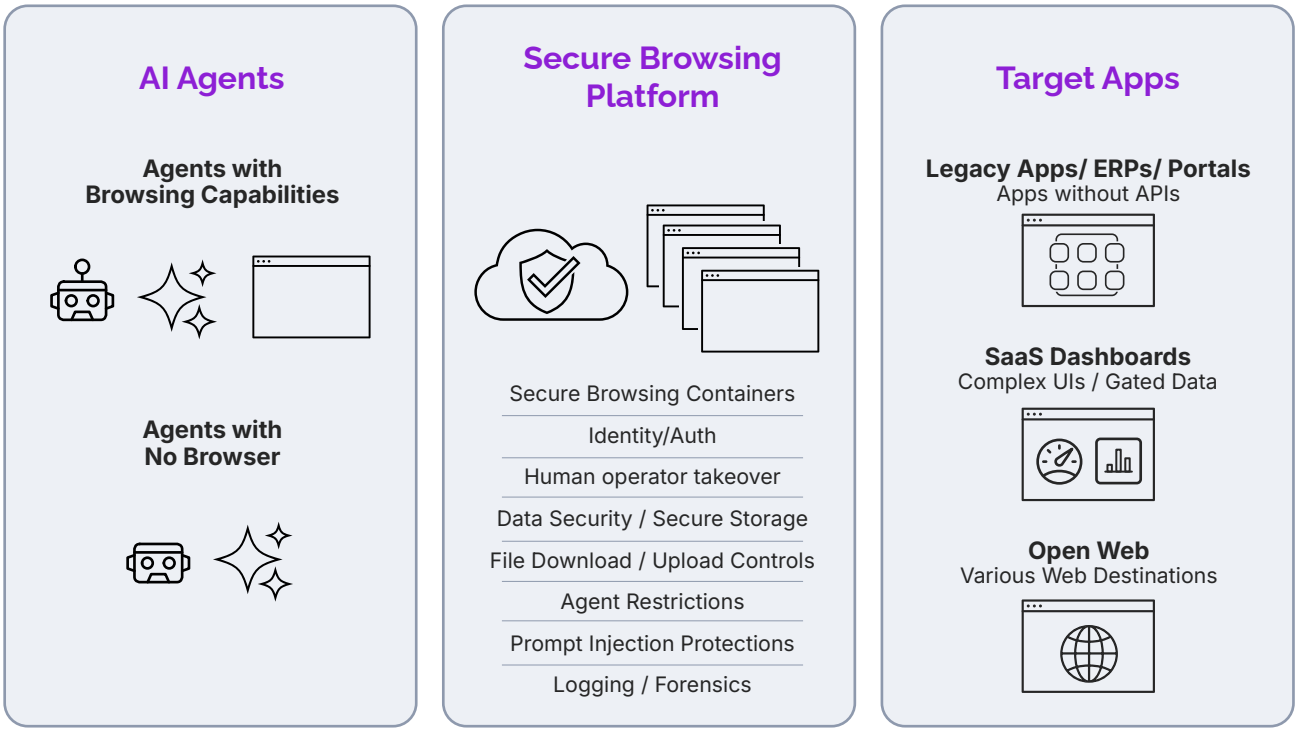
MENLO SECURITY

Menlo Agent Runtime Security (MARS)

WHITE PAPER

1. Overview

Menlo Agent Runtime Security (MARS) is a new solution that enables AI agents to autonomously browse to web portals and applications through the Menlo Secure Cloud-based browser security platform. The MARS solution provides integrated security and data controls that address the broad attack surface agents are exposed to in real-world environments, when browsing autonomously.



As organizations empower teams to build evermore sophisticated agents, these agents are increasingly browsing the web. This is driven by two key factors: the web is where the majority of the world's information resides, and it is the primary environment where employees conduct most of their work.

While some tasks can be managed by simply invoking APIs over HTTP, fully leveraging all applications and the web's rich, dynamic content requires a full-fledged browsing engine.

This shift is already underway. Agents are being embedded directly into browsers to access the web; tools like Claude's Cowork and OpenClaw's toolset are incorporating browser capabilities to enhance their effectiveness in creating autonomous AI assistants that automate complex, multi-step tasks.

