EDR Evasion Trend Report 2025

Prepared by Ransomwared Threat Intelligence - September 2025

Ransomwared
CTI Report

1. Management Summary

Endpoint Detection and Response (EDR) platforms have become the centerpiece of modern enterprise defense. For the last decade, organizations have increasingly relied on EDR solutions to detect and respond to malicious activity that bypasses traditional antivirus tools. Unlike signature-based detection, EDR platforms operate by monitoring system behavior, identifying anomalies, and giving defenders visibility into threats that unfold in real time.

The success of EDR, however, has created a predictable shift in adversary tactics. Attackers no longer see EDR as an obstacle to be worked around, but as a **primary target**. If they can neutralize the EDR — even briefly — they gain a critical window of opportunity to execute their attacks without interference.

What makes the latest wave of evasion techniques so concerning is the **stealth with which they operate**. In the past, ransomware operators and advanced persistent threat (APT) groups often resorted to blunt-force approaches: terminating antivirus services, deleting registry entries, or unloading drivers. While these methods sometimes succeeded, they were noisy. Most enterprises monitored for such actions, and defenders could often intervene before full-scale compromise.

Today's attackers are taking a different path. Instead of killing processes outright — which almost always raises alarms — adversaries exploit **legitimate Windows features, trusted binaries, and debugging APIs** to achieve the same outcome: a security tool that no longer functions as intended. The difference is that the process remains *alive*, tricking watchdogs into believing the security agent is still active. This is a subtle but powerful shift.

The result is the creation of short but highly dangerous **"invisibility windows."** These are moments when the EDR is running but not truly operating, leaving the attacker free to steal credentials, move laterally across the network, and deploy ransomware payloads undetected. Even a few seconds of blindness can be enough to extract privileged accounts or establish persistence mechanisms. In enterprise environments where ransomware can encrypt hundreds of machines within minutes, such windows are devastating.

Key Takeaways

- **Living-off-the-land evasion:** Trusted Microsoft-signed binaries weaponized for stealth.
- **Silent blinding:** EDR frozen, not killed, leaving watchdogs ineffective.
- **Operationalized:** Techniques moving from research into mainstream ransomware campaigns.
- **Impact:** Even seconds of blindness can mean catastrophic compromise.

Visualization: The Evolution of EDR Evasion

Era	Attacker Tactics	Defender View	Outcome
Past (2015–2020)	Process termination, driver deletion, service kill	Loud, obvious, often logged	Defenders alerted quickly, attacks noisy
Present (2020–2025)	API abuse (MiniDumpWriteDump), LOLBIN misuse, BYOVD	Security process looks "alive" but is frozen	Silent blinding creates short invisibility windows
Future (2025–2026)	Cross-platform debug abuse, modular evasion in ransomware kits, identity-layer evasion (e.g., SAML)	Harder to distinguish benign vs malicious activity	Evasion becomes a default stage of attacks, not an exception

Why This Matters Now

The rise of these stealth evasion techniques comes at a time when ransomware operators are under pressure to innovate. Law enforcement takedowns, improved patching, and growing global awareness have raised the bar for successful attacks. As a result, attackers are doubling down on methods that reduce their exposure and improve their operational stealth.

EDR evasion fits this need perfectly. It allows ransomware affiliates to continue operating without rewriting their malware families from scratch. By integrating simple, modular evasion techniques into existing frameworks, they can increase their success rates dramatically with minimal effort.

At the same time, defenders often lack visibility into these types of attacks. Security teams typically monitor for process termination, driver tampering, or registry modification. Few have robust detections for thread suspension, API abuse, or anomalous use of diagnostic binaries. This blind spot gives adversaries a reliable way to execute their campaigns.

The Road Ahead

Looking forward, organizations should expect these techniques to evolve further. Proof-of-concept attacks like **EDR-Freeze** will likely be weaponized by ransomware affiliates within the next year. The trend toward LOLBIN abuse will expand, as more signed binaries are discovered to have useful "dual-use" functionality.

We may also see similar debugging and diagnostic abuse appear on other operating systems. Linux crash-dump mechanisms and macOS debugging APIs could become attractive targets for attackers seeking parity across environments.

Finally, evasion will not be limited to endpoint security. Just as APT29 used **Golden SAML** to evade identity-layer defenses, we can expect adversaries to apply the same principles of stealth and invisibility to cloud, identity, and network detection systems.

Conclusion

EDR platforms remain essential, but they are not invincible. Attackers are learning how to neutralize them silently, creating dangerous windows of invisibility. The techniques being deployed are not theoretical — they are increasingly **operationalized in ransomware campaigns**.

For defenders, the message is clear:

- Do not assume your EDR is always functional.
- Monitor for the signs of blinding attempts.
- Layer defenses so that one tool's blindness does not mean total compromise.

The evolution of attacker tradecraft from noisy termination to stealthy suspension is one of the defining shifts in 2025. Organizations must be ready for a world where their EDR can be silenced at will — and where resilience depends on being able to detect and respond even when the "eyes and ears" of the endpoint are frozen.

