



Chrome 136 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 April 23, 2025.

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

| | |
|--|-----------|
| Chrome 136 release summary | 2 |
| Current Chrome browser updates | 4 |
| Current Chrome Enterprise Core updates | 8 |
| Current Chrome Enterprise Premium updates | 8 |
| Coming soon | 10 |
| Upcoming Chrome browser updates | 11 |
| Upcoming Chrome Enterprise Core updates | 18 |
| Upcoming Chrome Enterprise Premium updates | 21 |
| Previous release notes | 27 |
| Additional resources | 28 |
| Still need help? | 28 |

Chrome 136 release summary

| Current Chrome browser updates | Security / Privacy | User productivity / Apps | Management |
|---|--------------------|--------------------------|------------|
| Google Lens result presentation updates | | ✓ | |
| Malicious APK download checks (telemetry-only) | ✓ | | |
| Proactive notifications for Chrome Tips on iOS | | ✓ | |
| Custom data directory required for remote debugging | ✓ | | |
| Partitioning <code>:visited</code> links history | ✓ | | |
| Rename <code>string attr()</code> type to <code>raw-string</code> | ✓ | | |
| Update <code>ProgressEvent</code> to use double type for <code>loaded</code> and <code>total</code> | ✓ | | |
| New policies in Chrome browser | | | ✓ |
| Removed policies in Chrome browser | | | ✓ |
| Chrome Enterprise Core | Security / Privacy | User productivity / Apps | Management |
| WebAuthn Support for Remote Desktop Clients on managed devices | ✓ | | ✓ |
| Chrome Enterprise Premium | Security / Privacy | User productivity / Apps | Management |
| New reporting connector: CrowdStrike Falcon Next-Gen SIEM | ✓ | | ✓ |
| URL filtering capabilities on Android | ✓ | | ✓ |
| Upcoming Chrome browser updates | Security / Privacy | User productivity / Apps | Management |
| Removal of Private Network Access enterprise policies | ✓ | | |
| Remove <code>--load-extension</code> command line switch | ✓ | | |

| | | | |
|---|---------------------------|-------------------------------------|-------------------|
| Remove SwiftShader fallback | ✓ | | |
| Align error type thrown for <i>payment</i> WebAuthn credential creation: SecurityError => NotAllowedError | ✓ | | |
| Blob URL Partitioning: Fetching/Navigation | ✓ | | |
| IP Address logging and reporting | ✓ | | |
| Web serial over Bluetooth on Android | ✓ | | |
| Happy Eyeballs V3 | ✓ | | |
| Strict Same Origin policy for Storage Access API | ✓ | | |
| Web App Manifest: <i>update_token</i> and update eligibility | ✓ | | |
| Migrate extensions to Manifest V3 before June 2025 | ✓ | ✓ | ✓ |
| Chrome will remove support for macOS 11 | | | ✓ |
| Isolated Web Apps | ✓ | | ✓ |
| Disallow spaces in non-file:// URL hosts | ✓ | | |
| SafeBrowsing API v4 → v5 migration | ✓ | | |
| UI Automation accessibility framework provider on Windows | | ✓ | |
| Upcoming Chrome Enterprise Core updates | Security / Privacy | User productivity / Apps | Management |
| IP Address logging and reporting | ✓ | | |
| Inactive profile deletion in Chrome Enterprise Core | ✓ | | ✓ |
| Multiple identity support on iOS | | ✓ | |
| Google AgentSpace recommendations in the Chrome omnibox | | ✓ | ✓ |
| Upcoming Chrome Enterprise Premium updates | Security / Privacy | User productivity / Apps | Management |
| URL filtering capabilities on iOS | ✓ | | ✓ |

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.

