all of these fronts.

This feature is only exposed behind the MultiScreenCaptureAllowedForUrls enterprise policy, and users are warned before recording even starts, that recording *could* start at some point. The API will only work for origins that are specified in the MultiScreenCaptureAllowedForUrls allowlist. Any origin not specified there, will not have access to it.

- **Chrome 132 on Windows, macOS, Linux**

Remove prefixed HTMLVideoElement fullscreen APIs

The prefixed `HTMLVideoElement`-specific fullscreen APIs have been deprecated since approximately M38. They were replaced by the `Element.requestFullscreen()` API, which first shipped un-prefixed in M71, in 2018. As of 2024, most browsers have had support for the un-prefixed APIs for a few years now.
This feature tracks removing the following APIs from `HTMLVideoElement`:

- readonly attribute boolean webkitSupportsFullscreen;
- readonly attribute boolean webkitDisplayingFullscreen;
- void webkitEnterFullscreen();
- void webkitExitFullscreen();

// Note the different capitalization of the "S" in FullScreen.

- void webkitEnterFullScreen();
- void webkitExitFullScreen();

These methods are now only aliases for the modern API. Their use has declined steadily over the years.

- **Chrome 132 on Windows, macOS, Linux, Android**

Remove ThirdPartyBlockingEnabled policy

Due to unexpected issues, ThirdPartyBlockingEnabled will be removed in Chrome 135. If you have feedback about this removal, please file a bug here.

- **Chrome 132 on Windows:** Deprecation of ThirdPartyBlockingEnabled policy
- Chrome 135 on Windows: Removal of ThirdPartyBlockingEnabled policy

Keyboard-focusable scroll containers

We plan to improve accessibility by making scroll containers focusable using sequential focus navigation. Today, the tab key doesn't focus scrollers unless `tabIndex` is explicitly set to 0 or more. By making scrollers focusable by default, users who can't (or don't want to) use a mouse will be able to focus clipped content using their tab and arrow keys. This behavior is enabled only if the scroller does not contain any keyboard focusable children. This logic is necessary so we don't cause regressions for existing focusable elements that might exist within a scroller like a `<textarea>`.
Note: The previous rollout of this feature (started in Chrome 127) was stopped due to web compatibility issues, which should be fixed in the current implementation shipping in 130.
Note: The previous rollout of this feature (started in 130) was stopped due to an accessibility regression, which should be fixed in the implementation shipping in 132.

- **Chrome 132 on Windows, macOS, Linux, Android**

Throw exception for popovers or dialogs in non-active documents

This is a corner case change that hopefully does not impact developers. A corner case is where multiple unique conditions occur simultaneously. Previously, calling `` `showPopover()` `` or `` `showModal()` `` on a popover or dialog that resides within an inactive document would silently fail, that is, no exception would be thrown. Since the document is inactive, however, no popover or dialog would be shown. As of the https://github.com/whatwg/html/pull/10705 spec pull request (PR), these situations now throw the `InvalidStateError` exception.