Our Perspective: How Ransomwared Helps

At Ransomwared, we design our technology around the assumption that **EDR can and will be blinded**. That's why our platform provides:

- **Independent watchdogging:** We detect when security processes are "frozen" instead of truly functioning.
- **Behavioral detection of LOLBIN abuse:** We track anomalous use of trusted Windows binaries like WerFaultSecure.exe or msiexec.exe.
- **Ransomware behavior monitoring:** Even if EDR is blinded, Ransomwared catches telltale signs of mass encryption, backup deletion, or credential theft.
- **Rapid alerts on blinding attempts:** Instead of finding out after the damage is done, defenders are alerted the moment their defenses are put on ice.

Where attackers aim to silence your defenses, Ransomwared ensures someone is still watching.

2. Case Studies in EDR Evasion

Over the last decade, attackers and researchers have steadily uncovered new ways to neutralize endpoint defenses. What began as isolated proof-of-concepts has matured into **mainstream ransomware tradecraft**. The timeline below highlights key milestones, each representing a step forward in the evolution of EDR evasion.

2017 – GhostHook

GhostHook was a proof-of-concept published by CyberArk researchers, showing how attackers could abuse **Intel Processor Trace (PT)**, a hardware debugging feature, to hook kernel functions and bypass **PatchGuard** Microsoft's kernel integrity protection.

- **Why it matters:** It demonstrated that even hardware-assisted debugging intended for security and diagnostics could be **weaponized against defenders**.
- **Takeaway:** GhostHook proved that defenders couldn't just rely on PatchGuard or kernel signing — **debugging features themselves were attack surfaces**.

Defender Lessons:

- Monitor unusual use of advanced debugging features like Intel PT.
- Restrict access to kernel-level debugging for non-admin accounts.
- Advocate for vendor-side hardening of hardware-assisted features.

2017 – Process Doppelgänging

Presented at Black Hat Europe, this technique exploited **NTFS transaction features** and **process hollowing** to run malicious code inside the memory of legitimate processes without leaving traditional on-disk traces. Malware like **Dridex** and **Locky** later integrated aspects of this technique.

- **Why it matters:** It was one of the first high-profile examples of **fileless malware execution** that successfully bypassed both antivirus and early EDRs.
- **Takeaway:** Process Doppelgänging illustrated how **core Windows features** could be abused to make malicious processes appear completely legitimate.

Defender Lessons:

- Monitor for abnormal use of NTFS transactions.
- Track parent-child process creation chains for anomalies.
- Use memory integrity scanning tools capable of spotting hollowed processes.

2019–2022 – BYOVD (Bring Your Own Vulnerable Driver)

BYOVD became a staple evasion method in ransomware operations. Attackers install a **legitimate but vulnerable driver**, often signed by a trusted vendor, to gain kernel-level access and disable security agents. LockBit, BlackCat, and AvosLocker have all used this approach, including drivers from **Gigabyte** and **ASUS**.

- **Why it matters:** Signed drivers are trusted by the operating system, so attackers could bypass kernel protections without writing their own rootkits.
- **Takeaway:** BYOVD moved from **red-team technique to mainstream ransomware tradecraft**, showing how attackers exploit vendor trust.

Defender Lessons:

- Maintain strict allowlists for kernel drivers.
- Enable HVCI (Hypervisor-Protected Code Integrity) to block unsigned/legacy drivers.
- Track new driver loads for anomalies or known vulnerable versions.

2020 – Golden SAML

Used by **APT29** during the SolarWinds campaign, Golden SAML bypassed endpoint defenses not by touching processes, but by **forging Security Assertion Markup Language (SAML) tokens**. This allowed attackers to authenticate as any user, including admins, without triggering alerts.

- **Why it matters:** It proved that **identity systems can be blinded** just like EDRs. Even if endpoints were monitored, attackers could move laterally under the guise of legitimate authentication.
- **Takeaway:** Evasion techniques don't always target endpoints — **identity and cloud layers are equally at risk**.

Defender Lessons:

- Implement MFA across all privileged accounts.
- Monitor for anomalous SAML token lifetimes or unusual federation requests.
- Maintain tight visibility into identity provider logs.

2022 – HermeticWiper

During the cyberattacks on Ukraine, **HermeticWiper** leveraged a **legitimately signed driver** to disable defenses before unleashing destructive payloads. By undermining kernel protections, it prevented EDRs from seeing or blocking the attack in time.

- **Why it matters:** It showcased how **state-backed actors operationalize evasion** as a precursor to destructive campaigns.
- **Takeaway:** Signed drivers remain one of the most dangerous tools for adversaries, enabling stealth at the deepest layer of the OS.

Defender Lessons:

- Closely monitor for installation of new drivers outside normal update cycles.
- Enforce driver signing policies and revoke trust for vulnerable certificates.
- Deploy endpoint integrity checks that confirm EDR/AV drivers are still active.