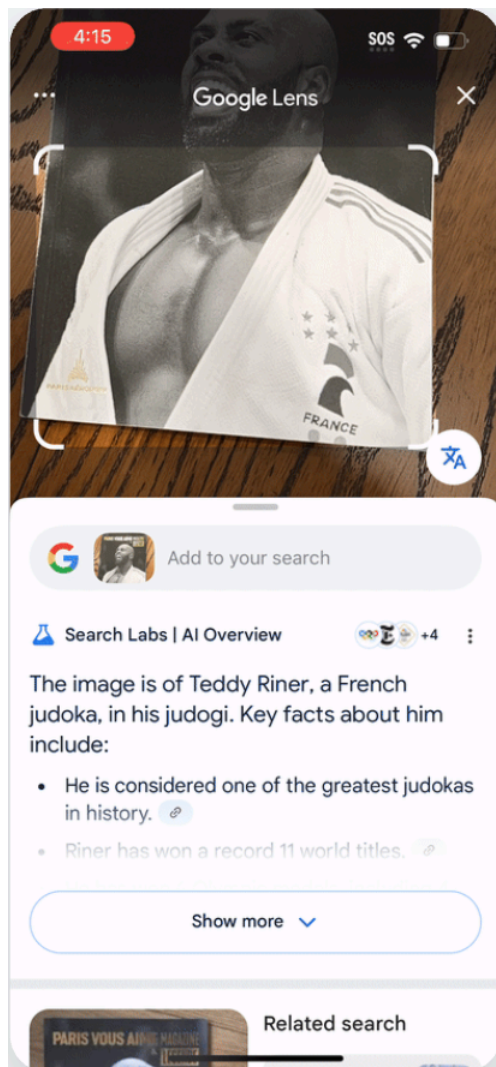
Chrome Enterprise and Education release notes are published in line with the [Chrome release schedule](#), on the Early Stable date for Chrome browser.

Current Chrome browser updates

Google Lens result presentation updates

Search results for Google Lens queries originating from the device camera and from searching images on web pages are presented on a native UI panel that slides from the bottom of the screen. Previously, these answers were presented on a separate web page on a new tab. Admins can control this feature with the existing policy [LensCameraAssistedSearchEnabled](#).

- **Chrome 136 on iOS**



Malicious APK download checks (telemetry-only)

Chrome on Android now contacts Google about Android Package Kit (APK) files downloaded in Chrome, to verify their safety. This is a telemetry-only experimental state of a feature that will eventually show warnings and block downloads of malicious APK files, to protect users against mobile malware. At this time, the malicious APK download check is telemetry-only: no warnings will be shown and downloads will not be blocked. In telemetry-only mode, the malicious APK download check will only be performed for users enrolled in Enhanced Protection from Google Safe Browsing.

This feature can be disabled by setting the Safe Browsing mode to *NoProtection* (value 0) via the [SafeBrowsingProtectionLevel](#) policy.

- **Chrome 136 on Android**

Proactive notifications for Chrome Tips on iOS

Users can now receive [Chrome Tips](#) as provisional notifications. Previously, only users who have explicitly opted into Chrome Tips notifications would receive these helpful notifications.

In this release, Chrome sends these proactively as notifications to users who have installed Chrome on iOS, but have been inactive for several days. In this way, users won't even have to open the app to learn about useful features such as Google Lens or Enhanced Safe browsing. Admins can turn these off by using the policy **ProvisionalNotificationsAllowed** (policy will be available in Chrome 137).

- **Chrome 136 on iOS**

Custom data directory required for remote debugging

Remote debugging via a TCP port or a pipe is no longer possible in Google Chrome with the default data directory on Windows, Linux, and macOS. A custom data directory must be specified to remotely debug Google Chrome using the `--user-data-dir` switch, when using the `--remote-debugging-pipe` or `--remote-debugging-port` switches.

We've made this change because these remote debugging switches are being abused by infostealers and malware to extract data from Google Chrome. A custom user data directory uses a different encryption key and so it prevents malware stealing encrypted data such as cookies.

This change does not affect Chrome for Testing and Chromium.

- **Chrome 136 on Linux, macOS, Windows**

Partitioning *:visited* links history

To eliminate user browsing history leaks, anchor elements are styled as `:visited` only if they have been clicked from this top-level site and frame origin before. On the browser-side, this means that the `VisitedLinks` hashtable is now partitioned by triple-keying, that is, by storing the following for each visited link: `<link URL, top-level site, frame origin>`. By only styling links that have been clicked on this site and frame before, the many side-channel attacks that have been developed to obtain `:visited` links styling information are now obsolete, as they no longer provide sites with new information about users.

There is an exception for *self-links*, where links to a site's own pages can be styled as `:visited` even if they have not been clicked on in this exact top-level site and frame origin before. This exemption is only enabled in top-level frames or subframes which are same-origin with the top-level frame. The privacy benefits above are still achieved because sites already know which of its subpages a user has visited, so no new information is exposed. This was a community-requested exception which improves user experience as well.

