MENLO SECURITY

White Paper

CISO's Guide to Enterprise Browsers

Finding the right approach to protecting the browser

Over the past few decades, digital transformation has driven fundamental change within organizations: redesigning supply chains, enhancing customer experiences, and more. Additionally, the transition of enterprise applications to Software as a Service (SaaS) platforms has further driven the continuation of digital transformation. The combination of these two fundamental changes has reshaped the way that organizations work.

One of these changes involves the application employees use for the majority of their day-to-day activities: the browser. According to Forrester, the typical enterprise worker spends 75% of their "device time" using their web browser. This includes activities such as accessing business-critical applications, accessing work emails, sharing files, and so much more.

This fact has not gone unnoticed by threat actors. The browser is now the target of choice for initial compromise of the user. According to the Verizon 2022 Data Breach Investigation Report (DBIR), web applications and email—which are primarily accessed via web browsers—constitute the primary attack vectors in security breaches, accounting for over 80% of such incidents. A majority of the client-side 0-days exploited in 2023 are browser vulnerabilities based on the record kept by Google Project Zero[1]. Attackers can then take advantage of these 0-days to install malware or ransomware on the victim's endpoint, or steal sensitive information.

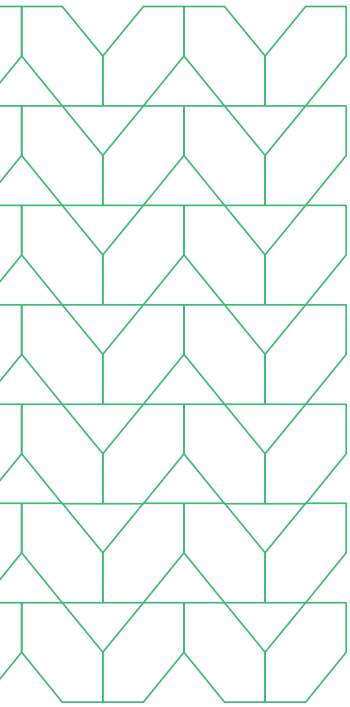
The continued success and evolution of phishing, malware, and ransomware attacks have shown that traditional network and endpoint security-based solutions such as Secure Web Gateways (SWGs), Endpoint Detection and Response (EDR) solutions, and firewalls have limited visibility into browser-based activity. This is because their detection methods rely only on approaches such as pattern matching of known bad threats or relying on classic network signals and don't have complete visibility into browser behaviors. For example, there's been a 70% rise in attacks using categorized reputable sites to bypass URL reputation engines[2]. Threat actors are outsmarting these traditional solutions using the same attacks by varying the sources and methods.

[1] "0day "In the Wild"

[2] Source: Menlo Labs

Security and IT teams should not rely on native browser policies to address these threats. Existing browser policies don't entirely align with what security teams are trying to accomplish to protect against these threats. Typically, policies are hard to push, it's difficult to figure out which policies make sense, and the user experience is often not well-suited for security administrators.

The current solutions are failing to protect the enterprise. Instead, organizations need a comprehensive solution to browser security that addresses all aspects of browser security.



What is Browser Security?

A number of solutions, which can be grouped into three main categories, are dedicated to addressing browser security:

Enterprise Browsers

- **Local Browsers**
  - **Mainstream Browsers**—hese include browsers such as Google Chrome, Microsoft Edge, and Apple Safari. Each browser vendor continues to add security functionality and further secure existing functionality in their browsers.
  - **Replacement Browsers**—These browsers, usually based on Chromium, are designed specifically for enterprise users. These browsers primarily focus on consistently enforcing enterprise policies across user devices.
- **Cloud-Based Browser Security**—A hybrid and flexible solution that includes capabilities of Enterprise Browsers, Browser Extensions, and Secure Cloud Browsing or traditional Remote Browser Isolation, delivered as a cloud-based service.

Enterprise Browsers Extensions—

These are solutions to add functionality to the browser in the form of a browser extension, designed to work with Mainstream Browsers.

Traditional Remote Browser Isolation—

RBI is a web solution designed to separate untrusted web content (typically from the internet) from users and their devices.