2022–2023 – Raspberry Robin

A worm-like malware campaign, Raspberry Robin spread via removable media and exploited **DLL sideloading** for persistence and evasion. By placing malicious DLLs where trusted binaries would load them, attackers executed code in the context of signed processes. The malware was later linked to ransomware distribution.

- **Why it matters:** It highlighted the **blend of persistence and evasion**. DLL sideloading ensured long-term stealth inside environments.
- **Takeaway:** Even without kernel-level tricks, attackers could abuse **loading mechanisms of trusted binaries** to stay hidden.

Defender Lessons:

- Implement application control policies to restrict DLL loading paths.
- Monitor for DLL loads from unusual directories.
- Harden removable media usage policies to reduce initial infection vectors.

2025 – EDR-Freeze

A new proof-of-concept by **Zero Salarium** demonstrates how attackers can abuse the **MiniDumpWriteDump** API via **WerFaultSecure.exe** to suspend EDR processes in a race condition. By freezing EDR threads indefinitely, attackers blind defenses without killing them, avoiding watchdog alerts.

- **Why it matters:** It represents a **modern evolution** — stealthier than termination, harder to detect, and leveraging a Microsoft-signed LOLBIN.
- **Takeaway:** EDR-Freeze shows the direction attackers are headed: **short-term, stealthy blinding that creates just enough of a window for ransomware staging.**

Defender Lessons:

- Alert on WerFaultSecure.exe targeting security processes.
- Monitor for long-lived thread suspension events.
- Require EDR vendors to implement thread-level heartbeat monitoring.

From **kernel-level abuse (GhostHook, BYOVD)** to **identity manipulation (Golden SAML)** and **stealthy process freezing (EDR-Freeze)**, attackers have steadily advanced their ability to **neutralize defenses without triggering alerts.**

These case studies show that evasion has matured into a **systematic tactic**, no longer limited to proof-of-concepts or state-level APTs. Today, ransomware affiliates and commodity malware operators alike are adopting these methods, making evasion a **mainstream stage of the attack lifecycle**.

3. Mapping to MITRE ATT&CK

The MITRE ATT&CK framework has become the de-facto global standard for classifying adversary behavior. By mapping known techniques to ATT&CK IDs, defenders can **understand where in the attack lifecycle evasion occurs**, align detection rules, and benchmark their coverage against a widely recognized model.

When it comes to **EDR evasion**, the mapping reveals a clear pattern: attackers consistently target the **Defense Evasion** tactic, but often in ways that intersect with **Execution, Persistence, Privilege Escalation, and Credential Access**. This is not accidental. By blending evasion with these stages, adversaries gain both stealth and operational flexibility — ensuring their malware not only runs, but runs undetected.

Below is a detailed mapping of case study techniques to ATT&CK, with context on how they fit into real-world attacks.

Technique	MITRE ID	Kill Chain Stage	Description
Process Doppelgänging	T1055.013	Execution, Defense Evasion	Uses NTFS transactions and process hollowing to execute malicious code inside legitimate processes.
BYOVD (Bring Your Own Vulnerable Driver)	T1547.006	Privilege Escalation, Defense Evasion	Loads a legitimate but vulnerable signed driver to gain kernel access and disable security agents.
DLL Sideloading	T1574.002	Persistence, Defense Evasion	Places malicious DLLs where trusted binaries load them, disguising malware as part of signed processes.
MiniDumpWriteDump abuse	T1003.001 + T1562.001	Credential Access, Defense Evasion	Abuses Windows debugging API to dump process memory or suspend security tools.
EDR-Freeze	T1562.001	Defense Evasion	Exploits WerFaultSecure.exe and MiniDumpWriteDump to suspend EDR threads indefinitely without killing the process.
Golden SAML	T1550.001	Credential Access, Defense Evasion	Forged SAML tokens allow attackers to impersonate users and bypass authentication/logging controls.

Process Doppelgänging

- **ATT&CK ID:** T1055.013 – *Process Injection: Process Doppelgänging*
- **Kill Chain Stage:** Execution, Defense Evasion
- **Description:** Introduced at Black Hat Europe in 2017, Process Doppelgänging abuses **NTFS transactions** and process hollowing to execute malicious code inside a legitimate process. Because the underlying file never formally "exists" on disk, many security tools miss it.
- **Impact:** Malware families like **Dridex** and **Locky** integrated this method to remain invisible to traditional antivirus.
- **Why it maps here:** It is simultaneously an **execution method** (running code) and an **evasion method** (disguising that code).

BYOVD (Bring Your Own Vulnerable Driver)

- **ATT&CK ID:** T1547.006 – *Boot or Logon Autostart Execution: Kernel Modules and Drivers*
- **Kill Chain Stage:** Privilege Escalation, Defense Evasion
- **Description:** Attackers install a **legitimate but vulnerable signed driver** to gain kernel-level access. Because Windows trusts signed drivers, attackers can load them without restriction. Once active, the driver can disable EDR/AV or hide processes.
- **Impact:** Ransomware groups like **LockBit, BlackCat, and AvosLocker** have repeatedly used BYOVD, often recycling the same vendor drivers (e.g., Gigabyte).
- **Why it maps here:** The driver provides **privilege escalation** into the kernel and allows **direct impairment of defenses**.