- **Chrome 136 on Windows, macOS, Linux, Android**

Rename `attr()` type *string* keyword to *raw-string*

The `attr()` type argument specifies how the attribute value is parsed into a CSS value. In a recent decision by the [W3C CSS Working Group](#), it was resolved to rename the `attr()` type `string` keyword to the more explicit `raw-string`. If the attribute value is given as the `raw-string` keyword, or omitted entirely, it causes the attribute's literal value to be treated as the value of a CSS string, with

no CSS parsing performed at all (including CSS escapes, whitespace removal, comments, etc). No value triggers fallback; only the lack of the attribute entirely does.

For more details about [attr\(\)](#) notation, see [CSS Values and Units Module Level 5](#).

- **Chrome 136 on Windows, macOS, Linux, Android**

Update ProgressEvent to use double type for *loaded* and *total*

The ProgressEvent has attributes `loaded` and `total` indicating the progress, and their type is `unsigned long long` now.

With this feature, the type for these two attributes is changed to `double` instead, which gives the developer more control over the value. For example, the developers can now create a ProgressEvent with the `total` of 1 and the `loaded` increasing from 0 to 1 gradually. This is aligned with the default behavior of the `<progress>` HTML element if the `max` attribute is omitted. For more details, see this [Web Hypertext Application Technology working group \(WHATWG\)](#) discussion on [GitHub](#).

- **Chrome 136 on Windows, macOS, Linux**

New policies in Chrome browser

| Policy | Description |
|---|---|
| <u>OnSecurityEventEnterpriseConnector</u> | Configuration policy for the OnSecurityEvent Chrome Enterprise Connector (now available on Android) |
| <u>WebAuthenticationRemoteDesktopAllowedOrigins</u> | Allowed Origins for Proxied WebAuthn Requests from Remote Desktop Applications. |
| <u>ReduceAcceptLanguageEnabled</u> | Control Accept-Language Reduction. |
| <u>HappyEyeballsV3Enabled</u> | Use the Happy Eyeballs V3 algorithm. |
| <u>EnterpriseRealTimeUrlCheckMode</u> | Check Safe Browsing status of URLs in real time (now available on Android). |
| <u>ProvisionManagedClientCertificateForBrowser</u> | Enables the provisioning of client certificates for managed browsers. |

Removed policies in Chrome browser

| Policy | Description |
|------------------------------------|---|
| ThirdPartyBlockingEnabled | Enable third party software injection blocking. |
| ProfilePickerOnStartupAvailability | Profile picker availability on startup. |

Current Chrome Enterprise Core updates

WebAuthn support for Remote Desktop Clients on managed devices

This change allows users on managed devices to securely access websites on remote hosts using their local security keys or passkeys. With the new [WebAuthenticationRemoteDesktopAllowedOrigins](#) enterprise policy, administrators will be able to specify which remote desktop client applications can make WebAuthn requests on behalf of other origins.

This addresses the challenge of using local authenticators with remote desktops, thereby enhancing both security and user experience. Administrators configure this policy by providing a comma-separated list of allowed remote desktop client app origins.

- **Chrome 136 on Android, ChromeOS, Linux, macOS, Windows**

Current Chrome Enterprise Premium updates

New reporting connector: CrowdStrike Falcon Next-Gen SIEM


Chrome 136 introduces a new Chrome Enterprise reporting connector for CrowdStrike Falcon Next-Gen SIEM. Admins can configure this connector in the Admin console to forward selected


Chrome event data to CrowdStrike for enhanced security monitoring and analysis. This provides more flexibility in SIEM choices and helps improve threat detection.


- **Chrome 136 on ChromeOS, Linux, macOS, Windows**


×


Set up a provider


 Google Security Operations


 Reporting


[SET UP](#) [LEARN MORE](#) 


 Google Cloud Pub/Sub


 Reporting


[SET UP](#) [LEARN MORE](#) 


 Splunk

 Reporting

[SET UP](#) [LEARN MORE](#) 

 CrowdStrike Falcon Next-Gen

 Reporting

[SET UP](#) [LEARN MORE](#) 

×

CrowdStrike Falcon Next-Gen configuration

Configuration name


new-config

Ingest Token

auheifhewitfhethwiuefhe

Host Name

cloud.google.com

 Test connection


Events this configuration is allowed to receive


User & browser events

Default event types [Learn more](#)

Allow all ▼

Optional event types

☒ Login 

☒ Password Breach 

Device events

Default event types [Learn more](#)