Browser Security key capabilities

Browser security can be broken down into three key pillars, each with key capabilities and use cases critical to protect the user, secure the enterprise, and maintain a good user experience for both the administrator and the end user:

- Managing the browser
- Protecting the user
- Securing access and data

Managing the browser

Managing the browser is not a new concept. In fact, the leading browser management platforms, Microsoft Intune and Google Chrome Enterprise Manager offer hundreds of management parameters that can be controlled centrally. Most enterprises are focusing on a minimal set of configuration parameters—notably around browser versioning and extension management, more or less leaving the browser in a default configuration. Best practice for browser management includes configuring an additional, minimal set of parameters to best secure the browser.

Browser management includes the following policy elements that are relevant to the overall security posture of the enterprise:

- Version management
- Enumerating allowed and/or blocked browser extensions.
- Disabling specific browser functionality—for example, disallowing USB interactions
- Limiting access to power user functionality—for example, access to developer tools

These policies need to be deployed and enforced consistently across a heterogeneous browser installed base across the enterprise.

Some enterprises have a specific use case around supporting unmanaged endpoints; an example could be seasonal contractors that need specific application access from a personal device such as a laptop. This specific use case presents some challenges from a legal liability standpoint which vary by region, so for the purpose of this document, those complexities will not be addressed. Browser management can be accomplished in a number of ways here::

- The unmanaged device can be mandated to use an enterprise browser; the user being forced to install an application on their personal equipment.
- The unmanaged device can be mandated to add a browser extension to their existing browser of choice, streamlining the end user learning curve and experience.

In both cases, policy enforcement is possible. This becomes more an issue of user experience, deployment overhead, and local regulations. More on this further in the document.

Protecting the user

Protecting the user is the epicenter of browser security. A solution should prevent all attacks, both common and advanced, including:

- Exploitation of browser vulnerabilities
- Download of malware, including ransomware
- Phishing (including zero-hour protection)

As browsers add capabilities, a constantly new list of potential flaws that bad actors can exploit is created. Between November 2022 and November 2023, 175 high and critical severity issues[3] were fixed by Google in Chrome. Since both Microsoft Edge and all Enterprise Browsers rely on Chromium as the underlying browser engine, virtually all these vulnerabilities affected these browsers as well. Many of these vulnerabilities can lead to arbitrary code execution simply by visiting a malicious website. Attackers also continue to explore the large attack surface that already exists in browsers, finding vulnerabilities that have laid dormant for years.

[3] Total count of issues fixed for the stable channel of Chrome between Nov. 1, 2022 and Nov. 1, 2023, with security severity set to high or critical. Many but not all of these issues have CVEs associated with them

Security and IT teams have to spend significant time reviewing and then addressing this ever-growing list of web vulnerabilities, including zero-day vulnerabilities where browsers need to be patched in short order on a large number of end-user devices. If left unaddressed, these vulnerabilities leave an open door that attackers can use to obtain remote code execution on a targeted system, leading to malware such as ransomware being installed on the victim's system. Despite this risk, more than 25% of pages loaded by enterprise users have been observed to be loaded via browsers that are 2 or more major versions behind the current version of Chrome, browsers with many disclosed high-severity vulnerabilities[4].

[4] Based on measurements collected by Menlo Security over a 24 hour period on Nov. 15, 2023, across hundreds of millions of page loads..

Additionally, on the malware and phishing fronts, attackers are constantly evolving their techniques and infrastructure. This means defenders have to constantly update their detection logic and signature databases. These evolving attacks are leveraging web browsers as their preferred entry point.

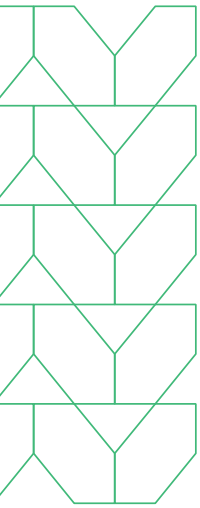
Securing access and data

Elements of this pillar are commonly seen in zero trust network access strategies, and adopted by enterprises that have evolved their remote access strategies and implementations away from the classic layer 3 IPSec VPNs towards a least-privilege application access (note this is application access, not network access) paradigm. This migration towards per application access control creates new opportunities for enterprises:

- Granular, least privilege access to applications—SaaS delivered and private cloud
- Ease of management for administrators
- Improved end user experience
- Infrastructure cost savings

This is usually paired with data leakage protection including:

- Document and archived file isolation
- Data masking
- Watermarking
- Read-Only access to common sharing platforms



As mentioned earlier, the majority of enterprise applications have transitioned from legacy client/server to modern, web-based applications. This allows enterprises to leverage browser security to create simple, yet powerful policies for access, as well as fine-grained group and per-user application-specific policies. Further, enterprises can create these fine-grained policies even in situations when the application itself does not support that level of policy natively.