DLL Sideloading

- **ATT&CK ID:** T1574.002 – *Hijack Execution Flow: DLL Side-Loading*
- **Kill Chain Stage:** Persistence, Defense Evasion
- **Description:** By placing a malicious DLL where a trusted binary expects to find a dependency, attackers cause the signed binary to load their code. This disguises execution and ensures persistence.
- **Impact:** Campaigns like **Raspberry Robin** used DLL sideloading to persist inside enterprise networks while staging ransomware.
- **Why it maps here:** DLL sideloading is both a **persistence mechanism** and an **execution hijack** that grants stealth by abusing trust in legitimate binaries.

MiniDumpWriteDump Abuse

- **ATT&CK ID:** T1003.001 – *Credential Dumping: LSASS Memory*
- **Additional ID:** T1562.001 – *Impair Defenses: Disable or Modify Tools*
- **Kill Chain Stage:** Credential Access, Defense Evasion
- **Description:** MiniDumpWriteDump is a Windows debugging API that can be abused to extract credentials from **LSASS** or suspend other processes. Attackers use it to dump secrets or interfere with security software.
- **Impact:** This is a common precursor to **lateral movement**, often used by both APTs and ransomware operators.
- **Why it maps here:** It touches both **credential theft** and **disabling defenses**, making it a dual-purpose evasion tool.

EDR-Freeze

- **ATT&CK ID:** T1562.001 – *Impair Defenses: Disable or Modify Tools*
- **Kill Chain Stage:** Defense Evasion
- **Description:** A 2025 proof-of-concept by **Zero Salarium**, EDR-Freeze exploits **WerFaultSecure.exe** (a Windows-signed binary) and a race condition in MiniDumpWriteDump. Instead of killing the EDR process, it freezes it indefinitely, leaving watchdogs unaware of the impairment.
- **Impact:** Creates short but dangerous **"invisibility windows"** where ransomware can stage operations without detection.

- **Why it maps here:** Its entire purpose is to **blind security software** without tripping alarms.

Golden SAML

- **ATT&CK ID:** T1550.001 – *Use of Stolen SAML Tokens*
- **Kill Chain Stage:** Credential Access, Defense Evasion
- **Description:** Famously used in the **SolarWinds campaign (APT29)**, Golden SAML allows attackers to forge SAML tokens. With a single compromise of a federation server, adversaries can impersonate any user — including admins — in cloud services.
- **Impact:** Gave attackers stealthy, long-term access across thousands of organizations without touching endpoints directly.
- **Why it maps here:** It is primarily a **credential access** technique but doubles as **defense evasion**, since logs reflect "legitimate" user authentication.

Cross-Technique Analysis

When viewed collectively, these mappings reveal three important insights:

1. **Defense Evasion is universal.** Every single technique involves impairing defenses, even when the primary purpose is credential theft or persistence.
2. **Credentials are a common prize.** Several techniques (MiniDumpWriteDump, Golden SAML) are explicitly used to steal authentication material, enabling lateral movement.
3. **Trusted components are the vehicle.** Whether it's signed drivers (BYOVD), trusted binaries (DLL sideloading), or Microsoft utilities (WerFaultSecure.exe), attackers increasingly weaponize what enterprises already trust.

Implications for Defenders

- **Mapping to ATT&CK helps coverage analysis.** Organizations can use the ATT&CK Navigator to visualize gaps — for example, if defenses only cover process termination (noisy) but not thread suspension (stealthy).
- **Defenders must track multi-stage use.** Techniques rarely occur in isolation. For instance, DLL sideloading may establish persistence, which then feeds into MiniDumpWriteDump for credential theft, which is followed by EDR-Freeze for ransomware deployment.
- **Identity-layer evasion is rising.** Golden SAML proved that evasion isn't just about endpoints. Cloud and identity systems are now equally at risk.

4. Real-World Examples

EDR evasion is no longer confined to proof-of-concepts or red-team demonstrations. Over the last five years, multiple high-profile campaigns have shown how attackers incorporate these techniques into real-world intrusions — often with catastrophic financial and operational consequences. Below are five representative examples, illustrating the variety of approaches adversaries use to blind defenses.

LockBit Ransomware (2022–2025): BYOVD with Gigabyte Driver (CVE-2018-19320)

LockBit has been one of the most active ransomware families in the world, responsible for hundreds of incidents across healthcare, government, education, and critical infrastructure. Between 2022 and 2025, LockBit affiliates repeatedly leveraged **BYOVD (Bring Your Own Vulnerable Driver)** attacks, most notably abusing the **Gigabyte driver vulnerability CVE-2018-19320**.

This vulnerable driver allowed attackers to load a signed kernel component that disabled EDR drivers, registry protections, and event logging. Because the driver was signed by a trusted vendor, it bypassed Windows kernel-mode signature enforcement. Once EDRs were blinded, LockBit operators deployed their encryption payloads across entire networks within minutes.