Additionally, organizations are also evaluating options to integrate modern AI tooling with their existing web applications, which often lack robust and complete API support. One solution to this is enabling AI agents to autonomously browse, input data, and complete tasks within these web applications. But these agent interactions still need to be secured.

The shift toward agent browsing is also being driven by advances in web standards, notably the proposed Web Model Context Protocol (WebMCP). WebMCP aims to standardize how AI agents interact with web applications, enabling them to perform actions with increased speed, reliability, and precision.

2. Addressing Agentic Browsing Challenges

The MARS platform is intended to address the following challenges faced by organizations seeking to deploy LLM-powered autonomous browsing agents to perform tasks on the public and private web.

2.1 Managing / Maintaining Browser Automation Infrastructure

The challenges associated with building and maintaining reliable browser based AI automation solutions at scale - including the constant need for maintenance and updates - represent a significant overhead for many organizations. Additional capabilities such as threat protection and data security protections must also be considered when developing a robust, scalable Chrome-as-a-Service platform for autonomous browsing agents.

2.2 Autonomous Browsing Agents Blocked Due to Website Bot Detections and Other Protections

Previously the predominant use case for bots accessing website content were crawlers. As a result many websites and applications implemented defenses to prevent abuse of their services. This includes detecting bot-like browsing behaviors, leveraging IP reputation, Geofencing and more. Legitimate autonomous browsing agents performing tasks on behalf of users are impacted by these same defenses.

2.3 Lack of Oversight / Human-in-the-Loop

Another challenge faced by organizations when attempting to enable autonomous browsing agents is the ability to capture sessions for compliance and troubleshooting. Additionally Security and IT teams require a mechanism for observing and intervening live as agents perform tasks in the browsers. This could include entering MFA codes, solving CAPTCHAs and pausing or stopping active tasks.

2.4 Access Policy Controls for Autonomous Browsing Agents

While autonomous browsing agents often operate under a human user's identity, they necessitate granular policy controls. Unlike human users, who require broad access to the open web, agents should be restricted by 'Least Privilege' protocols—limiting their scope strictly to the destinations and actions required for a defined task.

2.5 Preventing Threats Introduced Via Autonomous Browsing Agents

Autonomous browsing agents are capable of extensive and large scale actions, such as filling out forms and downloading files. At the same time, prompt injection remains an unsolved problem: the agent may decide to treat any content it reads as instructions.

For example, an agent browsing content in OneDrive or Google Drive could ingest files that contain malicious prompts that would initiate an attempt to exfiltrate data where the agent would be instructed to perform risky actions on corporate sensitive data.

Similarly, an agent browsing sites that contain user editable content, such as Reddit or YouTube (both the videos themselves and comments on the videos), is liable to be tricked into leaking sensitive information or performing unsanctioned actions. This results in a huge potential attack surface with every webpage, embedded document, ad, and script potentially carrying malicious instructions.

Attack vectors such as unauthorized task execution, credential exfiltration, or domain allow-listing bypass can arise from web content rather than strictly from adversarially optimized inputs to the model.

2.6 Preventing accidental or malicious data exfiltration

Having access to a significant amount of private and sensitive data, agents can mistakenly overshare data with unauthorized users and applications. This can be driven by a naive interpretation of requested action or through an intentionally malicious prompt injection attack.

For example, an agent can be instructed to send a list of credit card numbers in its possession to an external location.

3. Sample Use Cases for Autonomous Agent Browsing

3.1 Securing usage of Claude Code, OpenClaw and agentic CLI

Securing use of Claude Code, OpenClaw or any agentic CLI that can execute commands and reach the web. Apply controls to prevent unrestricted internet access - permitting connections only to essential domains - and prevent inadvertent data exfiltration. Ensure a complete forensic trail is available for compliance and auditing purposes.

3.2 Complex Form Filling / Data Entry

Automating completion of web-based forms where input fields change based on previous answers.

3.3 Procurement

Navigating through multiple vendor portals to compare prices and complete purchases

3.4 Invoice & Document Retrieval

Logging into different web applications or SaaS portals to download and organize PDF invoices

3.5 Data Migration

Moving data between legacy enterprise systems that do not have APIs instead using web portal interfaces.