Enterprises previously relying on virtual desktop infrastructure (VDI) want to retire the costly infrastructure and aggravating user experience in exchange for the familiar browser interface, as well as the expected, high performance usability that users have come to expect from modern web applications. At the same time, a robust solution should preserve the strong guarantees that are provided by VDI: sensitive information does not reach the endpoint unless it is displayed on the screen and backend servers are not directly exposed to potentially malicious clients.

Legacy web applications and immature SaaS solutions often lack fine-grained control to limit the exposure of sensitive data to users, forcing the addition of another content-filtering layer to redact content. Even when fine-grained controls exist, they can be challenging to configure on a per-application basis, or application configuration may not be under the control of the security team. Beyond viewing web content, a user may need to download a document to only view a few pages or to transfer it to another web application without ever needing to open the document. Finally, once web content or a document has been made accessible to a user, they may paste or upload this content in unsanctioned SaaS applications, intentionally or due to a lack of familiarity with company policies. A common scenario for this is the use of ChatGPT and similar generative AI sites. In all these scenarios, the browser security layer needs to help consistently and robustly enforce content-filtering policies.

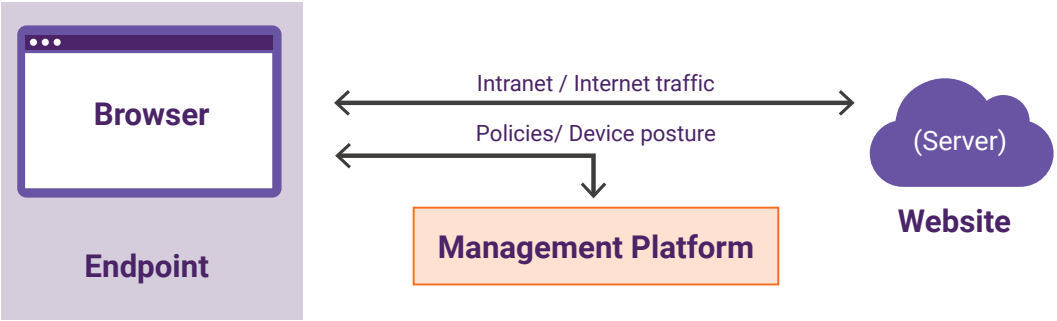
In addition to limiting the exposure of sensitive data to untrusted or potentially compromised endpoints, enterprises want to reduce the exposure of their web application servers to these clients. A malicious client can send exploit payloads to vulnerable servers. The Log4J vulnerability is the most recent high-profile example of the exposure that exists on the server side. A secure browser solution needs to help protect these servers from vulnerabilities, both known and yet unreported until they can be patched.

Local Browsers

Local browsers address the problem space of browser security by delivering a two component solution:

- A centralized (typically cloud based) management platform
- A browser

The browser is typically built atop Chromium with enterprise centric capabilities such as policy enforcement, DLP capabilities, and local browser and file isolation capabilities. For Enterprise Browsers specifically, the premise to the solution is that with an enterprise-first browser, the overall attack surface can be reduced, security policy and lightweight security features can offset the need for a full secure services edge (SSE) deployment and adequately secure the enterprise and the user.



How Local Browsers manage the browser

Local browsers can be managed via an on-premises or cloud-based management platform such as Microsoft Intune or Google Chrome Enterprise Manager, as well as others. Security policy and other settings can be defined and pushed to the local browsers.

For Mainstream Browsers, there are licensing complexities to unlock additional functionality. For example, to manage extensions via Microsoft Intune, customers would need to have a higher tier Enterprise license, a Microsoft Defender license, and an additional license on top of that..