- **Financial impact:** LockBit victims collectively paid an estimated **$91 million in ransoms between 2020 and mid-2023**, according to U.S. DOJ filings. The downtime and recovery costs were often far greater than the ransom itself, with healthcare and municipal targets reporting **losses in the tens of millions per incident**.
- **Takeaway:** BYOVD is no longer experimental — it is mainstream in ransomware. Even outdated vulnerabilities remain potent when defenders fail to block or monitor signed driver abuse.

APT29 (SolarWinds, 2020): Golden SAML Tokens

The 2020 SolarWinds compromise attributed to **APT29 (Cozy Bear, linked to Russian intelligence)** highlighted how attackers can evade not just endpoint defenses, but also **identity monitoring**. By forging **SAML authentication tokens** (the so-called **Golden SAML** attack), adversaries impersonated any user, including administrators, in Microsoft 365 and other cloud services.

Golden SAML bypassed traditional EDRs entirely — there was no malicious binary to detect. Instead, attackers operated under the guise of legitimate identity tokens, blending seamlessly into normal administrative activity. For months, they exfiltrated sensitive data from U.S. government agencies and Fortune 500 companies.

- **Financial impact:** SolarWinds itself spent **over $40 million in legal and remediation costs** within the first year after disclosure. Downstream victims, including U.S. government agencies, faced incalculable losses in data exposure, remediation, and national security risks.
- **Takeaway:** EDR evasion is not limited to processes — **identity-layer evasion can render endpoint defenses irrelevant**.

HermeticWiper (Ukraine, 2022): Signed Driver Neutralization

On the eve of the Russian invasion of Ukraine in February 2022, HermeticWiper malware was deployed against multiple Ukrainian organizations. A critical part of the attack was its use of a **legitimately signed driver** to disable security products before executing destructive wiping payloads.

This mirrors BYOVD in concept: trusted drivers give attackers deep kernel-level capabilities, which can be used to blind or disable defenses. Unlike ransomware, HermeticWiper was not financially motivated — its goal was disruption. Entire systems were rendered unbootable within hours, crippling government and financial operations.

- **Financial impact:** Ukraine's CERT reported **nationwide disruption across banks and ministries**, with estimated damages in the **hundreds of millions of dollars** due to lost services and emergency remediation.
- **Takeaway:** In geopolitical conflicts, disabling defenses with signed drivers is a **precursor to destructive, not just criminal, operations**.

Cobalt Strike: Sleep Masking and Process Hollowing

Cobalt Strike, originally designed as a red-team tool, has become one of the most abused frameworks by ransomware groups. Its **beacons** include multiple evasion techniques:

- **Sleep masking:** When idle, the beacon encrypts its memory and enters a "sleep" state, preventing scanners from identifying malicious code.
- **Process hollowing:** Code is injected into legitimate processes, making detection far more difficult.

These techniques allow long-term persistence inside networks, with operators staging ransomware or espionage campaigns weeks or months after initial compromise.

- **Financial impact:** Because Cobalt Strike is used across many ransomware operations, losses are hard to isolate. However, given its role in attacks like Ryuk, Conti, and LockBit, **billions of dollars in global damages between 2019–2024** can be tied in part to its evasion features.
- **Takeaway:** Commodity tools with built-in evasion features significantly lower the barrier to entry, allowing even mid-tier criminal groups to deploy nation-state-level stealth.

Raspberry Robin (2022–2023): DLL Sideloading

Raspberry Robin emerged in 2022 as a worm-like malware spreading via USB drives and network shares. A key feature of the campaign was its reliance on **DLL sideloading** — placing malicious DLLs in directories where trusted signed binaries would load them automatically.

This evasion technique provided both persistence and stealth. Once installed, Raspberry Robin was used to deliver ransomware loaders for families such as **Clop** and **LockBit**. Because the malicious code executed in the context of trusted binaries, many EDRs failed to flag it.

- **Financial impact:** While exact damages are difficult to calculate, Microsoft reported that Raspberry Robin became a **major access-as-a-service operation**, feeding ransomware affiliates who collectively extorted **hundreds of millions of dollars globally**.
- **Takeaway:** Persistence mechanisms like DLL sideloading serve as **on-ramps for ransomware**, making early evasion critical to criminal business models.

Across these examples, three trends are clear:

1. **Evasion is operationalized.** Techniques like BYOVD and DLL sideloading are not niche — they are widely deployed in ransomware.
2. **Losses are measured in billions.** From the $91M LockBit extorted to SolarWinds' $40M in legal costs and Ukraine's massive disruption, the financial stakes are enormous.
3. **Evasion is evolving beyond endpoints.** Golden SAML proved that blinding defenses can also mean **identity-level compromise**, not just disabling EDRs.