3.6 Research

Agents browse autonomously to perform various research tasks including: - collecting structured data from websites (prices, ratings), carrying out competitive research or prospecting for sales leads based on buying signals.

3.7 Web Scraping / Content Aggregation

Building content aggregators by scheduling agents to scrape multiple open web sources daily.

3.8 Monitoring

Tracking price changes and other values across multiple websites / applications - syncing that data directly to spreadsheets or databases.

3.9 Model Training

Crawling the web to build datasets for use in model training.

3.10 Security Investigations

SOC automation - Agents autonomously browse to malicious / suspicious sites to verify and collect information, download malicious payloads, download HAR files, understand redirect chains etc.

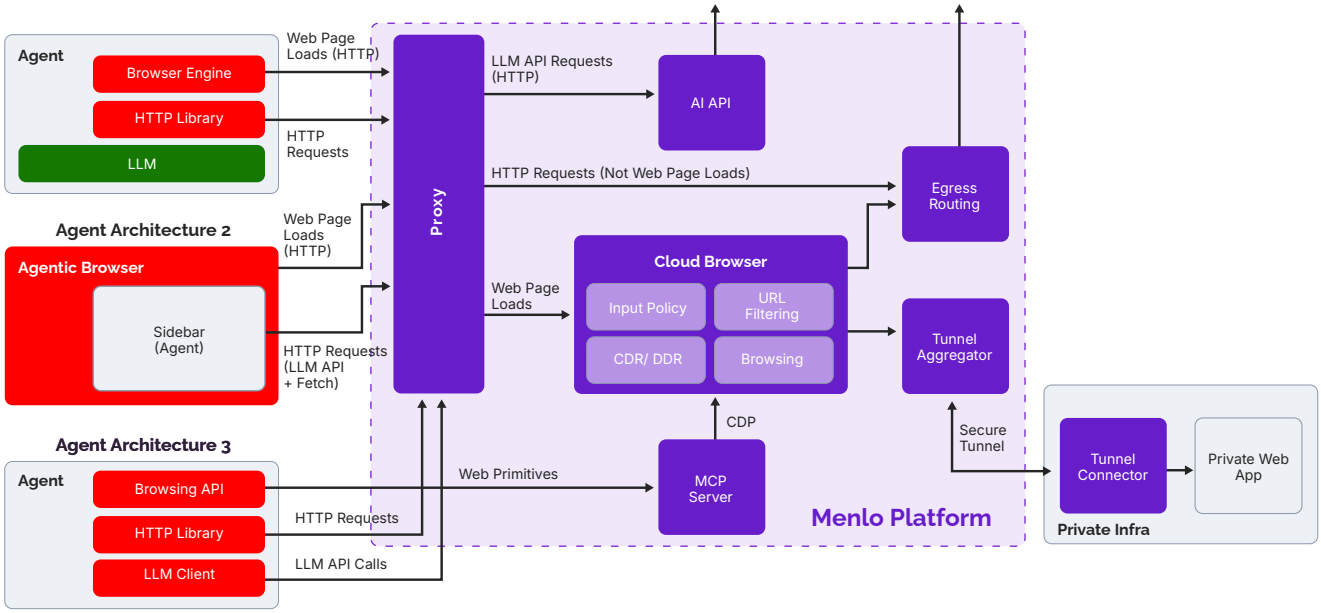
4. MARS Architecture

The MARS platform leverages the combination of Menlo's patented Isolation Core™ (including Adaptive Clientless Rendering (ACR) technology), Menlo's Web Deep Inspection, and Menlo's unique document processing capabilities. When an AI agent requests a webpage, the request is intercepted and routed to the Menlo Cloud. The Isolation Core spins up a disposable, virtualized browser container (the "cloud browser") that fetches the content and executes all active code (JavaScript, WebAssembly, etc.).

When an agent elects to directly invoke APIs over HTTP, the Menlo Proxy interposes on these API calls and applies destination based policies. Likewise, when the agent invokes a cloud LLM, the Menlo Proxy interposes on these interactions. In all cases, the Menlo platform is able to detect file transfers and analyze file content. The combination of multiple vantage points and rich semantic understanding of web applications and documents gives the Menlo platform an unparalleled ability to secure agentic workloads.