How Local Browsers protect the user

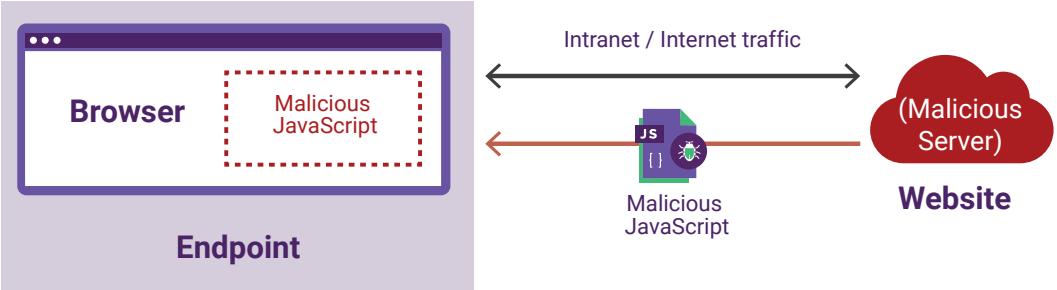
Any software defects or vulnerabilities in the defense mechanisms place the browser, the device, and network at risk if a threat actor or a given attack is able to take advantage of the vulnerabilities.

Browser vulnerabilities:

Mainstream and Enterprise Browsers rely exclusively on defense and protection schemes that run locally on the browser. While potentially effective for known threats, they are susceptible to zero-day threats. To mitigate this shortcoming, they primarily turn to disabling browser functions that have historically been sources of vulnerabilities, such as Just-In-

Time compilation of JavaScript, WebGL, etc. Removing functionality reduces not only the attack surface, but also breaks legitimate web applications that depend on this functionality, wasting users time, increasing support costs as well as overall frustration and dissatisfaction. Additionally, the Chrome engineering team's analysis of security issues in the browser is that "These bugs are spread evenly across our codebase[5]". As a result, many of the 175 vulnerabilities fixed in the past 12 months would still have been exploitable even after severely crippling the functionality of the browser.

[5] https://www.chromium.org/Home/chromium-security/memory-safety/



Malware (including ransomware):

Mainstream and Replacement Browsers provide support for scanning downloaded files for known bad content. This may involve updating a local database of malicious signatures or sending the file to a cloud service for inspection.

Phishing:

Mainstream and Replacement Browsers may offer support to detect content that may be phishing content. They rely on updated lists of known phishing domains, behavioral and content-based analysis may be used to determine in real-time whether or not a page is malicious. They can employ policy-based actions to block access to the page content or block user input.

How Local Browsers secure access and data

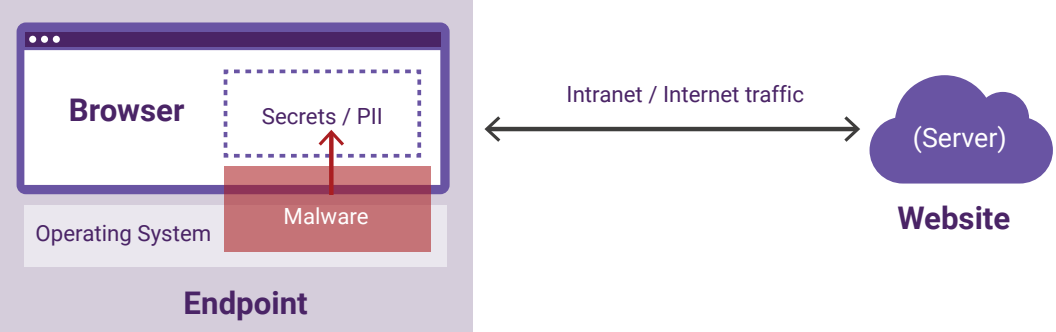
Application Access:

Mainstream and Replacement Browsers aim to limit access to application servers to only the sanctioned browser. This is often achieved by attempting to validate the security posture of the client, including the browser used for access, at application authentication time. The application is configured to use an Identity Provider, such as Okta, and Okta is configured to only allow authentication from IP addresses that belong to the SaaS component of the Mainstream and Replacement Browsers. Once authentication is complete, an authentication token is issued to the browser that can then present it to the application. While this provides a small roadblock that may stop an unmotivated attacker, this is a security-by-obscurity approach that can be circumvented. The sanctioned browser can be impersonated by a browser the attacker fully controls, or an attacker can extract the authentication token from the memory of the sanctioned browser and transfer it to a browser they fully control..

Data exfiltration:

Within Replacement Browsers, sensitive data can be "obscured" so that it cannot be seen. While the obscured information is not shown to the user, it is present in the endpoint memory because obscuring the data is done by the Replacement Browser, not by the server. An attacker with OS or hypervisor control can read the memory of the Replacement Browser and steal the information. They may also offer a feature to "de-obscure" the data. When a user decides to view the obscured data, an audit log entry may be created.