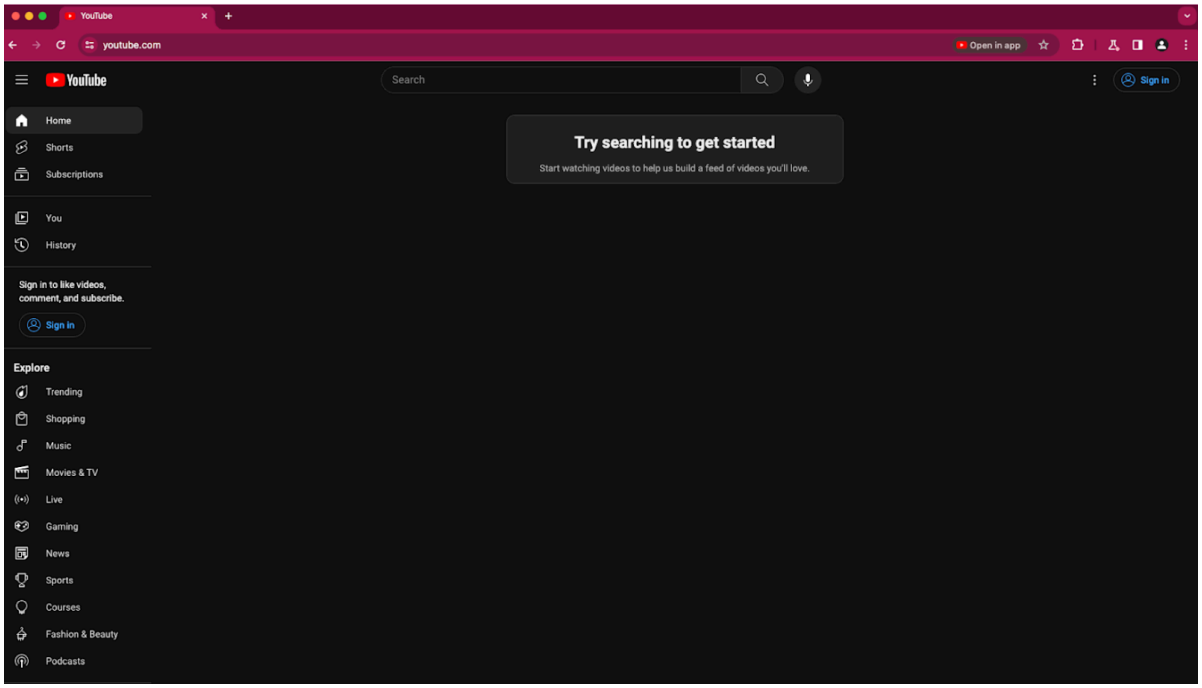
- **Chrome 132 on Windows, macOS, Linux, Android**

User Link capturing on PWAs

Web links automatically direct users to installed web apps. To better align with users' expectations around installed web apps, Chrome makes it easier to move between the browser and installed web apps. When the user clicks a link that could be handled by an installed web app, Chrome adds a chip in the address bar to suggest switching over to the app. When the user clicks the chip, this either launches the app directly, or opens a grid of apps that can support that link. For some users, clicking a link always automatically opens the app.

- Chrome 121 on Linux, macOS, Windows
  When some users click a link, it always opens in an installed PWA, while some users see the link open in a new tab with a chip in the address bar, clicking on which will launch the app. A flag is available to control this feature:
  `chrome://flags/#enable-user-link-capturing-pwa`.

- **Chrome 132 on Linux, macOS, Windows**
  Launch to 100% of Stable with either a default on (always launch apps on link clicks) or a default off (always open in a tab, only launch if the user clicks on chip on address bar).



Network Service on Windows will be sandboxed

To improve security and reliability, the network service, already running in its own process, will be sandboxed on Windows. As part of this, third-party code that is currently able to tamper with the network service may be prevented from doing so. This might cause interoperability issues with software that injects code into Chrome's process space, such as Data Loss Prevention software. The NetworkServiceSandboxEnabled policy allows you to disable the sandbox if incompatibilities are discovered. You can test the sandbox in your environment using these instructions.
You can report any issues you encounter.

- **Chrome 132 on Windows**
  Network Service sandboxed on Windows

Remove SwiftShader fallback

Allowing automatic fallback to WebGL backed by SwiftShader is deprecated and WebGL context creation will fail instead of falling back to SwiftShader. This was done for two primary reasons:

1. SwiftShader is a high security risk due to JIT-ed code running in Chromium's GPU process.
2. Users have a poor experience when falling back from a high-performance GPU-backed WebGL to a CPU-backed implementation. Users have no control over this behavior and it is difficult to describe in bug reports.

SwiftShader is a useful tool for web developers to test their sites on systems that are headless or do not have a supported GPU. This use case will still be supported by opting in but is not intended for running untrusted content.
To opt-in to lower security guarantees and allow SwiftShader for WebGL, run the chrome executable with the `--enable-unsafe-swiftshader` command-line switch.
During the deprecation period, a warning will appear in the JavaScript console when a WebGL context is created and backed with SwiftShader. Passing `--enable-unsafe-swiftshader` will remove this warning message.
Chromium and other browsers do not guarantee WebGL availability. You can test and handle WebGL context creation failure and fall back to other web APIs such as Canvas2D or an appropriate message to the user.

- **Chrome 133 on Windows, macOS, Linux, Android**

Privacy & security panel in Chrome DevTools

Starting in Chrome 133, developers will be able to use the new **Privacy & security** panel in Chrome DevTools to test how their site will behave when third-party cookies are limited. Developers will be able to temporarily limit third-party cookies, observe how their site behaves, and review the status of third-party cookies on their site.

