INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-05-06 09:04 UTC

Microsoft Edge Stores Cleartext Passwords in Memory Regardless of Session Activity

SECURITY ANALYSIS | **HIGH** | CVSS 7.5

SCC Item ID	SCC-STY-2026-0111
Type	Security Analysis
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Microsoft Edge (all versions retaining passwords in memory; related patch baseline: < 147.0.3912.60 per Tenable plugin 305979)
Published	2026-05-05T10:37:01
Discovery Source	Rss

Executive Summary

Microsoft has acknowledged that Edge retains all browser-stored passwords in cleartext within process memory while the browser is open, and has characterized this as intentional behavior rather than a security defect requiring remediation. Any attacker or malicious process with local memory read access can extract the full credential store without user interaction, a condition that post-exploitation tooling routinely satisfies. For enterprises that mandate Edge and store domain, SaaS, or privileged account credentials in the browser, this represents a persistent, unpatched credential exposure with no vendor-committed fix on the horizon as of this writing.

Technical Analysis

Security researcher @L1v1ng0ffTh3L4N disclosed that Microsoft Edge loads its entire password store into process memory in cleartext upon launch and retains it there regardless of whether those credentials are ever accessed during the session. The attack path is straightforward: an adversary with local memory read capability, achieved through any number of post-exploitation primitives, can dump Edge process memory and recover every stored credential in plaintext. No user interaction beyond Edge being open is required.

The applicable MITRE ATT&CK techniques map directly to the attack chain. Credential access via T1555.003 (Credentials from Password Stores: Credentials from Web Browsers) is the primary objective. T1003 (OS Credential Dumping) describes the memory access method. T1059 (Command and Scripting Interpreter) covers the execution layer attackers commonly use to invoke memory dumping tools. T1552.001 (Unsecured Credentials: Credentials in Files) reflects the downstream exposure once credentials are harvested.

Three CWEs apply simultaneously: CWE-316 (Cleartext Storage of Sensitive Information in Memory), CWE-312 (Cleartext Storage of Sensitive Information), and CWE-522 (Insufficiently Protected Credentials). The CVSS 7.5 High rating reflects local attack vector with low complexity, a realistic assessment given that memory scraping utilities are standard in commercial red team toolkits (e.g., Mimikatz, Metasploit, Cobalt Strike) and freely available in offensive frameworks.

Microsoft's 'by design' characterization is the most consequential element of this story. It means no CVE will be assigned, no patch is forthcoming through normal vulnerability channels, and enterprise defenders cannot rely on vendor remediation. The Tenable Nessus plugin 305979 tracks a related Edge security update baseline (< 147.0.3912.60), but that update addresses separate CVEs and does not remediate the cleartext memory retention behavior itself.

The enterprise risk profile is acute. Organizations that mandate Edge as their standard browser, particularly those where employees store domain credentials, privileged account passwords, or SaaS tokens in the browser password manager, face an unmitigated exposure on every endpoint where Edge runs. Post-compromise scenarios where an attacker achieves local code execution, a threshold already met by most commodity malware, immediately yield the full credential store. This can accelerate lateral movement, privilege escalation, and SaaS account takeover in ways that endpoint detection tools may not flag, since memory access by a process is not inherently anomalous.

Action Checklist

1. Step 1: Assess exposure, inventory all endpoints and user populations where Microsoft Edge is deployed, particularly in environments where Edge is the mandated or default browser and users store credentials in the built-in password manager.
2. Step 2: Review controls, evaluate whether EDR solutions deployed in your environment have behavioral rules capable of detecting memory scraping against Edge processes (e.g., unexpected cross-process memory reads targeting msedge.exe); verify whether a policy exists that prohibits storing privileged and domain credentials in browser password managers, and if no such policy exists, create one and audit current usage.
3. Step 3: Update threat model, add browser memory credential harvesting via Edge as an explicit post-exploitation technique in your threat register; cross-reference T1555.003 and T1003 detections in your SIEM and EDR rule sets.
4. Step 4: Communicate findings, brief leadership that this is a vendor-acknowledged, unpatched behavior with no remediation commitment; frame the risk in terms of credential exposure scope (domain accounts, SaaS tokens, privileged passwords) and the absence of a patch timeline.
5. Step 5: Monitor developments, track Microsoft's Edge security release notes (learn.microsoft.com/en-us/deployedge/microsoft-edge-relnotes-security) for any future change in vendor position; monitor for CVE assignment and any updates from Tenable plugin 305979 that may indicate a behavioral fix.