4.1 Channels

The MARS product will initially support three common channels used by agents to interact with the outside world.



4.1.1 HTTP Channel

This channel is used by agents to fetch content over HTTP. Examples include using command line tools such as cURL, embedded libraries such as Axios, or the sidebar leveraging HTTP fetch primitives exposed by the browser.

4.1.2 Browser Channel

This channel covers the following two scenarios where agents interact with a browser engine that can load and render web pages, including executing JavaScript, allowing interaction with dynamic web content.

[Dual Browser Engine mode] Agents with their own browsing capabilities. Examples include browser sidebars (sidebar interacts with the browser it is embedded in) and tools such as Claude Cowork + Claude Chrome extension (Claude interacts with the user's browser).

[Single Browser Engine mode] Agents without their own browsing capabilities. Here, the agent interacts directly with the Menlo platform via an MCP Server. The Menlo Browsing MCP server exposes APIs to create and manage cloud browsers.

For these browsers, it provides low-level access to the Chrome DevTools Protocol (CDP), enabling agents that want to manipulate web applications by directly manipulating the DOM (similar to Playwright/Puppeteer). It also provides a higher-level API that supports natural language processing, similar to Stagehand. This accommodates a broad spectrum of agentic use cases.

4.1.3 LLM Channel

Similar to the HTTP channel, this covers scenarios where the LLM used by the agent is external (e.g., a cloud service such as Google's Vertex AI). While the LLM may be running on the same device as the agent (e.g., Gemini Nano, embedded in Chrome and used by the sidebar), it is common for agents to use a well specified API that can be fully interposed on.

4.2 Mechanisms to Interpose/Apply Policy Controls

For each one of these channels, the methods for intercepting and interposing on agent-generated requests, enabling secure browsing for autonomous agents, include the following:

4.2.1 Cloud Proxy

Supported channels: HTTP, Browser (dual browser engine mode) and LLM.

This is applicable in scenarios where the customer has an existing agentic system that uses a browser as a tool and/or uses an HTTP library to fetch content. This agentic system can be configured to use a proxy. Menlo is configured as the proxy in this case.

4.2.2 MCP Server

Supported channels: Browser (single browser engine mode).

In scenarios where the customer has an existing agentic system that does NOT have access to a browser tool, the agent uses the MARS platform directly. The following modalities are exposed to agents to enable browsing autonomously through the Menlo platform:

- Chrome Devtools Protocol (CDP)
- MCP Server

Both customer hosted and Menlo hosted MCP Server options will be supported. Customer deployed MCP Servers could support libraries such as Playwright and Stagehand to convert natural language to CDP instructions - providing browser automation capabilities.

4.2.3 API Proxy

Supported channels: LLM

This provides support for inspecting and manipulating the interactions between the agent and an LLM over the LLM channel.

5. MARS Capabilities

5.1 Agent Identity / Authentication / Authorization

The MARS solution performs API token-based authentication of autonomous browsing agents that access websites and applications through the secure cloud browser platform. This includes agents using real user accounts vs. separate machine identities / designated service accounts, enabling least privilege access policies to be applied at runtime. The Menlo platform provides comprehensive support for modern Identity Providers (IdPs) as well as the option to use built-in user accounts.

To access restricted content or perform protected actions, many websites and applications require user authentication. The Secure Agent Browser (MARS) solution supports multiple authentication methods within autonomous browsing sessions, including securely retrieving credentials from password vaults to log into websites and the ability to inject ephemeral tokens.

The Secure Agent Browser (MARS) solution persists authentication state, store cookies, session tokens, and local storage minimizing the need for reauthentication, reducing login failures and improving session continuity.

Phishing-Resistant “Blind” Authentication

A significant security advantage of the MARS architecture is its handling of credentials. Hardcoding high-privilege passwords into agent scripts is a major vulnerability, as these can be leaked via prompt injection.

Credentials are only injected at the correct, verified URL and are never made visible to the agent itself. This “blind authentication” ensures that the agent never possesses the secret it is using to log in, preventing credential theft even if the agent’s logic is compromised.