This feature will not make any permanent changes to existing enterprise policies, but it will let third-party cookie related enterprise policies (that is,. BlockThirdPartyCookies and CookiesAllowedForUrls) be temporarily overridden to be more restrictive. If your enterprise policy already blocks third-party cookies using BlockThirdPartyCookies, this feature will be disabled.

The new **Privacy & security** panel will replace the existing **Security** panel. TLS connection and certificate information will continue to be available on the **Security** tab in the **Privacy & security** panel.

- **Chrome 133 on ChromeOS, Linux, macOS, Windows**

Chrome Sync to end support for Chrome versions more than four years old

Starting in February 2025, Chrome Sync (using and saving data in your Google Account) will no longer support Chrome versions that are more than four years old. You need to upgrade to a more recent version of Chrome if you want to continue using Chrome Sync.

- **Chrome 133 on Android, iOS, ChromeOS, Linux, macOS, Windows**
  This change affects only the old versions of Chrome and will be rolled out server-side. Chrome 133 is specified only to reflect the timeline when the change will make an effect.

Disallow spaces in non-file:// URL hosts

Per spec URL hosts [1] cannot contain the space character, but currently URL parsing in Chromium allows spaces in the host.

This causes Chromium to fail several tests included in the Interop2024 'HTTPS URLs for WebSocket' [2] and 'URL' focus areas [3].
To bring Chromium into spec compliance, we would like to remove spaces from URL hosts altogether, but a difficulty with this is that they are used in the host part in Windows `file://` URLs (Github)[4].
This feature will be part of the ongoing work to bring Chromium closer to spec compliance by forbidding spaces for non-file URLs only.

- **Chrome 133 on Android, ChromeOS, Linux, macOS, Windows, Fuchsia**

SafeBrowsing API v4 to v5 migration

Chrome calls into the SafeBrowsing v4 API will be migrated to call into the v5 API instead. The method names are also different between v4 and v5.
If admins have any v4-specific URL allowlisting to allow network requests to https://safebrowsing.googleapis.com/v4*, these should be modified to allow network requests to the whole domain instead: safebrowsing.googleapis.com. Otherwise, rejected network requests to the v5 API will cause security regressions for users.

- **Chrome 133 on Android, iOS, ChromeOS, Linux, macOS, Windows:** This will be a gradual roll-out.

Blob URL partitioning: Fetching or Navigation

As a continuation of Storage partitioning, Chromium will implement partitioning of Blob URL access by Storage Key (top-level site, frame origin, and the has-cross-site-ancestor boolean), with the exception of navigations that will remain partitioned only by frame origin. This behavior is similar to what's currently implemented by both Firefox and Safari, and aligns Blob URL usage with the partitioning scheme used by other storage APIs as part of Storage partitioning. In addition, Chromium will enforce noopener on renderer-initiated navigations to Blob URLs where the corresponding site is cross-site to the top-level site performing the navigation. This aligns Chromium with similar behavior in Safari, and we will pursue spec updates to reflect both of these changes.

This change can be temporarily reverted by setting the **PartitionedBlobURLUsage** policy. The policy will be deprecated when the other storage partitioning-related enterprise policies are deprecated.

- **Chrome 134 on Windows, macOS, Linux**

Deprecate mutation events

Synchronous mutation events, including DOMSubtreeModified, DOMNodeInserted, DOMNodeRemoved, DOMNodeRemovedFromDocument, DOMNodeInsertedIntoDocument, and DOMCharacterDataModified, negatively affect page performance, and also significantly increase the complexity of adding new features to the Web. These APIs were deprecated from the spec in 2011, and were replaced (in 2012) by the much better-behaved Mutation Observer API. Usage of the obsolete mutation events must be removed or migrated to Mutation Observer. Starting in Chrome 124, a temporary enterprise policy, MutationEventsEnabled, will be available to re-enable deprecated or removed mutation events. If you encounter any issues, file a bug here.