IR / Forensic Enrichment

Triage Priority	URGENT

Escalation Criteria	Escalate immediately to CISO and legal/privacy counsel if Sysmon Event ID 10 or EDR telemetry confirms any unauthorized process opened a memory-read handle to msedge.exe, or if the Login Data review reveals that domain administrator, PAM-managed, or regulated-data-system credentials (triggering breach notification obligations under HIPAA, PCI DSS, or applicable state privacy law) are stored in the Edge password manager on any endpoint.
Recovery Notes	If credential harvesting from Edge memory is confirmed or suspected, treat all credentials stored in the Edge password manager on affected endpoints as compromised: initiate forced password resets for domain accounts, revoke and reissue SaaS tokens, and rotate any privileged credentials via PAM. Disable the Edge built-in password manager enterprise-wide via GPO (`PasswordManagerEnabled = disabled` under `HKLM\SOFTWARE\Policies\Microsoft\Edge`) and validate the policy applied using `Get-ItemProperty 'HKLM:\SOFTWARE\Policies\Microsoft\Edge'` before returning endpoints to production. Monitor for 30 days post-rotation for authentication anomalies — specifically, Windows Security Event ID 4625 (failed logon) and 4648 (explicit credential use) patterns consistent with credential reuse from a harvested Edge credential store.
Forensic Artifacts	Edge Login Data SQLite database at %LOCALAPPDATA%\Microsoft\Edge\User Data\Default\Login Data — contains the full list of stored credentials (origin URL, username, encrypted password blob); presence of domain UPNs or privileged account usernames directly scopes the blast radius of a memory scraping event | Sysmon Event ID 10 (ProcessAccess) in Microsoft-Windows-Sysmon/Operational — records every process that opened a handle to msedge.exe with memory-read-capable access rights (GrantedAccess 0x1010, 0x1F0FFF, or 0x1F3FFF); the SourceImage field identifies the scraping tool or malicious process | Windows Security Event ID 4688 (Process Creation) in the Security event log — captures execution of known Edge credential dumping utilities (SharpChrome.exe, HackBrowserData.exe, BrowserGhost.exe) or any process spawning from msedge.exe that is inconsistent with normal browser child process behavior | Edge Local State file at %LOCALAPPDATA%\Microsoft\Edge\User Data\Local State — JSON file containing the encryption key used to protect the Login Data password blobs; if this file has been accessed or exfiltrated alongside Login Data, full offline decryption of stored credentials is possible | Windows Prefetch files at %SystemRoot%\Prefetch\ — entries for known credential dumping tools (e.g., HACKBROWSERDATA-*.pf, SHARPCHROME-*.pf) provide evidence of execution even if the binary was subsequently deleted, and include a timestamp of first and last execution relative to the Edge memory exposure window

Per-Action IR Details

Step 1: Assess exposure — inventory all endpoints and user populations where Microsoft Edge is deployed, particularly in environments where Edge is the mandated or default browser and users store credentials in the built-in password manager.

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: Establishing IR capability and understanding asset exposure before an incident occurs

Controls: NIST IR-4 (Incident Handling) — preparation sub-phase requires knowing which assets and credential stores are at risk, NIST RA-2 (Security Categorization) — categorize systems where Edge password manager use creates elevated credential exposure, CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) — identify all endpoints running Microsoft Edge as default or mandated browser, CIS 5.1 (Establish and Maintain an Inventory of Accounts) — correlate Edge deployments with accounts whose credentials may be stored, especially privileged and domain accounts

Compensating: Run the following PowerShell one-liner across endpoints via WinRM or a GPO startup script to identify Edge installations and whether the built-in password manager is enabled: `Get-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Edge\Extensions' -ErrorAction SilentlyContinue; (Get-Process msedge -ErrorAction SilentlyContinue).Count`. For password manager policy state, query: `Get-ItemProperty 'HKLM:\SOFTWARE\Policies\Microsoft\Edge' | Select-Object PasswordManagerEnabled`. Collect output to a CSV for triage. Osquery can also be used: `SELECT * FROM registry WHERE path LIKE 'HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Edge\%';`

Evidence: Before scoping, capture a baseline snapshot of Edge policy registry keys at `HKLM\SOFTWARE\Policies\Microsoft\Edge` and `HKCU\SOFTWARE\Policies\Microsoft\Edge` on sampled endpoints. Document whether `PasswordManagerEnabled` is explicitly set to 0 (disabled) or absent/enabled. Also collect the Edge Local State file at `%LOCALAPPDATA%\Microsoft\Edge\User Data\Local State` and the Login Data SQLite database at `%LOCALAPPDATA%\Microsoft\Edge\User Data\Default\Login Data` — the presence of entries in the `logins` table confirms active password manager usage and scopes credential exposure before any exploitation occurs.