To handle sensitive information in locally-downloaded files, files may be kept in an encrypted area that the Enterprise Browser attempts to shield from the rest of the user's device. The user can see file content through built-in viewers, but may not be able to easily copy the original file outside the fenced area created by the Enterprise Browser.



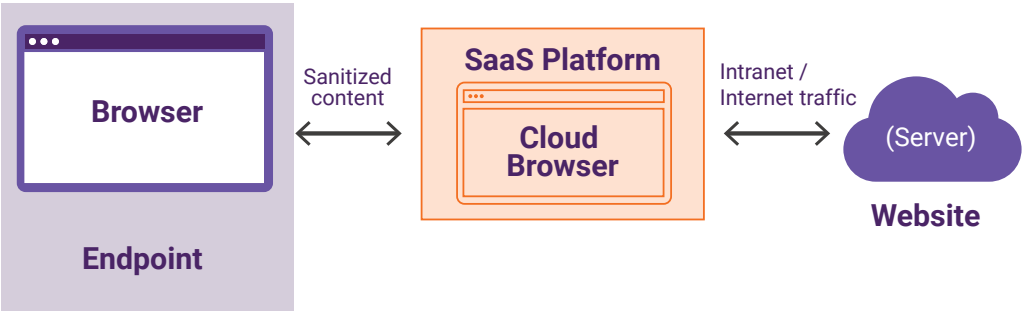
Screenshotting and copy and paste:

Screenshotting and copying of data to the clipboard can be disabled by the Replacement Browser using APIs provided by the client OS, a useful capability. Yet as the data is local to the machine, an attacker on a compromised endpoint can "steal" the document in network buffers before the document is protected by the Replacement Browser. For screenshotting, Replacement Browsers provide control against a non-technical attacker. It is still trivially possible for a non-technical attacker to use their phone to take photos of the screen, although this may be prevented through other means that require further restrictions. A skilled attacker in control of the OS or hypervisor can easily capture the content sent to the screen, short-circuiting any protection provided by the Replacement browser.

Cloud-Based Browser Security

Cloud-based Browser Security delivers a hybrid of the capabilities of Replacement Browsers, Browser Extensions, and Remote Browser Isolation.The browser is configured to route all of its web traffic to the Cloud-based Browser Security platform to block any potentially malicious active content and with sensitive information redacted when appropriate. This approach allows organizations to enable Browser Security for any device, and using any browser. Unlike traditional RBI, Cloud-based Browser Security uses more efficient content-based RBI. This allows a near native user experience and eliminates the high bandwidth requirement of traditional RBI. As a result, the approach is suitable for protecting all of the user's browsing, unlike traditional RBI that is often limited to a small percentage of "risky" websites.

For example, the Menlo Security Platform protects the vast majority of the web activities of millions of users everyday, while also controlling access to the data of thousands of web applications. Because security is delivered through the cloud, this protection follows a user wherever business takes them—in the office, on the road, at a conference, at a customer site, etc.— Moreover, it applies to all users accessing the protected web applications, regardless of the security posture of the user's endpoint. This provides security guarantees for the accessed data equivalent to significantly more expensive and cumbersome approaches such as Virtual Desktop Infrastructure (VDI).



How Cloud-Based Browser Security Manages the browser

Management capabilites:

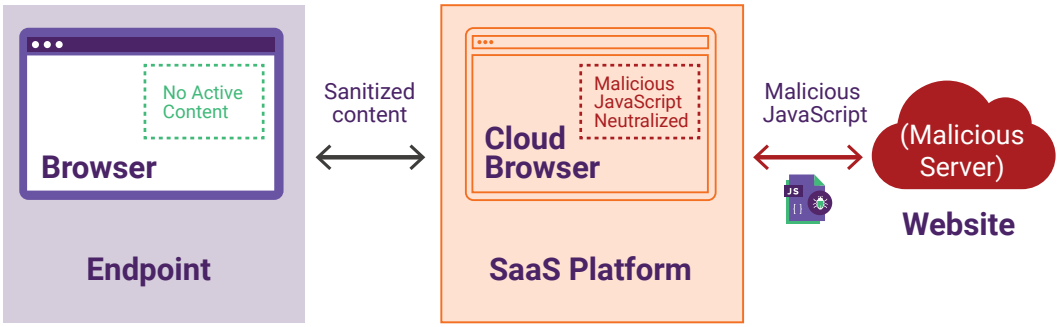
Cloud-based Browser Security takes a browser agnostic approach, making security a focal point when considering browser management. Cloud-based Browser Security leverages and builds upon the management capabilities of Mainstream Browser vendors like Google and Microsoft.