Mutation event support will be disabled by default starting in Chrome 127, around July 30, 2024. Code should be migrated before that date to avoid site breakage. If more time is needed, there are a few options:

- The Mutation Events Deprecation Trial can be used to re-enable the feature for a limited time on a given site. This can be used through Chrome 134, ending March 25, 2025.
- A MutationEventsEnabled enterprise policy can also be used for the same purpose, also through Chrome 134.

To read more, see this blog post. Report any issues here.

- **Chrome 135 on Android, Linux, macOS, Windows:** The MutationEventsEnabled enterprise policy will be deprecated.

UI Automation accessibility framework provider on Windows

Starting in Chrome 126, Chrome started directly supporting accessibility client software that uses Microsoft Windows's UI Automation accessibility framework. Prior to this change, such software interoperated with Chrome by way of a compatibility shim in Microsoft Windows. This change is

being made to improve the accessible user experience for many users. It provides complete support for Narrator, Magnifier, and Voice Access; and will improve third-party apps that use Windows's UI Automation accessibility framework. Users of Chrome will find reduced memory usage and processing overhead when used with accessibility tools. It will also ease development of software using assistive technologies.
Administrators might use the UiAutomationProviderEnabled enterprise policy, available from Chrome 125, to either force-enable the new provider (so that all users receive the new functionality), or disable the new provider. This policy will be supported through Chrome 136, and will be removed in Chrome 137. This one-year period is intended to give enterprises sufficient time to work with third-party vendors so that they may fix any incompatibilities resulting from the switch from Microsoft's compatibility shim to Chrome's UI Automation provider.

- Chrome 125 on Windows:The UiAutomationProviderEnabled policy is introduced so that administrators can enable Chrome's UI Automation accessibility framework provider and validate that third-party accessibility tools continue to work.
- Chrome 126 on Windows: The Chrome variations framework will be used to begin enabling Chrome's UI Automation accessibility framework provider for users. It will be progressively enabled to the full stable population, with pauses as needed to address compatibility issues that can be resolved in Chrome. Enterprise administrators may continue to use the UiAutomationProviderEnabled policy to either opt-in early to the new behavior, or to temporarily opt-out through Chrome 136.
- **Chrome 137 on Windows:** The UiAutomationProviderEnabled policy will be removed from Chrome. All clients will use the browser's UI Automation accessibility framework provider.

Upcoming Chrome Enterprise Core changes

Remove enterprise policy used for legacy same site behavior

In Chrome 79, we introduced the LegacySameSiteCookieBehaviorEnabledForDomainList policy to revert the SameSite behavior of cookies to legacy behavior on the specified domains. The LegacySameSiteCookieBehaviorEnabledForDomainList policy's lifetime has been extended and will be removed on the milestone listed below.

- **Chrome 132 on Android, ChromeOS, Linux, macOS, Windows:** Remove LegacySameSiteCookieBehaviorEnabledForDomainList policy.

Upcoming Chrome Enterprise Premium changes

DLP file download access prevention

When a file download DLP rule is set by the admin, a scan is triggered after the download is completed, this feature prevents Chrome Enterprise enrolled users from accessing the contents of a downloaded file before a deep scan verdict is returned.
This feature is gated by the existing policy, OnFileDownloadedEnterpriseConnector, and is only available to Chrome Enterprise Premium users.

- **Chrome 132 on ChromeOS, Linux, macOS, Windows**

Previous release notes

Chrome version & targeted Stable channel release date	PDF
Chrome 130: October 9, 2024	PDF
Chrome 129: September 11, 2024	PDF
Chrome 128: August 14, 2024	PDF
Chrome 127: July 17, 2024	PDF
Archived release notes	

Additional resources

- For emails about future releases, sign up here.
- To try out new features before they're released, sign up for the trusted tester program.
- Connect with other Chrome Enterprise IT admins through the Chrome Enterprise Customer Forum.
- How Chrome releases work—Chrome Release Cycle
- Chrome browser downloads and Chrome Enterprise product overviews—Chrome browser for enterprise
- Chrome version status and timelines—Chrome Platform Status | Google Update Server Viewer
- Announcements: Chrome Releases Blog | Chromium Blog
- Developers: Learn about changes to the web platform.

Still need help?

- Google Workspace, Cloud Identity customers (authorized access only)—Contact support
- Chrome browser Enterprise Support—Sign up to contact a specialist
- Chrome Administrators Forum
- Chrome Enterprise Help Center