5.2 Threat Protections

Browsing AI agents integrate web navigation, autonomous decision-making, and external tool usage. Consequently, their threat landscape spans beyond adversarial manipulations of a model's parameters, encompassing vulnerabilities in prompt handling, user-supplied goals, and interactions with potentially malicious web resources.

Attack vectors such as unauthorized task execution, credential exfiltration, or domain whitelisting bypass can arise from web content rather than strictly from adversarially optimized inputs to the model.

5.2.1 Destination Controls (Domain / URL / Category-based)

Ensures Agents can only connect to specified domains and applications to perform defined tasks - preventing them from accessing potentially unsafe sites. Controls can be applied based on URL category, individual domain or url.

Additionally, real-time intent-based analysis of browser sessions (including screenshots and DOM content) provides additional protections against malicious and fraudulent sites on the open web.

5.2.2 Multi-Stage Prompt Injection Defense

MARS provides multiple levels of protection against prompt injection. These include:

Invisible Content Removal: The system automatically identifies and strips content in both web pages and documents that is invisible to human users but readable by models—such as zero-font text or hidden layers. In doing so, MARS significantly reduces the attack surface for Indirect Prompt Injection and increases the effectiveness of visual inspection of the content by a human operator to detect malicious content.

Shadow-Model Scanning: MARS employs lightweight "Judge" models—such as Llama-Prompt-Guard-86M—to scan chunks of the web page for adversarial intent in real-time. If a segment is flagged, it is blocked or sanitized before MARS releases the content to the agent.

In the case of Agents with their own browsing capabilities configured to use the Cloud Proxy, MARS pushes the page content (including stylesheets and images) to the client browser, meaning redaction happens before the content is pushed. For Agents without their own browsing capabilities (CDP), MARS redacts the content that is being pulled from the page by the agent.

Document Sanitization (CDR): All files accessed by the agent are routed through a Menlo-powered Content Disarm and Reconstruction (CDR) pipeline. This pipeline employs both Invisible Content Removal and Shadow-Model Scanning to ensure documents retrieved via either the HTTP or the Browser channel are sanitized before they reach the agent. It also applies to content sent to the LLM on the LLM channel.

5.2.3 File Download Protections

Comprehensive protections from potentially weaponized files downloaded by autonomous agents browsing the open web, or accessing legacy apps and other portals. This protects both the agent itself from malware (including zero-day attacks) as well as environments where the agent might be forwarding files to.

Protections are multi-layered and include:

- File sanitization (CDR positive selection technology)
- Static analysis (AV engines, File hash checks)
- Dynamic analysis / Sandboxing
- File type controls

Coverage for all major file types, including password protected files and archives which can be extracted and scanned automatically. There are several potential approaches for enabling passwords to be inputted when an agent encounters a protected file. These include integration with credential vaults, prompting a human to intervene or via an API.

5.2.4 Compromised Agent Protections

Considers scenarios where an agent has been compromised and is under an attacker's control, or otherwise exhibiting anomalous and abnormal behaviors.

Provides an oversight system, enabling Admins / Users / App owners to step in to pause or throttle the agent. A restricted "safe mode" can permit only a minimal set of read-only actions; any attempt at a 'high-risk' operation prompts a failure / request for human review.

5.3 Data Security Protections

Addresses risks of data exfiltration from AI agents, browsing autonomously, through integrated Data Security protections. Ensures autonomous agents only access data appropriate to their role and the task performed. Also ensures agents can't share data with unauthorized users and agents.

5.3.1 Data Detection

Agents that have access to private and sensitive data are prone to prompt injection luring the agent to exfiltrate data.

Menlo MARS detects private and sensitive data accessed by an agent and provides visibility to agent actions on data such as downloading/uploading sensitive content, sending private data through email or attachments. This could include insights into data access in cloud storage.

Menlo MARS Identifies signs of data exfiltration by agents browsing autonomously - analyzing behavior, access patterns, and data sensitivity, including unusual downloads, suspicious file movements or data exfiltration attempts.

5.3.2 DLP controls & Dynamic Data Masking

Files

Applies real-time dynamic protections to sensitive data contained in files by allowing, blocking and masking data in motion based on policies and context. This takes effect in real-time when files are uploaded / downloaded. Granular policy controls allow enforcing rules based on agent identity, target application, data direction, context and more.