Additionally, this approach streamlines cross-team delegation. Rather than centralizing all policy management within one team, it allows the desktop team to delegate control of a subset of policies to the security team. Cloud-based Browser Security also aggregates and simplifies policy creation in a browser agnostic way. Instead of working on each policy individually, policies are grouped together and can be enabled with a click of a button across multiple browsers.

How Cloud-Based Browser Security protects the user

Browser capabilites:

Cloud-based Browser Security is able to broadly take the attack surface off the table. Instead of having organizations decide on which functionality to turn off in the browser, functionality is run in the cloud by default and not exposed on the endpoint. Even when a feature is leveraged client-side, it is not exposed directly to websites that could maliciously abuse the API. This approach eliminates the need for customers to choose between not supporting a feature and creating a security risk. Instead, the feature can be safely leveraged in the cloud. The attacker's inability to run code on the user's browser means they cannot compromise the endpoint through the aforementioned Chrome vulnerabilities.

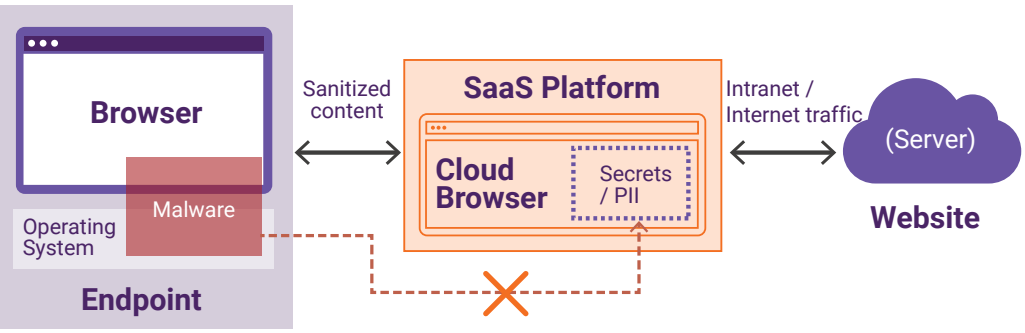


For example, WebGL, a JavaScript API for rendering interactive 2D and 3D graphics within any compatible web browser, can be supported and there is no need to identify WebGL as a source of security issues. This is because all client-side content creation that leverages WebGL happens in the cloud browser. Likewise, Just-In-TIme (JIT) compilation doesn't have to be disabled: the page JavaScript is executed in the cloud where it can safely leverage JIT compilation. As a result, user experience is not affected (disabling JIT compilation can significantly slow down JS heavy pages).

A key part to protecting the user is always running an up-to-date browser. With this approach, the browser is updated automatically in the cloud. This means the browser can be updated even if the user's device is powered off or connecting through a low bandwidth link, where pushing large updates is not practical. Again, there is no trade off between an insecure decision (letting the user browse with an outdated client) and a user unfriendly decision (forcing the user to download an update prior to browsing, even if they have bad connectivity).

Malware (including Ransomware):

The cloud-based browser has full visibility into all files downloaded by the end user and is able to analyze these files ahead of the files being sent to the client. Similar to Mainstream and Replacement browsers, the content of the file can be scanned via AV and Sandbox type approaches. Unlike Mainstream and Replacement Browsers, the scanning engines and signature databases are constantly kept up-to-date on the Cloud-Based Browser Security SaaS platform. And unlike Mainstream and Replacement Browsers, payloads determined to be malicious never touch the endpoint.



Phishing:

Since the cloud-based browser renders all the content sent to the user and observes all user input, it can implement the same type of known bad behavior and content based approaches as a Replacement Browser. However, it can do so without requiring a presence on the endpoint, offering consistent protection to all enterprise users. Because all the logic and signatures used for the detection is executed in the cloud, updated detection capabilities can be instantly deployed to the entire fleet of browsers. Zero-delay protection is key to fighting ever evolving phishing content.

How Cloud-Based Browser Security protects the user

Application Access:

The user is forced to interact with the protected server through the cloud browser. A threat actor has no ability to tamper with the cloud browser and does not control the more privileged layers of the software stack. The threat actor cannot examine the cloud-based browser's memory and steal session tokens to interact with the vulnerable application directly. This is because a secure network path is in place, which enforces all traffic that passes through the Browser Security platform.