Allow all ▼

SAVE CONFIGURATION

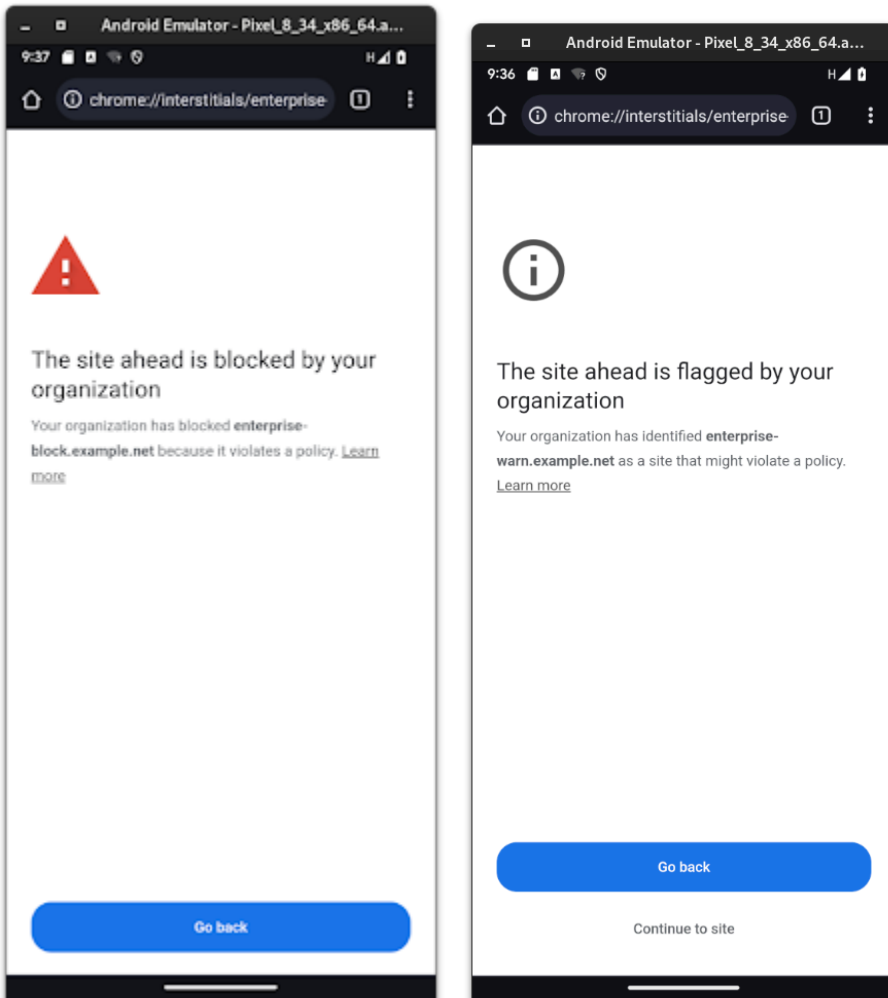
CANCEL

URL filtering capabilities on Android

WebProtect URL filtering is now extended to Android for Chrome Enterprise Premium customers. This allows admins to apply URL block, warn, or audit rules on managed Android devices via the [EnterpriseRealTimeUrlCheckMode](#) policy, providing consistent web content control across

platforms. Filter events are reported via the Reporting Connector, and configuration is done in the Admin console.

- **Chrome 136 on Android**



Coming soon

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

Upcoming Chrome browser updates

Removal of Private Network Access enterprise policies

Private Network Access (PNA 1.0) is an unshipped security feature designed to limit website access to local networks. Due to deployability concerns, PNA 1.0 was never able to ship by default, as it was incompatible with too many existing devices.

PNA 1.0 required changes to devices on local networks. Instead, Chrome is implementing an updated proposal, Private Network Access 2.0 (PNA 2.0). PNA 2.0 ([Github](#)) only requires changes to sites that need to access the local network, rather than requiring changes to devices on the local network. Sites are much easier to update than devices, and so this approach should be much more straightforward to roll out.

The only way to enforce PNA 1.0 is via enterprise policy. To avoid regressing security for enterprise customers opting-in to PNA 1.0 prior to shipping PNA 2.0, we will maintain the

[PrivateNetworkAccessRestrictionsEnabled](#) policy, which causes Chrome to send special preflight messages, until such time that it becomes incompatible with PNA 2.0.

The [InsecurePrivateNetworkRequestsAllowedForUrls](#) and [InsecurePrivateNetworkRequestsAllowed](#) policies, which loosen PNA 1.0 restrictions, will be removed immediately. These policies currently have no effect, since PNA 1.0 is not shipped, and they will have no meaning once PNA 1.0 is removed.

