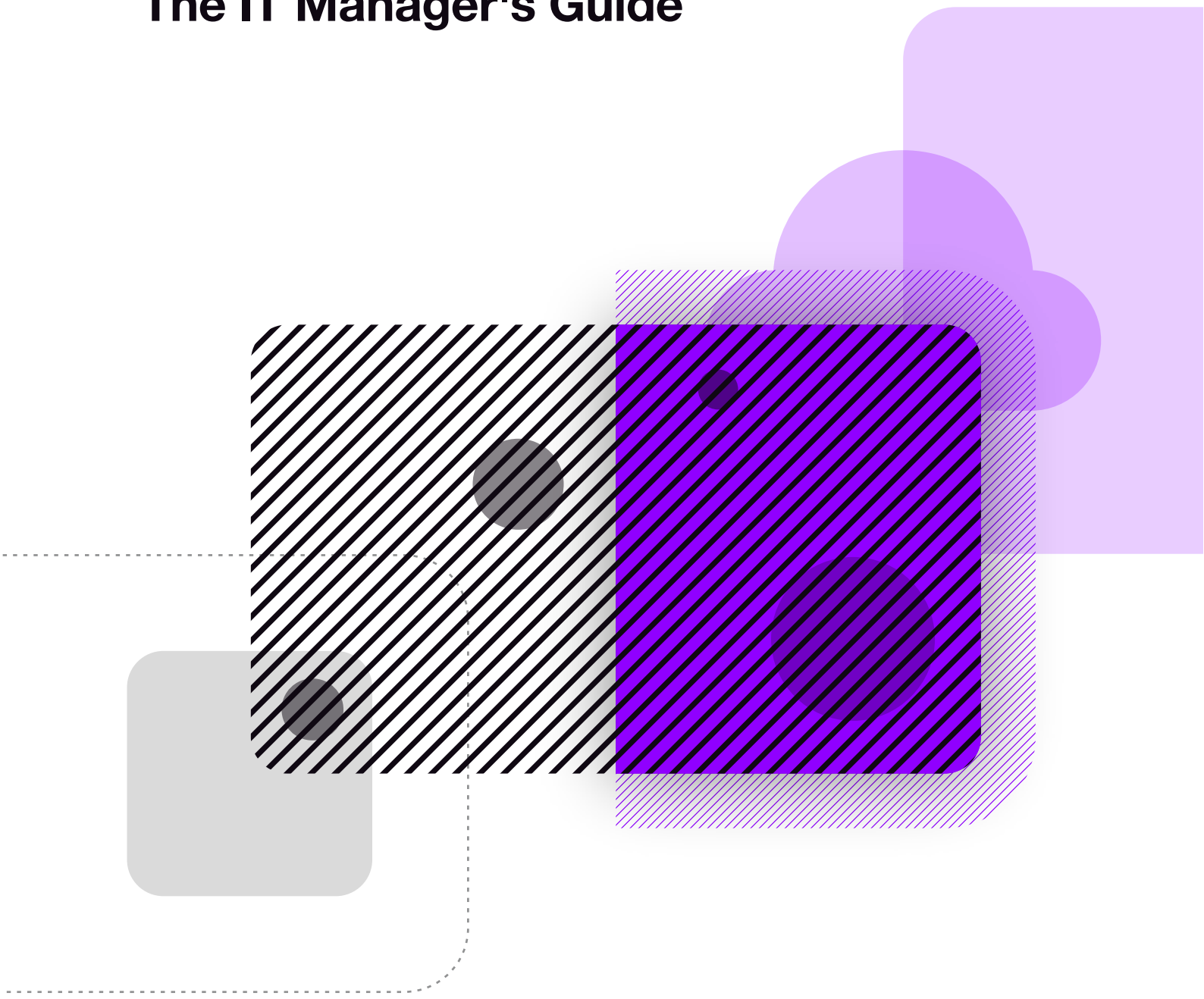
HYSOLATE

Browser Isolation vs OS Isolation

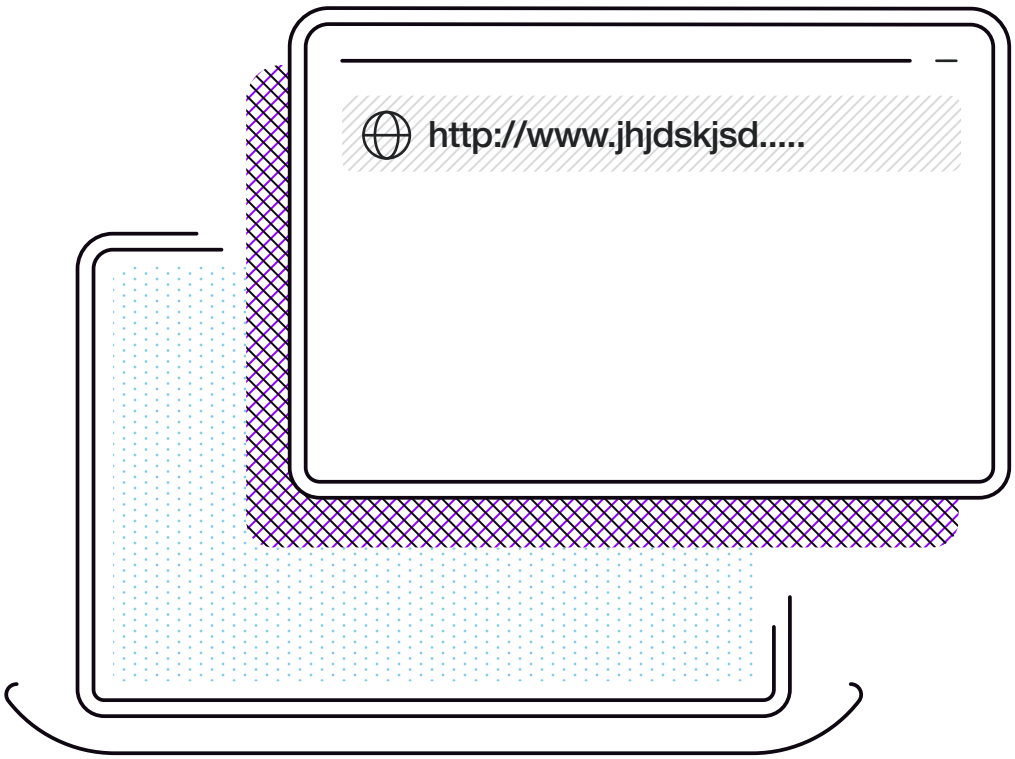
The IT Manager's Guide



What is Browser Isolation?

Browser isolation is a security model that physically isolates internet users' browsing activity from their local computers, networks, and infrastructure. In this model, browser sessions are isolated from the hardware the browser is running on and the Internet connection being used, ensuring that harmful activities cannot affect the user's compute environment as they are contained in the isolated browser environment. This model is also known as a virtual browser.

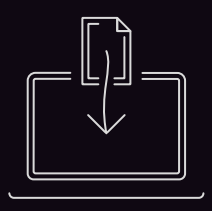
Browser isolation can be done in a number of ways, but usually includes virtualization, containerization, or cloud-based application virtualization. The isolated environment is reset or deleted when the user closes the browsing session or the session times out. In addition, malware and malicious traffic are also discarded, so they do not reach the endpoint device or network.



What Threats Does Browser Isolation Defend Against?

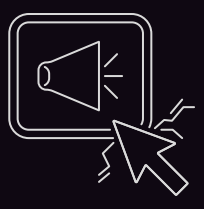
Most modern web pages use JavaScript, and attackers can use JavaScript code to perform a variety of malicious activity on user devices. Because browsers execute JavaScript by default on a web page, these malicious scripts run as soon as a user visits the page. The scripts could be planted by malicious site owners, or by others, unbeknownst to the site owners, as in cross site scripting (XSS) attacks.

This leads to attacks like:



Drive-by downloads, in which the browser downloads files without the user's consent

Malvertising, in which malicious code is executed when the user views an ad



Clickjacking, which involves tricking users into clicking links they did not intend to click

XSS can also be used to hijack user sessions and steal credentials

There are several other browser-based threat vectors, including forced redirects to malicious URLs, and exploiting unpatched browser vulnerabilities.

* Almost all these threats can be prevented by using browser isolation

because malicious activity occurs in an isolated or remote environment, not directly on the user's device. For example, if a malicious script forces a redirection or a drive-by download, this would not affect the user, as the URL or file are executed in an isolated environment.

Browser Isolation: Key Security Features

Blocking malware

Allows users to browse the web without being exposed to malicious downloads or malicious scripts on websites.

Blocking unsafe plugins and technologies

If users access websites rendered with legacy technologies like Adobe Flash, or install plugins that have security vulnerabilities, attacks will be shielded from the personal device.