5. Metrics & Frequency

Measuring the prevalence of EDR evasion techniques across the threat landscape helps distinguish isolated proof-of-concepts from mainstream tradecraft. Over the past five years, a clear pattern has emerged: **techniques that once lived in academic research or niche red-team toolkits are now deeply embedded in ransomware ecosystems.**

BYOVD: A Mainstream Ransomware Tactic

Bring Your Own Vulnerable Driver (BYOVD) has become one of the most widely adopted methods of disabling endpoint defenses. Initially a curiosity in security research circles, it is now an operational reality across multiple ransomware families.

- **Prevalence:** Present in at least **seven major ransomware families** between 2020 and 2025.
- **Ransomware families using BYOVD:**
  - **LockBit** – leveraged the Gigabyte driver (CVE-2018-19320) and others.
  - **BlackCat / ALPHV** – incorporated BYOVD into its Rust-based framework.
  - **AvosLocker** – deployed vulnerable drivers to terminate EDR services.
  - **Hive** – weaponized driver exploits before its 2023 law enforcement takedown.
  - **Magniber** – used BYOVD to elevate privileges.
  - **RobbinHood** – early ransomware noted for abusing Gigabyte drivers.
  - **Vice Society** – linked to driver abuse in Windows environments.

Why it matters: BYOVD has moved from being a red-team technique to a **mainstream criminal tactic**. Its adoption across unrelated ransomware families demonstrates how easily it can be integrated into affiliate playbooks.

DLL Sideloading: Common in Advanced Campaigns

DLL sideloading remains one of the most popular persistence and evasion methods. Attackers place a malicious DLL in a directory where a trusted signed binary expects to find its dependencies, causing the legitimate binary to execute attacker-controlled code.

- **Prevalence:** Observed in **20–25% of advanced campaigns** tracked by Microsoft, CrowdStrike, and other CTI providers during 2022–2024.
- **Examples of ransomware families using DLL sideloading:**
  - **Clop** – consistently uses sideloading in its initial loaders.
  - **RansomExx** – leverages trusted binaries for stealth execution.
  - **BlackMatter** – descendant of DarkSide, used sideloading in staging.
  - **Babuk** – abused DLL paths during early access operations.
  - **Raspberry Robin** (worm with ransomware links) – core propagation mechanism.

Why it matters: DLL sideloading provides both **persistence and evasion**, making it a reliable tool for attackers who need long-term access before deploying ransomware payloads.

EDR-Freeze: Proof-of-Concept Today, Reality Tomorrow

The **EDR-Freeze** technique (Zero Salarium, 2025) represents the cutting edge of evasion. By exploiting **WerFaultSecure.exe** and a race condition in MiniDumpWriteDump, attackers can freeze EDR processes without killing them, tricking watchdogs into believing the tools remain operational.

- **Prevalence:** Currently a **proof-of-concept**, but expected to be adopted by **red teams within 12 months** and **ransomware affiliates within 18–24 months.**
- **Why it matters:** Unlike BYOVD or DLL sideloading, which are well-understood, EDR-Freeze introduces a **new stealth model** — temporary blinding. Even a few seconds of blindness can allow ransomware staging, credential dumping, or lateral movement.

Forecast: Expect this technique to appear in commodity malware kits by 2026, just as BYOVD spread from research into ransomware within a similar timeframe.

Targeted Sectors

Attackers adopting these techniques tend to focus on **high-value, high-pressure sectors** where downtime translates quickly into ransom leverage:

- **Finance:** Banks, payment processors, and insurance providers face relentless ransomware attempts due to their ability to pay.
- **Healthcare:** Hospitals are frequent targets, with downtime literally endangering lives. Ransomware families like LockBit, Hive, and Vice Society have repeatedly hit this sector.
- **Government:** Municipal governments, ministries, and national agencies are prime targets for both financially motivated groups and state-backed adversaries.

Why these sectors? Each combines **sensitive data, limited tolerance for downtime, and regulatory pressure**, making ransom demands more likely to be paid.

Regional Distribution

While ransomware is a global phenomenon, certain regions are more frequently impacted by evasion-enabled campaigns:

- **North America:** The most heavily targeted region, due to concentration of high-value enterprises and critical infrastructure.
- **Europe:** Consistently second, with major attacks on healthcare and municipal systems in Germany, France, and the UK.
- **Asia-Pacific & Latin America:** Attacks are increasing, especially as ransomware-as-a-service affiliates expand operations beyond traditional Western targets.

Trend: What begins in North America and Europe quickly spreads globally — BYOVD followed this pattern, and DLL sideloading did the same. EDR-Freeze will likely follow suit.

Key Observations from Metrics

1. **BYOVD is mainstream:** At least seven ransomware families already operationalize it, proving its value to attackers.
2. **DLL sideloading is common:** Roughly a quarter of advanced campaigns leverage it, showing its broad utility.
3. **EDR-Freeze is emerging:** While still PoC, the adoption curve suggests widespread use within 1–2 years.
4. **Sector focus:** Finance, healthcare, and government are prime targets due to impact leverage.
5. **Geographic spread:** North America and Europe remain the epicenter, but attacks are expanding worldwide.