- Chrome 135 on Android, ChromeOS, Linux, macOS, Windows, Fuchsia: Removal of [InsecurePrivateNetworkRequestsAllowedForUrls](#) and [InsecurePrivateNetworkRequestsAllowed](#) policies.
- **Chrome 137 on Android, ChromeOS, Linux, macOS, Windows, Fuchsia:** Removal of [PrivateNetworkAccessRestrictionsEnabled](#). This is dependent on a not-yet-defined replacement policy to enable PNA 2.0 being available.

Remove `--load-extension` command line switch

To enhance the security and stability of the Chrome browser for our users, official Chrome branded builds will begin to deprecate the ability to load extensions using the `--load-extension` command-line flag, starting in Chrome 137. This change aims to mitigate the risks associated with harmful and unwanted extensions.

Unpacked extensions can be loaded with the **Load unpacked** button on the **Extensions** management page (`chrome://extensions/`) with **Developer mode** enabled.

Developers can still use the `--load-extension` switch in non-branded builds such as Chromium and [Chrome For Testing](#).

- **Chrome 137 on ChromeOS, Linux, macOS, 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. It is important to test and handle WebGL context creation failure and fall back to other web APIs such as `Canvas2D` or an appropriate message to the user. A temporary enterprise policy will be available to revert the change.

- **Chrome 137 on Linux, macOS:** Swiftshader will be disabled on macOS and Linux as early as Chrome 137. Users on machines without a GPU will not be able to use WebGL.
- **Chrome 137 on Windows:** SwiftShader will be disabled and replaced with another software WebGL fallback, WARP. Tests depending on the exact pixel values generated by SwiftShader may start failing.

Align error type thrown for `payment` WebAuthn credential creation: `SecurityError` => `NotAllowedError`

This change corrects the error type thrown during WebAuthn credential creation for `payment` credentials. Due to a historic specification mismatch, creating a `payment` credential in a cross-origin iframe without a user activation would throw a `SecurityError` instead of a `NotAllowedError`, which is what is thrown for non-payment credentials. Code that previously detected the type of error thrown, for example, `e instanceof SecurityError`, would be affected. Code that just generally handles errors during credential creation, for example, `catch (e)`, will continue to function correctly.

- **Chrome 137 on Windows, macOS, Linux, Android**

Blob URL Partitioning: Fetching/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 top-level navigations which 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 top-level 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 the relevant specs have been updated to reflect 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 137 on Windows, macOS, Linux, Android**

Web serial over Bluetooth on Android

This feature allows web pages and web apps to connect to serial ports over Bluetooth on Android devices. Chrome on Android now supports Web Serial API over Bluetooth RFCOMM. Existing enterprise policies ([DefaultSerialGuardSetting](#), [SerialAllowAllPortsForUrls](#), [SerialAllowUsbDevicesForUrls](#), [SerialAskForUrls](#), and [SerialBlockedForUrls](#)) on other platforms are enabled in future_on states for Android. All policies except [SerialAllowUsbDevicesForUrls](#) will be enabled after the feature is enabled. [SerialAllowUsbDevicesForUrls](#) will be enabled in a future launch after Android provides system level support of wired serial ports.

- **Chrome 137 on Android**

Happy Eyeballs V3

This launch is an internal optimization in Chrome that implements [Happy Eyeballs V3](#) to achieve better network connection concurrency. Happy Eyeballs V3 performs DNS resolutions asynchronously and staggers connection attempts with preferable protocols (H3/H2/H1) and address families (IPv6/IPv4) to reduce user-visible network connection delay. This feature is gated by a temporary policy [HappyEyeballsV3Enabled](#).

- **Chrome 138 on Android, ChromeOS, Linux, macOS, Windows**

Strict Same Origin policy for Storage Access API

This feature updates the Storage Access API semantics to strictly follow the Same Origin policy, to enhance security. This means using `document.requestStorageAccess()` in a frame only attaches cookies to requests to the iframe's origin (not site) by default. Note: the [CookiesAllowedForUrls](#) policy or Storage Access Headers might still be used to unblock cross-site cookies.

- **Chrome 138 on Windows, macOS, Linux, Android**

Web App Manifest: *update_token* and update eligibility

Introduces an `update_token` field and updates the eligibility algorithm to the manifest spec. This makes the update process more deterministic and predictable, giving the dev more control over whether (and when) updates should apply to existing installations, and allowing removal of the `update_check_throttle` that user agents currently need to implement to avoid wasting network resources.

