B

Browser Isolation: How RBI Works, What It Stops, and Its Limitations

Browser isolation runs website code away from the endpoint and streams a safer representation. Learn RBI rendering methods, deployment choices, limitations, and evaluation criteria.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
6 min

CONTENTS

In this guide

01 How remote browser isolation works
02 Isolation and rendering methods
03 Threats browser isolation can reduce
04 What browser isolation does not solve
05 Common deployment models
06 Browser isolation vs related controls
07 Browser isolation evaluation checklist
08 Operational limitations
09 Frequently asked questions
10 References

Browser isolation runs website code in an environment separated from the user's endpoint and delivers only a safer representation of the session. Remote Browser Isolation (RBI) executes the browser in a cloud or data-center container; local isolation uses a sandbox or virtualized environment on the device.

The goal is to keep untrusted HTML, JavaScript, advertisements, and exploit attempts away from the local operating system. Browser isolation reduces web-execution risk, but it does not make every page trustworthy or prevent users from submitting credentials and sensitive data to a phishing site.

01 How remote browser isolation works

1. The organization routes selected web sessions to an isolation service.
2. A disposable remote browser loads and executes the website.
3. The service applies browsing, download, upload, clipboard, and data-loss policies.
4. The user receives a visual or reconstructed representation of the page.
5. The remote session and its temporary storage are destroyed or reset according to policy.

The endpoint still needs a browser or client to display and interact with the session, but the original website code should not execute directly on that endpoint.

02 Isolation and rendering methods

Method	How content reaches the user	Tradeoff
Pixel streaming	The remote browser sends visual frames and receives keyboard or pointer input	Strong separation, but can add bandwidth, latency, and accessibility challenges
DOM or content reconstruction	The service rewrites and sends a safer version of page structure	More native interaction, but reconstruction must safely handle active content

Method	How content reaches the user	Tradeoff
Network vector rendering	Rendering instructions represent the page without forwarding original code	Can balance experience and isolation, with implementation-specific limitations
Local container or sandbox	The site runs in an isolated environment on the endpoint	Lower network latency, but security depends on local isolation and patching

03 Threats browser isolation can reduce

- Browser exploits and drive-by downloads that rely on code executing locally.
- Malvertising and compromised-site scripts.
- Some web-based fileless attacks and cryptocurrency-mining scripts.
- Exposure from newly registered, uncategorized, or risky websites.
- Direct access to local cookies, files, and operating-system resources by page code.
- Risk when analysts or support teams must view suspicious links.

The isolation service itself must be patched, segmented, monitored, and designed to prevent one session from reaching another. Moving execution to the cloud changes the trust boundary; it does not eliminate one.

04 What browser isolation does not solve

Risk	Why it remains	Additional control
Credential phishing	The user can type a real password into a fake page rendered correctly	Phishing-resistant MFA, domain verification, and password-manager origin checks
Sensitive-data disclosure	Users may paste, upload, or type data into the session	DLP policies, upload and clipboard controls, and user training

Risk	Why it remains	Additional control
Malicious downloads	A downloaded file may leave the isolated session and run locally	Block, sanitize, scan, or open downloads in a separate sandbox
Account takeover	Stolen sessions or identity-provider compromise are outside page-code isolation	Identity monitoring, conditional access, and session controls
Unsafe allowed applications	Users may grant OAuth consent or perform harmful business actions	SaaS governance and application approval
Non-browser traffic	Email clients, native apps, APIs, and remote tools may bypass RBI	Endpoint, email, network, and application controls

05 Common deployment models

- **Isolate all web traffic:** maximizes coverage but increases cost and the effect of compatibility problems.
- **Risk-based isolation:** isolates unknown, uncategorized, personal, or risky sites while trusted destinations load normally.
- **Read-only isolation:** blocks typing, uploads, clipboard, downloads, or printing on high-risk pages.
- **Email-link isolation:** opens links from messages in RBI, often with additional download restrictions.
- **Privileged-use isolation:** protects administrators who access vendor portals, management interfaces, or untrusted research sites.

Selective isolation usually requires reliable traffic steering and clear fallback behavior. If an agent, proxy, VPN, PAC file, DNS policy, or browser extension fails, define whether access is blocked or silently bypasses isolation.

06 Browser isolation vs related controls

Control	Primary function
Secure Web Gateway	Filters and inspects requests according to URL, content, identity, and policy
Browser isolation	Moves or contains active page execution away from the endpoint
File sandbox	Executes a downloaded object to observe its behavior
Content Disarm and Reconstruction	Removes active elements and rebuilds documents
VDI	Provides a full remote desktop or application environment, not only isolated web rendering
Enterprise browser	Applies management and security policy inside a controlled browser; it may also integrate isolation

Organizations often combine a gateway for policy, RBI for risky sessions, scanning or CDR for files, and endpoint protection for anything that reaches the device.

07 Browser isolation evaluation checklist

1. **Define the use case.** Identify users, sites, threats, data, and workflows that justify isolation.
2. **Test real applications.** Include WebRTC, video, printing, accessibility, password managers, file transfers, browser extensions, and certificate-based authentication.
3. **Measure user experience.** Check latency, bandwidth, visual quality, keyboard layouts, mobile use, and behavior on poor connections.
4. **Validate policy controls.** Test uploads, downloads, clipboard, copy and paste, printing, screen capture, and data classification.

5. **Inspect the bypass path.** Confirm what happens when the service, agent, tunnel, or policy engine is unavailable.
6. **Review privacy and residency.** The provider may process page content, typed data, credentials, uploads, and browsing metadata.
7. **Verify isolation claims.** Understand tenant separation, session disposal, logging, patching, and incident notification.
8. **Plan operations.** Assign owners for policy changes, exceptions, troubleshooting, alert review, and vendor outages.

08 Operational limitations

Some applications depend on direct device access, browser extensions, client certificates, hardware acceleration, WebUSB, WebBluetooth, local network discovery, or specialized media. Isolation can limit or alter these features. Current vendor documentation should be tested rather than assuming every site behaves like an ordinary page.

Pixel streaming may increase bandwidth and affect text clarity. Reconstruction may break complex sites. Remote downloads and uploads can complicate DLP and malware inspection. Exceptions should be narrow, documented, time-limited, and monitored.

09 Frequently asked questions

Does RBI stop phishing?

It can prevent page code from compromising the endpoint, but a user can still submit credentials to a convincing fake site. Identity controls remain necessary.

Can users download files through browser isolation?

That depends on policy. Downloads may be blocked, scanned, sanitized, or transferred to the endpoint. Each path should be tested and monitored.

Is Incognito mode browser isolation?

No. Private browsing mainly limits local history and cookie persistence. Website code still runs in the local browser.

Does RBI replace endpoint protection?

No. Files, native applications, email, removable media, credentials, and non-browser traffic still require protection.

10 References

- Palo Alto Networks: How Remote Browser Isolation works
- Cloudflare: Browser Isolation known limitations

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com