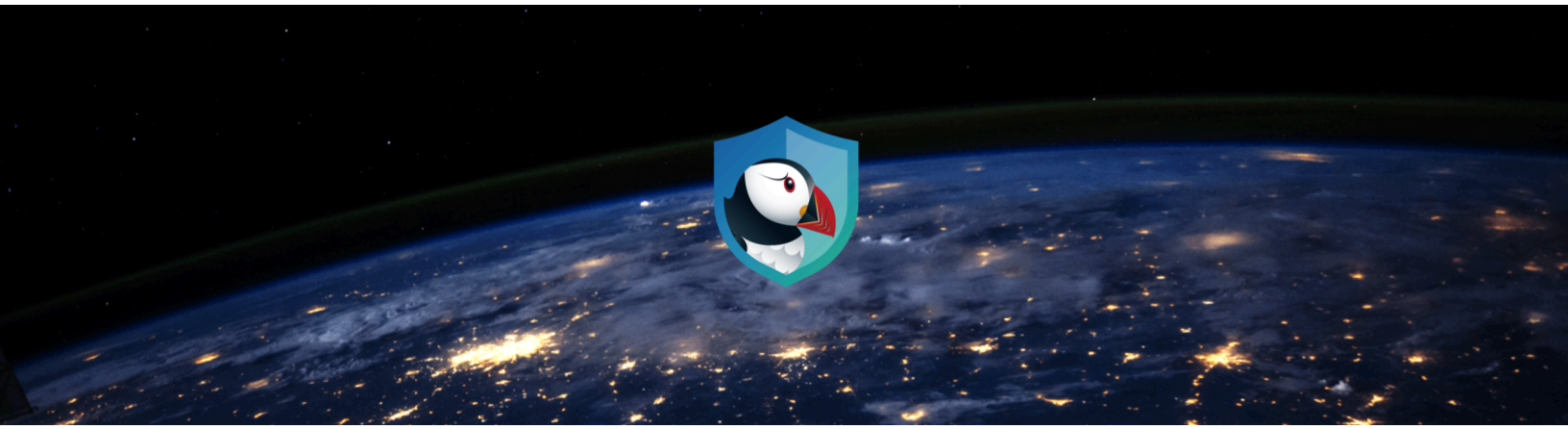
The End of Patch-and-Pray: Why AI-Driven Zero-Days Demand a New Endpoint Strategy

CloudMosa, Inc.
March 2026



The AI Acceleration of Software Exploitation

For decades, software security has relied on a fundamental assumption: vulnerabilities are discovered slowly enough that vendors can patch them before attackers exploit them at scale.

That assumption is weakening rapidly.

Recent research demonstrates that AI-assisted code analysis can discover vulnerabilities at rates previously impossible for human auditors. In early 2026, Anthropic reported that AI code agents identified hundreds of previously unknown vulnerabilities across widely deployed open-source projects. In targeted experiments with Mozilla, automated agents significantly increased the rate of vulnerability discovery in the Firefox codebase—one of the most heavily scrutinized software projects in the world.

The implications extend beyond any single organization or browser. AI dramatically reduces the cost of vulnerability discovery. Automated agents can analyze large codebases continuously, explore unusual execution paths, and iterate rapidly. As these tools improve, the global supply of exploitable software vulnerabilities will increase.

For defenders, the asymmetry remains unchanged: every vulnerability must be fixed; attackers need only one. Even organizations with aggressive patching practices operate on deployment cycles measured in days or weeks due to testing, compatibility validation, and staged rollouts.

If vulnerability discovery accelerates while patch deployment remains bounded by operational realities, the window of exposure inevitably expands.

This shift is not hypothetical. It is a structural change in the economics of software security.

The Isolation Imperative

Conventional endpoint defenses—EDR, signature-based detection, patching cadences, allow/deny policies—were designed for a world where zero-days were rare and expensive. That world is ending.

AI-accelerated vulnerability discovery does not just increase the number of known vulnerabilities. It compresses the timeline between discovery and exploitation. Threat actors with access to the same AI capabilities can weaponize vulnerabilities before patches exist, before signatures are written, and before threat intelligence is distributed.

No amount of faster patching, smarter heuristics, or deeper telemetry addresses this structural gap. These defenses all share a common assumption: that dangerous code executes on the endpoint, and the defender's job is to detect or prevent it in time.

There is one class of architecture that is immune to zero-day volume: **isolation**. If untrusted code never reaches the endpoint, it does not matter how many zero-days exist. The browser—the single largest attack surface in any enterprise—can be architecturally removed from the endpoint's threat model entirely.

VDI Was Right About Isolation, Wrong About Everything Else

Virtual Desktop Infrastructure has demonstrated that enterprises will invest in isolation. For years, VDI has been the default answer when organizations need to keep sensitive environments separated from untrusted content. The security model is sound: execute everything remotely, deliver only pixels to the endpoint.