- **Chrome 138 on Windows, macOS, Linux**
- Chrome 139 on Android

Migrate extensions to Manifest V3 before June 2025

Extensions must be updated to leverage Manifest V3. Chrome extensions are transitioning to a new manifest version, Manifest V3. This will bring improved privacy for your users—for example, by moving to a model where extensions modify requests declaratively, without the ability to see individual requests. This also improves extension security, as remotely hosted code will be disallowed on Manifest V3.

Beginning June 2024, Chrome will gradually disable Manifest V2 extensions running in the browser. An enterprise policy, [ExtensionManifestV2Availability](#), can be used to test Manifest V3 in your organization ahead of the migration. Additionally, machines on which the policy is enabled will not be subject to the disabling of Manifest V2 extensions until the following year - June 2025 - at which point the policy will be removed.

You can see which Manifest version is being used by all Chrome extensions running on your fleet using the **Apps & extensions usage** page in Chrome Enterprise Core.

- **Chrome 127 on ChromeOS, LaCrOS, Linux, macOS, Windows:** Chrome will gradually disable Manifest V2 extensions on user devices. Only those with the [ExtensionManifestV2Availability](#) enterprise policy enabled would be able to continue using Manifest V2 extensions in their organization.
- **Chrome 139 on ChromeOS, Linux, macOS, Windows:** Removal of [ExtensionManifestV2Availability](#) policy.

Chrome will remove support for macOS 11

Chrome 138 will be the last release to support macOS 11; Chrome 139+ will no longer support macOS 11, which is outside of its support window with Apple. Running on a supported operating system is essential to maintaining security.

On Macs running macOS 11, Chrome will continue to work, showing a warning infobar, but will not update any further. If a user wishes to have their Chrome be updated, they need to update their computer to a supported version of macOS.

For new installations of Chrome 139+, macOS 12+ will be required.

- **Chrome 139 on Windows, macOS, Linux**

Isolated Web Apps

Isolated Web Apps (IWAs) are an extension of existing work on PWA installation and Web Packaging that provide stronger protections against server compromise and other tampering that is necessary for developers of security-sensitive applications.

Rather than being hosted on live web servers and fetched over HTTPS, these applications are packaged into Web Bundles, signed by their developer, and distributed to end-users through one or more of the potential methods described in the [Isolated Web Apps explainer](#) on GitHub.

In this initial release, IWAs will only be installable through an admin policy on enterprise-managed ChromeOS devices.

- **Chrome 140 on Windows:** This rollout adds support for Isolated Web Apps in enterprise-managed browser configurations on Windows.

Disallow spaces in non-file:// URL hosts

According to the [URL Standard specification](#), URL hosts 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](#) and [URL focus](#) areas. 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](#)).

Thus this status entry tracks the work to bring Chromium closer to spec compliance by forbidding spaces for non-file URLs only.

- **Chrome 141 on Android, ChromeOS, LaCrOS, Linux, macOS, Windows, Fuchsia**

SafeBrowsing API v4 → 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. For more details, see [Migration From V4 - Safe Browsing](#).

- **Chrome 145 on Android, iOS, ChromeOS, Linux, macOS, Windows**

UI Automation accessibility framework provider on Windows

Starting in Chrome 126, Chrome will start 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 can use the [UiAutomationProviderEnabled](#) enterprise policy starting in 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 146, and will be removed in Chrome 147. 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 146.
- **Chrome 147 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 updates

IP Address logging and reporting

Chrome Enterprise will enhance security monitoring and incident response capabilities by collecting and reporting local and remote IP addresses and sending those IP addresses to the [Security Investigation Tool](#) (SIT) logs. In addition, Chrome Enterprise will allow admins to optionally send the IP addresses to first-party and third-party security information and event management (SIEM) providers via the Chrome Enterprise reporting connector. For more details, see [Manage Chrome Enterprise reporting connectors](#). This will be available for Chrome Enterprise Core and Chrome Enterprise Premium customers.

- **Chrome 137 on Windows, macOS, Linux**

Inactive profile deletion in Chrome Enterprise Core

In April 2025, the inactive period for profile deletion policy starts rolling out. In June 2025 (Chrome 138), the policy will begin to automatically delete managed profiles in the Admin console that have been inactive for more than the defined inactivity period. When releasing the policy, the inactivity

period of time has a default value of 90 days. Meaning that by default, all managed profiles that have been inactive for more than 90 days are deleted from your account. Administrators can change the inactive period value using this policy. The maximum value to determine the profile inactivity period is 730 days and the minimum value is 28 days.