EDR evasion is no longer a specialist trick. With BYOVD and DLL sideloading already integrated into ransomware families like **LockBit, BlackCat, Clop, and AvosLocker**, these tactics are part of the everyday toolkit of cybercriminals. Meanwhile, new proof-of-concepts like **EDR-Freeze** hint at the next wave of stealthy blinding techniques.

For defenders, the data is clear:

- **Expect evasion.** Ransomware families are adopting it as a standard step.
- **Plan globally.** Even if your region is not the primary focus now, tactics spread quickly.
- **Prioritize critical sectors.** Finance, healthcare, and government face disproportionate risk.

EDR evasion is not just evolving — it is becoming a **core pillar of modern ransomware operations**, with measurable global financial and operational consequences.

6. Detection & Mitigation

The increasing sophistication of EDR evasion techniques underscores the need for proactive defense. Attackers are not trying to crash or kill endpoint defenses outright — they are looking for ways to **silently blind** them, using trusted binaries and OS-level features. This means defenders must shift from detecting noisy, obvious tampering to spotting **subtle behavioral anomalies**. Below we break down recommended **defensive actions** and **vendor/OS-level mitigations**.

Defensive Actions for Enterprises

Monitor Thread Suspension APIs

Techniques like **EDR-Freeze** exploit Windows APIs such as `NtSuspendProcess` and `NtSuspendThread`. While these are legitimate debugging calls, their use against security-critical processes (e.g., EDR, AV, SIEM agents) should raise immediate red flags.

- **Action:** Configure telemetry and detection rules to alert whenever these APIs are invoked against security tools.
- **Why it matters:** EDR-Freeze relies on suspending EDR threads without killing the process. Thread-level monitoring allows defenders to spot such blinding attempts in real time.

Alert on Unusual WerFaultSecure.exe Activity

The proof-of-concept for EDR-Freeze abuses **WerFaultSecure.exe**, a signed Windows binary normally used for error reporting. In almost all environments, its interaction with EDR processes should be rare to nonexistent.

- **Action:** Monitor command-line arguments and parent/child process chains involving WerFaultSecure.exe.
- **Why it matters:** Any attempt to direct WerFaultSecure.exe against security agents is likely malicious. This is a high-fidelity detection point.

Strengthen Watchdogs with Thread-Level Heartbeats

Most EDR watchdogs today monitor whether the process is alive. EDR-Freeze sidesteps this by leaving the process running but suspended.

- **Action:** Vendors and SOCs should implement **thread-level heartbeats** that verify security-critical threads are active and responsive.
- **Why it matters:** This moves defenses from “is the process running?” to “is the process doing its job?”

Block Unsigned or Vulnerable Drivers

The **BYOVD** technique has been exploited by at least seven ransomware families, often with drivers that are years old but still signed.

- **Action:**
  - Enable **Hypervisor-Protected Code Integrity (HVCI)** on Windows.

    - Maintain strict allowlists for kernel-mode drivers.
    - Revoke trust for certificates associated with vulnerable drivers.
- **Why it matters:** Signed drivers are an attacker's best friend. Blocking vulnerable ones cuts off one of the most reliable evasion methods.

Harden Identity Systems with MFA and SAML Anomaly Detection
Golden SAML showed that endpoint defenses can be bypassed entirely if attackers control identity tokens.

- **Action:**
    - Enforce **multi-factor authentication (MFA)** for all privileged accounts.
    - Deploy monitoring for anomalous SAML token lifetimes, replay attempts, or unusual geographic usage.
- **Why it matters:** Identity-layer evasion bypasses EDR entirely — defenders need visibility into federation and cloud logs.

Vendor and OS Mitigations

Restrict WerFaultSecure.exe from Targeting PPL-Protected Processes
Protected Process Light (PPL) was designed to shield EDRs from tampering. However, WerFaultSecure.exe currently operates at a high trust level, allowing it to interact with protected processes.

- **Recommendation:** Microsoft should restrict WerFaultSecure.exe from targeting PPL processes by default, or require explicit administrative overrides.
- **Impact:** Prevents abuse of this binary in EDR-Freeze-style attacks.

Enforce HVCI and Strong Driver Signing Policies
BYOVD continues to succeed because vulnerable drivers remain signed and loadable.

- **Recommendation:**
    - Vendors should revoke certificates associated with vulnerable drivers.
    - Enterprises should enforce HVCI and restrict kernel drivers to allowlisted versions.
- **Impact:** Reduces BYOVD effectiveness, forcing attackers back to noisier techniques.

EDR Vendors Must Add Self-Healing Against Frozen States
Current watchdogs often miss EDR-Freeze because the process still appears alive. Vendors must adapt to recognize **thread-level suspension** and self-recover automatically.

- **Recommendation:**
    - Implement thread integrity checks.
    - Add self-healing modules that restart suspended threads.
    - Provide telemetry for defenders when suspension attempts are detected.
- **Impact:** Eliminates the "invisibility window" created by freezing attacks.