However, VDI carries significant costs. Infrastructure is expensive to provision and maintain. Licensing is complex. The user experience—bounded by pixel streaming latency, compression artifacts, and input lag—consistently underperforms local applications. Moreover, VDI was designed to virtualize entire desktops, which is architecturally excessive when the threat is overwhelmingly concentrated in a single application: the web browser.

Over 90% of enterprise work now takes place inside a browser. The attack surface is the browser. The isolation target should be the browser—not the entire desktop.

What enterprises need is not a virtual desktop. They need **VDI-grade isolation applied specifically to the browser, with the usability of a native application**.

Puffin Secure Browser: Isolation That Feels Native

Puffin Secure Browser, developed by CloudMosa, delivers exactly this. Rather than streaming remote pixels or reconstructing sanitized DOM content, Puffin extends Chromium's own internal architecture across the network.

At its core is **RemoteMojo**, a network-extended implementation of Chromium's native inter-process communication framework. RemoteMojo relocates all untrusted execution—the renderer, JavaScript engine, network service, and storage—into disposable cloud-side containers. The client retains only what is needed for presentation: GPU compositing, local input handling, and display scheduling.

The result is a browser where no web code, no DOM, and no active scripts ever reach the endpoint—while the user experience remains indistinguishable from a locally installed browser.

This is not pixel streaming. Puffin transmits graphics commands and compositor state deltas, not compressed video frames. Scrolling, animations, and gestures respond instantly at local display cadence through a Distributed Compositor that predicts safe visual updates client-side and converges asynchronously with authoritative server state. Bandwidth consumption is a fraction of video-based alternatives. Full Chromium web compatibility is preserved because the engine is real Chromium—not a proxy or a sanitized reconstruction.

Security Properties

Puffin’s architecture provides several important security characteristics:

- **No Web Code on the Endpoint**
  HTML, JavaScript, and WebAssembly execute exclusively in remote environments.
- **Disposable Execution Environments**
  Browser workloads run in short-lived cloud containers that can be reset frequently.
- **Reduced Endpoint Attack Surface**
  The endpoint contains only trusted rendering components rather than full browser engines.
- **Centralized Policy Enforcement**
  Enterprise security policies—data loss prevention, URL filtering, access controls—are enforced in the cloud environment, not on untrusted local devices that may be compromised, misconfigured, or outside IT control.

- **Rapid Security Updates**
  Because execution occurs in the cloud environment, browser engine updates can be deployed centrally.

These properties significantly reduce the likelihood that browser vulnerabilities can compromise enterprise endpoints.

Operational Efficiency

Puffin applies isolation specifically to the browser rather than the entire desktop environment. This targeted architecture offers several operational advantages over traditional VDI and RBI deployments:

	VDI	**Pixel-Streaming RBI**	**Puffin Secure Browser**
Isolation Scope	Entire desktop	Browser	Browser
User experience	Remote desktop	Video stream	Native-like browser
Bandwidth	High *(continuous video)*	High *(continuous video)*	Low *(state deltas)*
Web compatibility	Full *(but heavy)*	Varies	Full *(real Chromium engine)*
Infrastructure cost	High	Moderate	Low

CloudMosa has refined this distributed-browser architecture since 2009, deploying it across desktop PCs, smartphones, and feature phones—serving over 200 million cumulative users. Since inception, Puffin Secure Browser has not experienced a publicly documented zero-day exploit resulting in endpoint compromise. The architecture tracks upstream Chromium closely—enabling rapid adoption of security patches without the merge debt that plagues deep browser forks.

For a deeper look at the technology behind Puffin Secure Browser, see our technical white papers:

Part I: Distributed Chromium Architecture for Remote Browser Isolation

Part II: Distributed Compositor Architecture for Responsive Remote Browsing

A Structural Shift in Endpoint Security

AI-driven vulnerability discovery is not a temporary phenomenon. It is a permanent acceleration of the threat landscape. The volume of exploitable zero-days will continue to grow, and the time advantage defenders once held will continue to shrink.

Enterprises face a choice: continue investing in detection-based defenses that assume a manageable vulnerability rate, or adopt an architecture that is structurally immune to zero-day volume.

Puffin Secure Browser offers VDI-grade endpoint protection with native browser usability—at a fraction of VDI's cost, complexity, and infrastructure burden. It is built on over 15 years of production-proven distributed browser technology, engineered for the threat environment that AI is now creating.

The era of patch-and-pray is ending. Isolation is the answer. Puffin is isolation done right.

To learn more about Puffin Secure Browser, contact CloudMosa, Inc. at https://www.cloudmosa.com