Step 2: Review controls — evaluate whether EDR solutions deployed in your environment have behavioral rules capable of detecting memory scraping against Edge processes (e.g., unexpected cross-process memory reads targeting msedge.exe); also verify whether privileged and domain credentials are being stored in browser password managers against policy.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: Identifying indicators of attack and validating detection coverage for a known threat mechanism

Controls: NIST SI-4 (System Monitoring) — verify monitoring capability specifically covers cross-process memory read operations targeting msedge.exe, NIST AU-2 (Event Logging) — confirm that process access events (Windows Event ID 4663 or Sysmon Event ID 10) are being logged for Edge process handles, NIST IR-5 (Incident Monitoring) — validate that existing detection rules would surface memory scraping tools such as Mimikatz browser modules or custom LSASS-style dumpers targeting Edge, CIS 8.2 (Collect Audit Logs) — ensure audit logs capture the specific process interaction events needed to detect OpenProcess calls against msedge.exe

Compensating: Deploy Sysmon with a configuration that includes Event ID 10 (ProcessAccess) filtering for `TargetImage` matching `*msedge.exe` and `GrantedAccess` values of `0x1010` or `0x1F0FFF` (full read/memory access). Example Sysmon rule: `msedge.exe0x1F0FFF`. Review Sysmon Event ID 10 logs in Windows Event Viewer under `Microsoft-Windows-Sysmon/Operational`. For policy compliance, use the osquery query: `SELECT * FROM logged_in_users;` combined with manual review of `%LOCALAPPDATA%\Microsoft\Edge\User Data\Default\Login Data` using SQLiteBrowser to confirm whether domain or privileged account credentials are stored.

Evidence: Capture Sysmon Event ID 10 logs from the `Microsoft-Windows-Sysmon/Operational` channel focusing on any process that has opened a handle to `msedge.exe` with memory-read-capable access rights. Also review Windows Security Event ID 4663 (Object Access) if object-level auditing is enabled on the Edge process. Export the Edge `Login Data` SQLite file from `%LOCALAPPDATA%\Microsoft\Edge\User Data\Default\Login Data` (read-only copy) and use `sqlite3 'Login Data' 'SELECT origin_url, username_value FROM logins;'` to document which credential categories (domain, SaaS, privileged) are stored — this establishes blast radius before any exploitation is confirmed.

Step 3: Update threat model — add browser memory credential harvesting via Edge as an explicit post-exploitation technique in your threat register; cross-reference T1555.003 and T1003 detections in your SIEM and EDR rule sets.

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: Updating IR capability and threat models to reflect newly acknowledged vendor behavior and associated post-exploitation techniques

Controls: NIST IR-8 (Incident Response Plan) — update the IR plan to explicitly include browser memory credential harvesting as a post-exploitation scenario requiring its own playbook entry, NIST RA-3 (Risk Assessment) — document this vendor-acknowledged behavior as a standing risk item with no patch timeline, triggering residual risk acceptance or compensating control requirements, NIST SI-5 (Security Alerts, Advisories, and Directives) —

incorporate the vendor advisory and Tenable plugin 305979 into the organization's threat register and risk register, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — add T1555.003 (Credentials from Password Stores: Credentials from Web Browsers) and T1003 (OS Credential Dumping) as explicit detection objectives in the vulnerability and threat management process

Compensating: Deploy the public Sigma rule for T1555.003 (browser credential access) — search the SigmaHQ repository for `credential_access_credentials_from_web_browsers.yml` and convert it to your log format using `sigma convert -t splunk` or equivalent. For T1003 coverage against Edge-targeting tools, use the Sigma rule `proc_access_win_lsass_dump_comsvcs.yml` as a baseline and modify the `TargetImage` field to also match `msedge.exe`. If no SIEM is available, schedule a nightly PowerShell script that parses Sysmon Event ID 10 XML logs and alerts on any source process outside a defined allowlist that accessed `msedge.exe` with high-privilege access rights.