Web Content

Menlo MARS applies dynamic protections to sensitive data at the point autonomous agents interact with SaaS and other web applications. operating at the presentation layer of the browser— dynamic data masking can mask sensitive data on the screen or in transit before it is inadvertently viewed, copied, or uploaded by the agents to unauthorized locations.

In addition, Menlo MARS can restrict use of copy-paste functions based on policy and context.

5.3.3 Secure File Storage for Agents

Enable autonomous browsing agents to download files only to designated secure storage provided by Menlo - with the option to also define corporate sanctioned collaboration platforms (e.g. Box, OneDrive, Google Drive etc.).

Can also extend to file uploads - where agents are only permitted access to designated file storage locations / repositories, when performing tasks on the open web.

5.4 Session Visibility and Control

5.4.1 Human Operator Takeover

In some cases it will be necessary for a human user to intervene in real-time should an autonomous agent become ‘stuck’ when attempting to perform tasks in the browser. This could include entering credentials, solving CAPTCHAs or completely shutting down the agent e.g. ‘Panic button’.

5.4.2 Logging & Analytics

Provides detailed tracking of agent activity, capturing event logs for agent web access and related classification metadata for resources accessed. The platform provides tools for visibility, analytics, and reporting of agent activity. The activity logs can also be ingested into SIEM tools for correlation and compliance requirements.

5.4.3 Session Recording

Provides the ability to automatically record agent browser sessions, enabling Admins and other teams to inspect the actions performed, review network requests, and debug issues page by page.

Thanks to the underlying cloud browser platform it is possible to capture a complete version of the browser session, including mouse clicks, text inputs, DOM interactions, and network packet capture. Each step / action performed by the autonomous browsing agent can be logged for SOC2 and other compliance frameworks.

5.5 Additional Policy Controls

Task-specific Policy Enforcement

Where applicable, assessment of agent actions against an established baseline can be performed. This could include ensuring agents have a valid 'Task ID' and are performing actions in accordance with a "Task-specific Policy".

Where an agent violates a task-specific policy - such as attempting to access an unauthorized server, exceeding the character count within a session, or performing an unauthorized action e.g. clicking a "Delete" button - a range of enforcement actions can be applied.

The Menlo isolation platform provides clear advantages in this regard. In the previous example it can intercept the call at the surrogate level before the mouse event ever reaches the destination app / portal.

5.6 Private Application Access for Agents

The MARS product will enable agents to access private enterprise applications without requiring a VPN client, which is often impractical to install in lightweight, ephemeral containers.

Leveraging the existing prepend capability that prefixes a specific gateway string (e.g., https://safe.menlosecurity.com/) to the requested URL, forces the request to be routed through and executed within the Menlo Isolation Core. This approach avoids many of the complexities typically associated with accessing private destinations (IP access, DNS, Certificate).

6 Summary

Menlo Agent Runtime Security (MARS) enables autonomous AI agents to safely access web applications and the open web. It addresses the API gap in dynamic or legacy systems - providing an alternative way to access these applications, securely through the browser and unlock the data stored within them.

MARS addresses key deployment challenges for enabling LLM-powered agents to browse autonomously: scaling infrastructure, defeating bot detection, and providing essential security, policy, and oversight controls.

The MARS architecture uses Menlo's patented Isolation Core™ and Adaptive Clientless Rendering (ACR). The Isolation Core runs web code in virtualized, disposable remote browsers, isolating agents from threats. MARS supports various interaction channels (HTTP, browser APIs, LLM services) through proxy-based and MCP server deployments.

MARS provides multi-layered security against web threats and data exfiltration. Features include destination controls, a Multi-Stage Prompt Injection Defense (content removal, shadow-model scanning, CDR), and file download protections. Data security controls include integrated DLP, dynamic data masking, and secure file storage.

For compliance and safety, the MARS solution will provide session recording and a "Human operator takeover" option.

Menlo is actively seeking to speak with customers who are currently (or considering) deploying autonomous browsing agents to better understand their specific use cases, security requirements and deployment types.