This approach is similar to VDI-based approaches where the software is running securely in the cloud away from the attacker. But unlike VDI, where the user can try to install software or otherwise go after the remote OS, a cloud-based approach to Browser Security has zero exposure to the underlying operating system running the browser. In this setup, attackers are limited to clicking around the page and entering keyboard input into forms to target the

cloud browser. A read-only mode can also be enforced, where users are unable to enter data when input is turned off, leaving them to interact with the target site only through mouse input.

Data Exfiltration:

Cloud-based Browser Security does not download the sensitive information to the end device at all. In addition to providing reliable protection for obscured data, this approach can also provide tamper-proof audit logging. This means if a user decides to unobscure a sensitive field, this action can be logged in a way that an attacker cannot tamper with. Should the user prove to be malicious, this will provide an accurate picture of the "blast radius" for the stolen Personally Identifiable Information (PII).

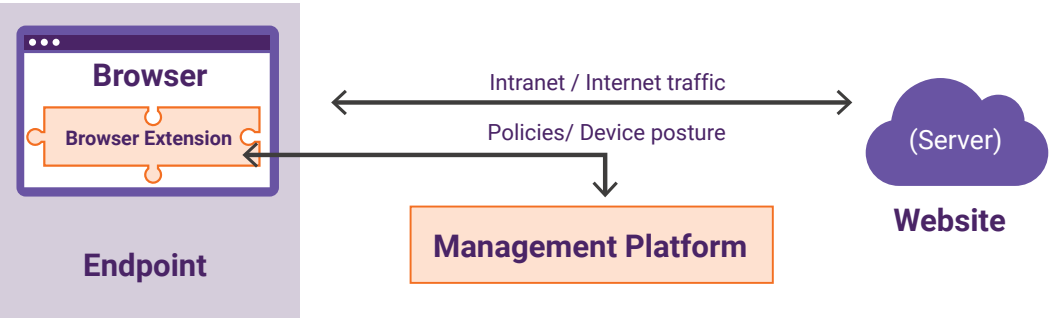
This applies to files as well. They can be kept in a truly protected area in the cloud, away from the user. A user can view only part of the file via a cloud viewer (and the platform can accurately track which portions of the file were displayed) or copy the file between web applications (e.g. take a Gmail attachment and put it in a Box folder) without the file ever touching the endpoint.

Browser Extensions

Browser Extensions installed on existing Mainstream Browsers with the intent of delivering alternative implementations of security features to the Mainstream Browser. There are an array of Browser Extension solutions on the market with varying levels of claimed capability. Browser Extensions and the associated functionality is subject to API policy changes of the browser vendors. The capabilities and APIs that the browser security extensions leverage can also be used by nefarious parties to create malware, and as such, browser vendors have started to restrict the capabilities granted to Browser Extensions. This varies by browser vendor.

How Browser Extensions manage the browser

Browser Extensions can be installed without a lot of interaction with the end user and support almost any browsers users may be utilizing. The extension can then interact with a central management tool.



How Browser Extensions protect the user

Browser vulnerabilities:

Once deployed the extension provides visibility and lightweight protection from web threats. These extensions provide a subset of the capabilities that other browser security solutions provide. They can block content from known bad URLs but cannot more significantly alter the behavior of the browser to reduce attack surface.

Malware (including ransomware):

Downloaded content cannot be viewed and scanned for malicious content, leading to almost no protection against malware

Phishing:

Browser Extensions take a similar approach to Enterprise Browsers for phishing protection.

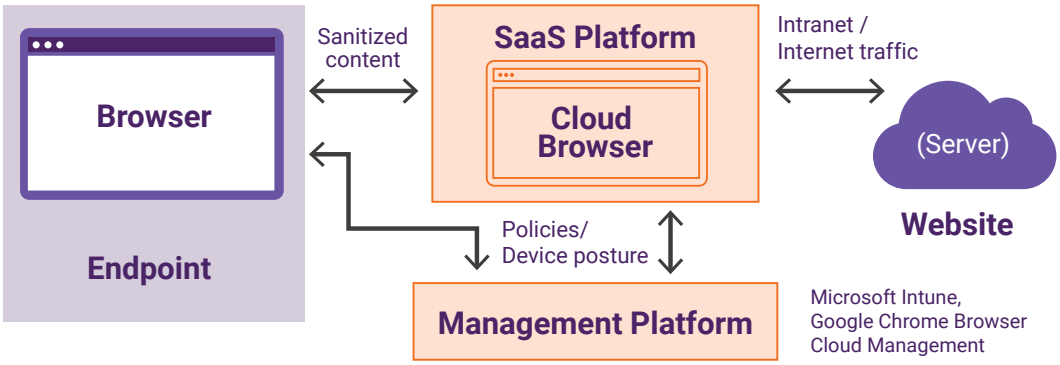
How Browser Extensions secure access and data