If you lower the set policy value, it might have a global impact on any currently managed profiles. All impacted profiles will be considered inactive and, therefore, be deleted. This does not delete the user account. If an inactive profile is reactivated on a device, that profile will reappear in the console.

If you lower the set policy value, it might have a global impact on any currently managed profiles. All impacted profiles will be considered inactive and, therefore, be deleted. This does not delete the user account. If an inactive profile is reactivated on a device, that profile will reappear in the console.

- **Chrome 138 on Android, ChromeOS, Linux, macOS, Windows:** Policy will roll out in April. Deletion will start in June and the initial wave of deletion will complete by the end of July. After the initial deletion rollout, inactive profiles will continue to be deleted once they have reached their inactivity period.

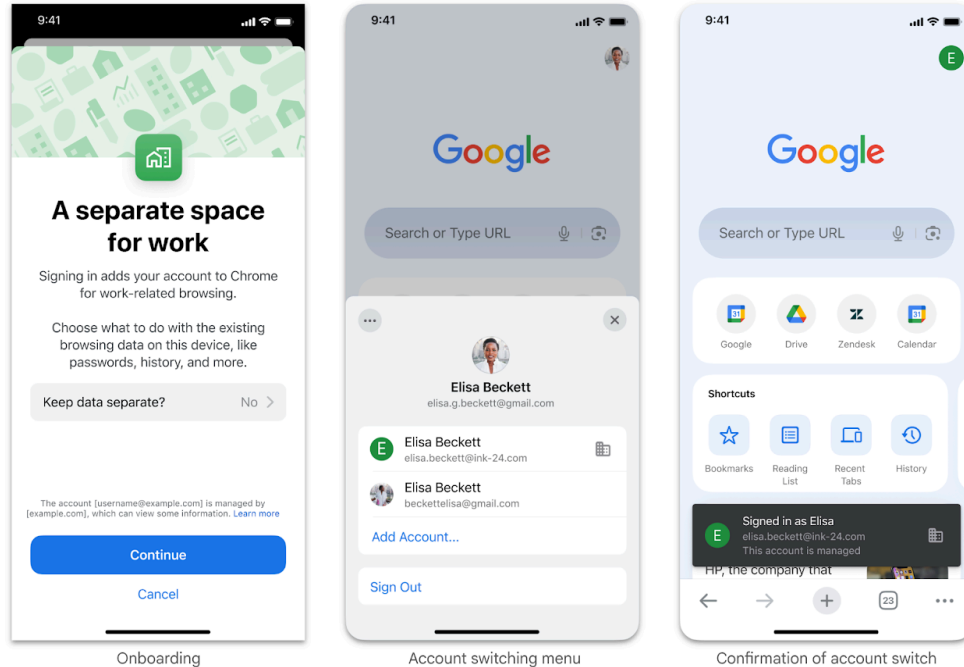
Multiple identity support on iOS

Chrome on iOS is introducing support for multiple accounts, particularly for managed (work/school) accounts. This update introduces separate browser profiles for each managed account, ensuring strict data separation between work and personal browsing. Regular accounts will continue to share a single profile. This change aims to improve Chrome's enterprise offering and provide a more secure and organized browsing experience, especially for end users with both personal and work accounts on their device. Users will experience a one-time onboarding flow when adding a managed account to the device. They will be able to switch between accounts by tapping on the account particle disk on the **New tab** page.

Admins can continue to use the following existing policies to manage iOS accounts:

- Chrome on iOS: Allows admins to apply policies to signed in users on iOS. For more information, see [Turn on Chrome browser management \(Android and iOS\)](#).
- [ProfileSeparationDataMigrationSettings](#): This policy affects the onboarding experience and how prior browsing data is handled when a user adds a work profile.
- [BrowserSignin](#): It allows you to specify if the user can sign in to Google Chrome.
- [RestrictAccountsToPatterns](#): Controls which accounts are visible on the device.