Document isolation

Many document formats can contain malware. In an isolated browser, users view documents within the isolated environment, meaning that malicious scripts do not affect the local device. After scanning the file for malware, the user can be allowed to download it to their personal device.

Reporting & forensics

With browser isolation, administrators can monitor and audit browsing activity, see when users access unsafe content, and when attacks occur within an isolated browser, determine the root cause.

Types of Isolated Browsing

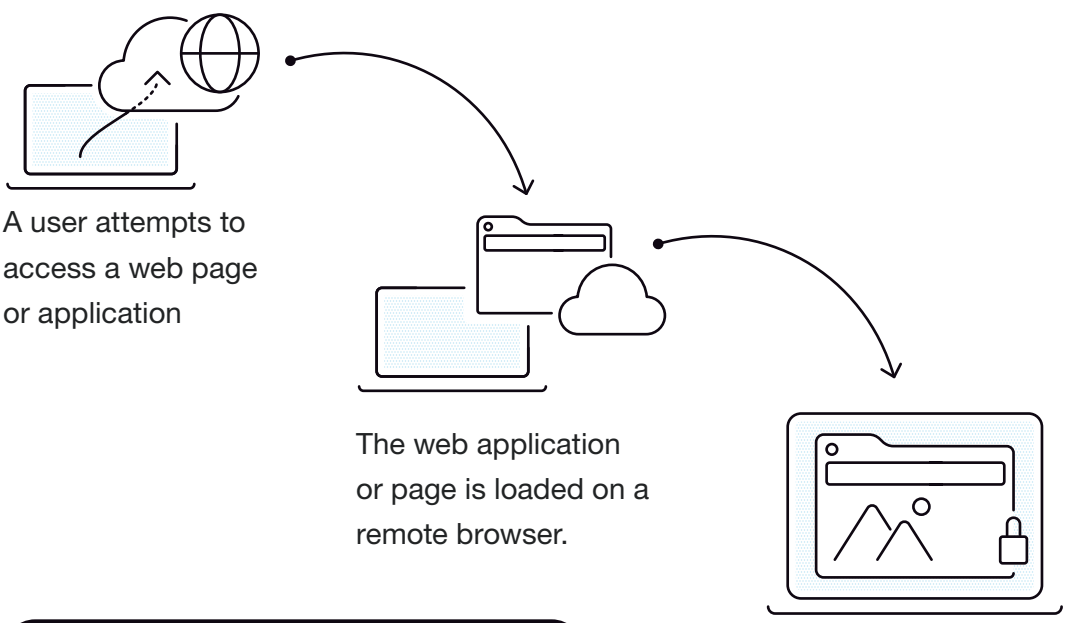
1/ Local Isolation

This is the traditional isolation method. It includes running a sandbox or virtual machine on the user's local computer to isolate its data from dangerous web browsing.

2/ Remote Isolation

Remote browser isolation provides an additional security layer against threats originating from web browsers. RBI helps you reduce the attack surface by separating user browsing activities from endpoint hardware.

Here is how the process typically works:



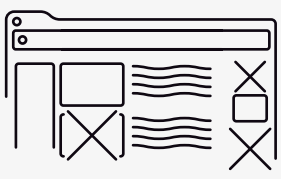
This process ensures that active content, including malware, is not downloaded—the endpoint device remains safe.

The remote browser serves the user with a rendering of the requested page.

The page loads as usual, but the remote browser delivers only a flattened version or the page or a video stream to the end-user device, not full HTML.

Remote browser isolation uses virtualization to create an isolated browser environment on a remote server. The user browses the Internet on the remote virtual environment. The remote server can be located in an organization's network or hosted in the cloud.

Two primary ways to isolate the user's local device from web content



DOM Mirroring

A technique that excludes certain types of web content that is considered dangerous, while displaying other types of web content in their original form—but the browser is not fully isolated.

Visual Streaming

The browser runs on the remote server and only its visual output is transmitted to the user's device. This works similarly to virtual desktop infrastructure (VDI) systems. This provides complete isolation between the remote browser and endpoints.

How Do Remote Browsers Work?

The user's endpoint device interacts with a remote browser isolation service, which manages a number of containerized or virtualized browser instances. The RBI service also facilitates communication between this browser and the Internet. Finally, the RBI service delivers rendered web content back to the endpoint device.

Two primary techniques used to stream content from cloud-based browsers to end-user devices

Pixel Pushing

captures pixel images of content rendered in the remote browser, and transmits them to the client's browser or a locally-deployed agent. This is similar to desktop sharing solutions. The inherent advantage of this approach is that it is very secure, since files or executable code never reaches the endpoint device.