Application access:

Browser Extensions provide organization with application visibility. Once installed, these extensions provide information to help identify sensitive users and sensitive applications. With the right information, this enables organizations to understand what actions need to be taken to provide secure access to applications. However, extensions are not recommended for unmanaged devices because they are somewhat easier to uninstall.

Traditional Remote Browser Isolation

Traditional Remote Browser Isolation (RBI) protects organizations from web- and email-based malware using a Zero Trust approach to security. This preventative strategy routes all web traffic through a cloud-based remote browser. It doesn't matter if content is good or bad, categorized or uncategorized, isolation treats everything as potentially malicious—delivering only safe, sanitized content to the end user. In this approach, the rendered pixels are captured from the remote display and transmitted to the client, possibly after adding video compression. This approach requires a high bandwidth link between the user and the cloud platform and often results in a non-native user experience.



How RBI Manages the browser

Not in Scope

How RBI protects the user

Browser vulnerabilities, malware and phishing:

Web traffic, typically uncategorized, is forced through and executed in a virtualized cloud environment. By doing this, any potentially malicious web content is executed away from the user and never touches the endpoint. Instead, the user sees and interacts with a safe rendered down version of the malicious site or attachment.

How RBI secures access and data

Not in Scope

Summary

While there is a broad variety of solutions designed to protect the browser, it's important to identify business needs first and understand the key capabilities of Browser Security to find the right solution for your organization. Enterprises have been deploying aspects of Browser Security for years, via a patchwork of security tools. However as the industry continues to shift to cloud based solutions, ultimately Cloud Based Browser Security is able to best address, at scale, the threats targeting the browser, the user, and applications.

To learn more about Browser Security, visit menlosecurity.com or email us at ask@menlosecurity.com.

MENLO SECURITY

To find out more, contact us:

menlosecurity.com

(650) 695-0695

ask@menlosecurity.com

About Menlo Security

Menlo Security eliminates evasive threats and protects productivity with the Menlo Secure Cloud Browser. Menlo delivers on the promise of cloud-based security—enabling Zero Trust access that is simple to deploy. The Menlo Secure Cloud Browser prevents attacks and makes cyber defenses invisible to end users while they work online, reducing the operational burden on security teams.

Menlo protects your users and secures access to applications, providing a complete enterprise browser solution. With Menlo, you can deploy browser security policies in a single click, secure SaaS and private application access, and protect enterprise data down to the last mile. Secure your digital transformation with trusted and proven cyber defenses, on any browser.

Work without worry and move business forward with Menlo Security.

Browser Security comparison

This table illustrates how each mentioned technology supports critical capabilities within each pillar of browser security. These capabilities are critical to protecting the user, securing the enterprise, and maintain a good user experience for both the administrator and the end user.

		Local Browsers	Browser Extensions	Traditional RBI	Cloud Based Browser Security
Manage the browser	Configuration of the browser		Limited		
	Screenshot capability	Bypassable			
	Device posture check				An agent may be required
	Extensions				
Protect the user	Protection against browser vulnerabilities (including zero-days)			All active content in the cloud	All active content in the cloud
	Protection against Malware		Limited visibility into downloaded files	Engine/signature update is easier ; Cross device	Engine/signature update is easier; Cross device
	Protection against Phishing			Engine/signature update is easier ; Cross device	Engine/signature update is easier; Cross device
Secure access and data	Protection against Data exfiltration	Can be bypassed by skilled attacker		Not in scope	Protected data does not touch endpoint
	Data redaction	Can be bypassed by skilled attacker		Not in scope	
	Watermarking			Not in scope	
	Application access	Can be bypassed by skilled attacker		Not in scope	Only possible path is through cloud browser
	Copy & paste	Can be bypassed by skilled attacker		Not in scope	
	Extension exposure			Not in scope	No exposure to client extensions
	Logging and auditing	Not tamper-proof		Not in scope	