Evidence: Before updating the threat model, pull any historical Sysmon Event ID 10 records for `msedge.exe` targets from the past 90 days to determine if T1555.003-consistent behavior has already occurred in your environment. Also review Windows Security Event ID 4688 (Process Creation) logs for known browser credential dumping utilities: `SharpChrome.exe`, `HackBrowserData.exe`, `BrowserGhost.exe`, or any `msedge.exe` child processes that spawned `cmd.exe` or `powershell.exe` unexpectedly. This retrospective hunt should be documented as part of the threat model update to establish a clean baseline or identify a pre-existing compromise.

Step 4: Communicate findings — brief leadership that this is a vendor-acknowledged, unpatched behavior with no remediation commitment; frame the risk in terms of credential exposure scope (domain accounts, SaaS tokens, privileged passwords) and the absence of a patch timeline.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: Communicating findings, updating risk posture, and informing organizational decision-making based on confirmed threat assessment

Controls: NIST IR-6 (Incident Reporting) — report confirmed credential storage risk to organizational leadership and document the vendor's acknowledgment that this is intentional behavior, NIST IR-4 (Incident Handling) — ensure the incident handling capability includes communication workflows for vendor-acknowledged, unpatched conditions with no committed remediation timeline, NIST RA-3 (Risk Assessment) — frame the leadership brief around residual risk: the attack surface is every endpoint running Edge with stored credentials, and exploitation requires only local memory read access achievable by any post-exploitation framework, CIS 7.2 (Establish and Maintain a Remediation Process) — document the absence of a vendor patch as a formal exception in the remediation process, requiring compensating control approval from risk ownership

Compensating: Prepare a one-page risk summary quantifying blast radius: count of endpoints running Edge (from Step 1 inventory), number of stored credentials by category (domain vs. SaaS vs. privileged, from Login Data review), and the access requirement (local code execution — achievable via phishing, malicious macro, or any existing foothold). Reference the Tenable plugin 305979 finding and Microsoft's public acknowledgment to establish vendor-confirmed status. This brief requires no tooling — it is derived from the inventory and policy review outputs from Steps 1 and 2.

Evidence: Compile a credential exposure summary from the `Login Data` SQLite exports collected in Step 2, categorizing stored credentials by type (domain accounts identifiable by UPN format, SaaS tokens by origin URL domain, privileged accounts by username patterns). Document the Edge version baseline across your fleet using `%LOCALAPPDATA%\Microsoft\Edge\Application\msedge.exe` version properties or the registry key `HKLM\SOFTWARE\Microsoft\Edge\BLBeacon\version`. This evidence package establishes the concrete blast radius for the leadership brief without requiring speculation.

Step 5: Monitor developments — track Microsoft's Edge security release notes (learn.microsoft.com/en-us/deployedge/microsoft-edge-relnotes-security) for any future change in vendor position; monitor for CVE assignment and any updates from Tenable plugin 305979 that may indicate a behavioral fix.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: Ongoing monitoring for changes in threat landscape, vendor remediation status, and intelligence updates following incident documentation

Controls: NIST SI-5 (Security Alerts, Advisories, and Directives) — establish a recurring process to monitor Microsoft Edge security release notes and Tenable plugin 305979 for status changes, NIST IR-5 (Incident Monitoring) — maintain an open tracking record for this vendor-acknowledged behavior and update it when Microsoft's position or Edge's behavior changes, NIST SI-2 (Flaw Remediation) — even without a current patch, maintain readiness to deploy a remediation rapidly if Microsoft releases a behavioral fix in Edge >= 147.0.3912.60 or a successor build, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — include Tenable plugin 305979 in recurring vulnerability scan reports and flag any plugin status change (new CVSS, CVE assignment, or fix verification) for immediate triage

Compensating: Create a free RSS or webhook alert using a service such as Visualping or a Python `feedparser` script targeting the Microsoft Edge release notes page and the Tenable plugin 305979 changelog. Additionally, configure a monthly calendar reminder to manually check the NVD (nvd.nist.gov) for any new CVE referencing 'Microsoft Edge password manager memory' to catch a CVE assignment that may not surface through existing feeds. For Tenable plugin tracking without a Tenable subscription, monitor the Tenable plugin search page (plugins.tenable.com) for plugin 305979 modification dates.