DOM Reconstruction

attempts to clean web page code before sending it to the local endpoint, where it is rendered on the browser as usual. The remote browser removes potentially malicious code. This technique was introduced in response to the challenges of pixel pusing (detailed below), and provides a much faster user experience and high fidelity rendering of web pages.

Another element of RBI systems is a remote file viewer, that allows users to view files like Microsoft Office documents or PDFs, without having to download them. The remote browser may offer the option of downloading files to the user's local device in a controlled manner, after scanning and verifying the files are safe.

Challenges with Remote Browser Technology

1/ Pixel Pushing

High Cost

Encoding and transmitting video streams to multiple user endpoints is computationally intensive, and requires high bandwidth.

High Latency

Because of the need to render browser pages on a remote browser, create a video stream and push it to the user, typically over a public network, this technique involves high latency and creates a poor user experience compared to local browsing.

Mobile Support

The need for high bandwidth makes it difficult to support this technique with common mobile devices.

Low Resolution

Pixel pushing does not display well on high DPI displays, such as Apple Retina.

2/ DOM Reconstruction

Security Issues

Although DOM reconstruction aims to “clean” website code from malicious elements, it is not foolproof. There is a major risk that malicious code will not be identified or properly cleaned and will make its way to the user’s device.

Limited Fidelity

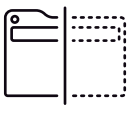
In the attempt to remove malicious elements, this technique often breaks web pages, especially if they are dynamically generated using JavaScript. Modern web users access a wide variety of complex web applications using their browsers, and many of these applications will not work or will present limited functionality.

Evaluating Remote Browser Solutions

Important considerations when evaluating remote browsers for your organization

Need for Local Agent

Check if the solution requires deployment of an agent or local proxy on user endpoints. This can make deployment and operations of the solution much more complex.



Rendering Engine

Check how content is rendered and delivered by the remote browser service, and whether it uses the pixel pushing or DOM reconstruction technique.

Support for Plugins

Check which browser plugins are supported, and whether the remote browser solution supports common extensions like PDF and Java.

Support for Web Applications

Check if the remote browser supports SaaS applications used by your users, such as Gmail and Office 365. In some cases, web applications may be blacklisted by the remote browser due to security concerns.

Copy and Paste

If your security policy allows users to copy and paste content to the local device, check if the remote browser solution supports this, and whether copy-paste is enabled only for text, or also for rich objects like images and documents.

Operating System Licensing

Check which operating system is used for browser containers or VMs. If it is Windows, identify if licensing is included in the service price or if you need to provide licenses for each remote browser.

Virtualization Model

Check if browsers run in full VMs or containers. VMs provide stronger isolation, but they require more resources to run and take longer to start. Containers offer faster startup and better server utilization.

OS Isolation: A More Secure Alternative to Browser Isolation

OS Isolation is a new security approach, where all data and applications are isolated within a VM on an end user device, with secure networking, web browsing, and application download policies.

Full OS isolation also includes document isolation, and isolating peripherals like USB and printer applications. Unlike browser isolation, which only isolates risks via web browsing, OS isolation offers a more robust security solution.

With OS isolation security solutions, the user has a completely isolated local OS that looks like another workspace on the user's device. Risky content is automatically launched in this isolated local OS. This enables users to be fully productive, including:

- Installing any desktop app
- Getting full local admin rights
- Safely viewing/editing risky documents
- Accessing any website/cloud service
- plugging in risky peripherals

Because of the level of isolation that OS isolation offers, there is no risk to the corporate network or to corporate data/apps.

All of these activities are done in an isolated virtual machine that provides the highest level of security against advanced OS-level threats.

How Does OS Isolation Address the Challenges of Browser Isolation?

1/ Improved Security

Browser isolation solutions provide security from endpoints attacks that enter while the user is browsing the web. While Web browsing is one of the most common forms of endpoint attacks, malware, spyware and other malicious threats can also reach the endpoints via downloaded applications, especially communication applications like Zoom or Slack. They can also access the user's endpoint via untrusted documents that have been downloaded, or even via external devices such as printers or USBs. OS solution provides an all encompassing security solution, isolating all forms of risk to the user's endpoint device.