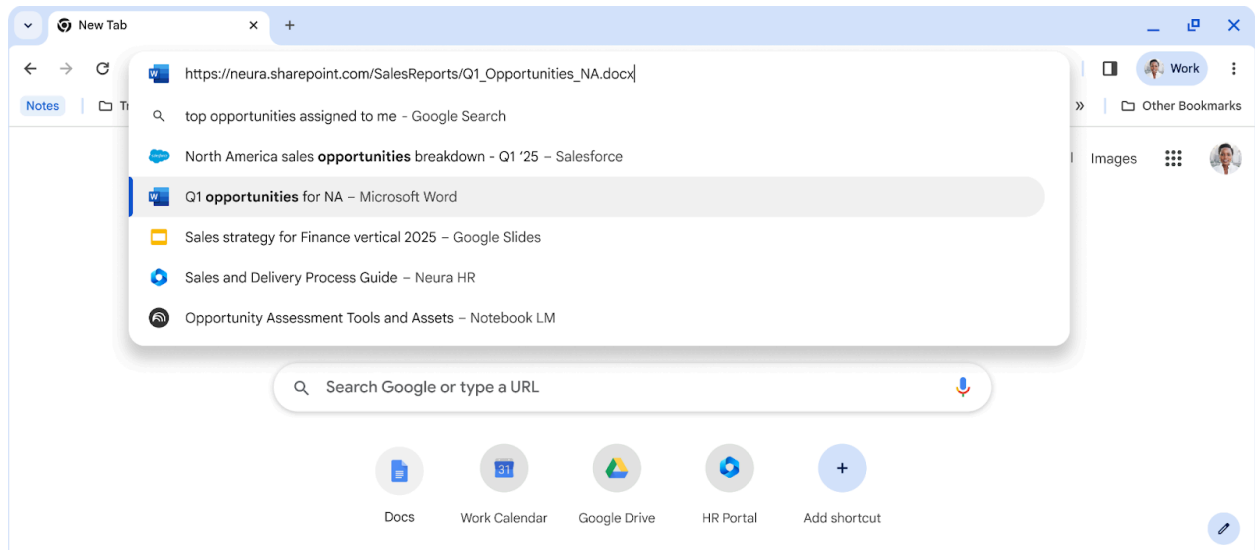
- **Chrome 138 on iOS**



Google Agentspace recommendations in the Chrome omnibox

To help enterprise users with their internal information needs, Admins will soon be able to add enterprise search results, such as people, file, or query suggestions, from [Google Agentspace](#) to the Chrome address bar. Results can be shown by default in Chrome's address bar recommendations or only when triggered by a custom keyword. Users can also use keyword mode to trigger actions through Agentspace, such as *help me write an email that summarizes the current project status*. The enterprise search provider will be shown when the user types @ in the address bar. The organization will be able to customize a keyword or shortcut and the icon shown. This can be configured via the policy called [EnterpriseSearchAggregatorSettings](#).

- **Chrome 139 on ChromeOS, Linux, macOS, Windows**



Upcoming Chrome Enterprise Premium updates

URL filtering capabilities on iOS

The current WebProtect URL filtering capabilities on Desktop are being extended to mobile so that organizations can audit, warn, or block certain URLs or categories of URLs from loading on managed Chrome browsers or managed user profiles on mobile devices. This feature is part of Chrome Enterprise Premium and aims to provide secure and safe internet access for enterprise users on any device. Admins will be able to create URL filtering rules to ensure that employees can only access safe and authorized URLs on iOS devices. Chrome will report URL filtering events and unsafe site events via the Reporting Connector on mobile.

- **Chrome 137 on iOS:** In this milestone, the URL Filtering feature will be launched on iOS for Chrome Enterprise Premium customers. This will enable administrators to manage which URLs can be accessed on managed Chrome browsers or profiles on company-owned or BYOD iOS devices.

Key changes include:

- Admins can block, warn, or audit users when accessing certain sites or categories.

- End-users will see interstitial pages when attempting to visit blocked or warned URLs.
- Chrome will report URL filtering events.
- Updates to the `chrome://management` page will reflect the new functionality.

Admin Search for users, groups or settings

Rules

Google protects you by default

With **system defined rules**, you will be notified when important events occur in your organization, like phishing, malware, suspicious activities, and more. [Learn more](#)

[View list](#)

Collaborate securely

Use **trust rules** to help your users collaborate and flexibly, both inside and outside your orga [Learn more](#)

[View list](#) [Create rule](#)

| Name | Status | Rule type | Actions | Alerts | Last modified |
|------------|--------|-----------------|---------|--------|---------------|
| test print | Active | Data protection | Block | Off | 5/17/20 |

Create rule dropdown menu:

- Activity
- Chrome action
- Data protection**
- Trust

Previous release notes

| Chrome version & targeted Stable channel release date |
|---|
| Chrome 134: February 26, 2025 |
| Chrome 133: January 9, 2025 |
| Chrome 132: January 8, 2025 |
| Chrome 131: November 6, 2024 |
| 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](#)

Google and related marks and logos are trademarks of Google LLC. All other company and product names are trademarks of the companies with which they are associated.