Evidence: Maintain a versioned log of Edge builds deployed across your fleet, updated with each Edge auto-update cycle, by querying the registry key `HKLM\SOFTWARE\Microsoft\Edge\BLBeacon\version` or parsing `%LOCALAPPDATA%\Microsoft\Edge\Application\msedge.exe` file version. When Microsoft releases Edge >= 147.0.3912.60 or a subsequent build with a security note referencing password manager memory handling, immediately re-run the Sysmon Event ID 10 and Login Data review from Step 2 to verify whether the cleartext retention behavior has changed in the updated build before updating the risk register.

Detection Guidance

Focus detection efforts on two layers: process memory access behavior and credential-use anomalies downstream.

Memory access: Hunt for processes issuing OpenProcess calls with PROCESS_VM_READ permissions against msedge.exe or its child processes. EDR telemetry should surface cross-process memory reads where the reading process is not a known legitimate parent (e.g., antivirus, debugger under controlled conditions). PowerShell, cmd.exe, or unsigned binaries accessing Edge process memory are high-confidence indicators of credential harvesting attempts.

Log sources to check: Windows Security Event Log (Event ID 4656, 4663 for handle requests to processes; Event ID 10 in Sysmon for cross-process access), EDR process telemetry for OpenProcess and ReadProcessMemory API calls targeting msedge.exe, and PowerShell script block logging for invocations of memory dumping modules or LSASS-style tooling.

Downstream credential anomalies: After a potential memory scrape, watch for authentication events using credentials that have no recent browser-based access pattern, particularly domain account logins from unusual hosts, SaaS platform access from new geolocations or device fingerprints, and privileged account use outside business hours.

Policy audit: Verify whether your organization's browser policy disables or restricts the built-in Edge password manager (configured via Microsoft Edge group policy: PasswordManagerEnabled). If the policy is not enforced to block credential storage, users may be accumulating credentials in the memory-exposed store without realizing it.

Hypothetical hunting query structure (adapt to your SIEM): Alert on any process other than msedge.exe, msedgewebview2.exe, or your defined EDR agent requesting VM_READ access to an msedge.exe process,

correlated with a subsequent network authentication event from that same host within 60 minutes (adjust this window based on your environment's typical post-compromise-to-lateral-movement timeline).

Indicators of Compromise

Type	Value	Context	Confidence
TOOL	`Memory dumping utilities (e.g., ProcDump, custom LSASS-style scrapers)`	Memory dumping tools leveraged via local process access to read msedge.exe process memory and extract cleartext browser-stored credentials without user interaction	MEDIUM

Framework Mappings

MITRE-ATTACK

- **T1552.001** — Credentials In Files
- **T1059** — Command and Scripting Interpreter
- **T1555.003** — Credentials from Web Browsers
- **T1003** — OS Credential Dumping

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **AC-6** — Least Privilege
- **IA-5** — Authenticator Management

OWASP-TOP10-2021

- **A04:2021** — Insecure Design
- **A07:2021** — Identification and Authentication Failures

CIS-V8

- **5.2** — Use Unique Passwords
- **6.3** — Require MFA for Externally-Exposed Applications
- **7.3** — Perform Automated Operating System Patch Management
- **7.4** — Perform Automated Application Patch Management

HIPAA-SECURITY

- **164.308(a)(5)(ii)(D)** — Password Management
- **164.312(d)** — Person or Entity Authentication

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.23** — Information security for use of cloud services

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1552.001	Credentials In Files	Credential-Access
T1059	Command and Scripting Interpreter	Execution
T1555.003	Credentials from Web Browsers	Credential-Access
T1003	OS Credential Dumping	Credential-Access

Sources

Source	URL	Tier
Security News	https://x.com/L1v1ng0ffTh3L4N	T3
Microsoft Says Edge Password Security Vulnerability Is 'By ...	https://www.forbes.com/sites/daveywinder/2026/05/05/microsoft-says-...	T3
Release notes for Microsoft Edge Security Updates	https://learn.microsoft.com/en-us/deployedge/microsoft-edge-relnote...	T1
Microsoft Edge Multiple Vulnerabilities	https://www.hkcert.org/security-bulletin/microsoft-edge-multiple-vu...	T3
Microsoft Edge (Chromium) < 147.0.3912.60 Multiple ...	https://www.tenable.com/plugins/nessus/305979	T3

Generated 2026-05-06 09:04 UTC by TJS Security Command Center