2/ Improved User Experience

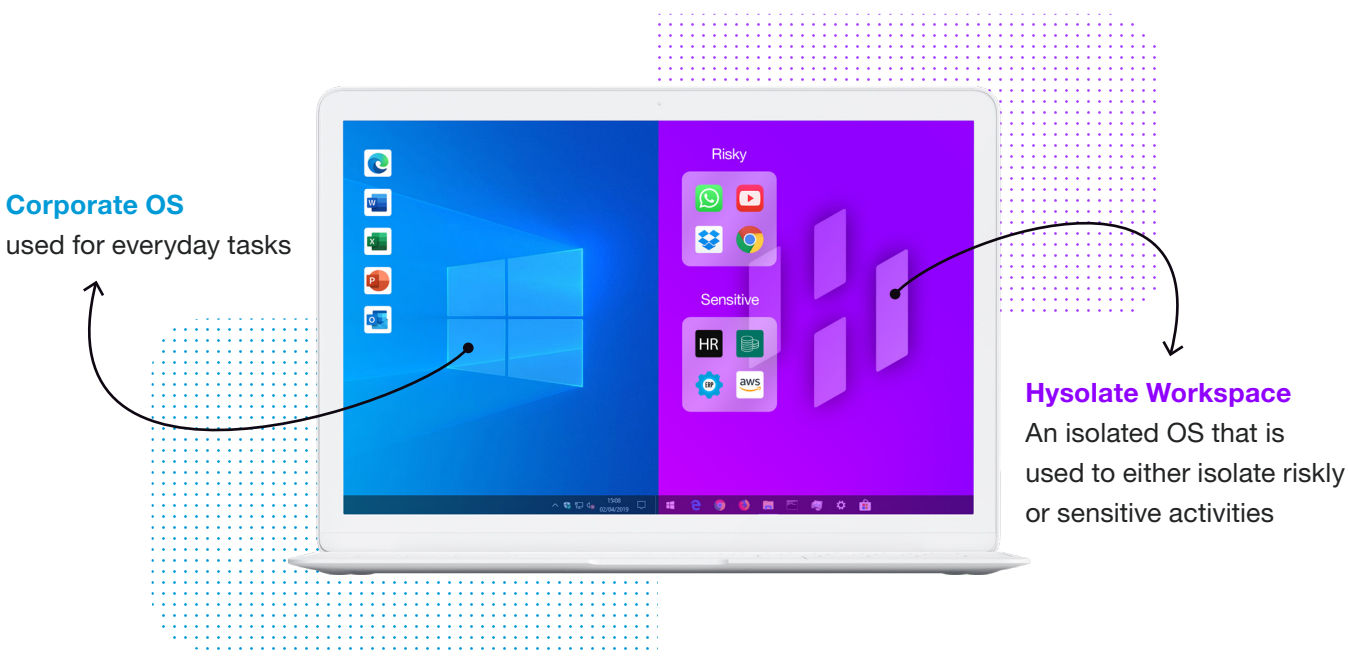
Browser isolation involves high latency and creates a poor user experience compared to local browsing, because of the need to render browser pages on a remote browser, create a video stream and push it to the user, typically over a public network. OS isolation on the other hand works through a Virtual Machine on a user's endpoint, so it creates a better user experience, with minimal latency issues.

Hysolate

Your Fully Managed OS Isolation Solution

Hysolate isolates all user activity within a local VM

acting either as a sandbox for risky activities and untrusted applications, or alternatively, providing an isolated environment for sensitive corporate access and data, with full separation on the network level from the host OS.



While browser isolation solutions cannot provide full endpoint security outside of web browser threats, Hysolate provides full OS isolation including isolating:

- Websites
- Applications
- Documents
- Peripheral devices like printers and USBs

Admins can deploy and scale Hysolate out to whole teams in minutes, each with their own policies, and users can begin to use the solution almost immediately with a native user experience, reducing IT overhead and resources.

*

Sitting on user endpoints means that Hysolate provides a smooth user experience, even when isolating heavier applications like popular communication solutions.

Unlike standard virtual machines, Hysolate is fully managed from the cloud, with granular management policies. Admins can control policies including a fully fledged IP/domain-name firewall, fine grained clipboard (copy/paste/file transfer) policy and USB / Webcam / Printer security controls.

Hysolate is Enterprise-ready

with features including SSO compatibility and SIEM integration (including Splunk and QRader) Workspace access can be protected by a password or a PIN, and it can be wiped remotely in case of need.

Want to learn more about Hysolate's OS Isolation solution? Request a demo or try Hysolate Free for yourself.

Request a Demo **Download Hysolate Free**