Defending against EDR evasion requires a **shift in mindset**. Instead of monitoring only for **kills, crashes, or service stops**, defenders must look deeper — at thread states, debugging API usage, and trusted binaries behaving abnormally. Enterprises should enforce stronger driver and identity protections, while vendors must innovate in watchdog design and process integrity validation.

The good news is that most evasion techniques leave subtle but detectable traces. By combining process telemetry, strict driver policies, and identity monitoring, defenders can **raise the cost of stealth** for adversaries. The earlier noisy days of process termination are gone — but with the right mitigations, even stealth techniques like **EDR-Freeze, BYOVD, and DLL sideloading** can be exposed before ransomware achieves its objectives.

7. Forecast 2025–2026

Looking ahead, the threat landscape shows no sign of slowing down. If anything, the techniques of EDR evasion will become **more modular, automated, and cross-platform**, making them accessible to a wider set of ransomware affiliates and even mid-tier criminal groups. Based on current research and operational trends, we forecast the following developments for 2025–2026.

EDR-Freeze Adoption: From PoC to Playbook

The 2025 EDR-Freeze proof-of-concept demonstrated how a simple race condition, combined with a trusted binary (WerFaultSecure.exe), could suspend EDR processes indefinitely without killing them. Today, it remains primarily in research circles.

- **Forecast:** Within 12–18 months, we expect **EDR-Freeze or similar suspension techniques** to be integrated into ransomware affiliate playbooks. By late 2026, it may be standard across popular frameworks just as BYOVD has become.
- **Sector Impact:**
    - **Finance & Insurance:** Attackers may use invisibility windows to extract credentials for lateral movement into payment systems.
    - **Healthcare:** Freezing EDR during encryption staging would allow ransomware to hit hospitals with no alarms triggered.
    - **Government:** Municipal agencies with lean SOC teams are at higher risk of delayed detection.

LOLBIN Expansion: More Privileged Windows Binaries Abused

Attackers have already shown that **living-off-the-land binaries (LOLBINs)** like WerFaultSecure.exe and msiexec.exe can be repurposed for stealth. Historically, Microsoft's large ecosystem of trusted binaries has given attackers a wide hunting ground.

- **Forecast:** Expect to see **new LOLBIN discoveries** that allow evasion, credential dumping, or process suspension. In particular, **privileged Windows utilities** tied to diagnostics, error reporting, and update management will be attractive targets.
- **Sector Impact:**
    - **Critical Infrastructure (Energy, Utilities):** Reliance on legacy Windows systems increases exposure to newly discovered LOLBIN abuse.
    - **Technology & SaaS:** Attackers may abuse update installers or crash handlers in enterprise software environments.

Cross-Platform Expansion: Debugging API Abuse in Linux and macOS

So far, most evasion techniques have targeted Windows. But Linux and macOS both include powerful debugging and crash-dump mechanisms that could be abused in similar ways. For example, `ptrace` in Linux or the `task_for_pid` API in macOS could be leveraged to suspend processes or tamper with security agents.

- **Forecast:** By 2026, we anticipate **Linux and macOS debugging APIs** to be the focus of proof-of-concept attacks, eventually maturing into ransomware playbooks. This will coincide with the rise of **cross-platform ransomware families** (e.g., BlackCat, Hive) that already ship Windows and Linux payloads.
- **Sector Impact:**
    - **Cloud Service Providers:** Linux-based infrastructure will be directly at risk.
    - **Education & Research:** Mac-heavy environments (universities, labs) could see targeted attacks.

Automation: Red-Team Frameworks Shipping Evasion Modules

Tools like **Cobalt Strike** normalized built-in evasion. Now, successors like **Sliver** and **Brute Ratel** are becoming popular among both penetration testers and criminals.

- **Forecast:** By 2026, these frameworks will likely ship with **plug-and-play evasion modules** — everything from DLL sideloading templates to suspension exploits like EDR-Freeze. This lowers the barrier to entry for affiliates with minimal technical expertise.
- **Sector Impact:**
    - **SMBs (Small & Mid-Sized Businesses):** Criminal affiliates with lower skill levels can weaponize advanced evasion without building custom code.
    - **Manufacturing & Logistics:** Already frequent ransomware targets, these sectors will see **mass campaigns** with standardized evasion.

Identity Evasion: Growth of SAML & OAuth Bypass Techniques

The SolarWinds campaign proved that **Golden SAML** can bypass endpoint monitoring. With organizations increasingly shifting to cloud-first identities, adversaries are investing in identity-layer evasion.

- **Forecast:** Expect **growth in OAuth token theft, SAML forgery, and MFA bypasses**. Attackers may combine EDR blinding with identity manipulation to gain persistence both on-premise and in the cloud.
- **Sector Impact:**
    - **Finance:** Access to cloud-hosted trading and payment systems could be catastrophic.
    - **Healthcare:** Cloud-based patient record systems are high-value targets for extortion.
    - **Government & Defense:** Identity evasion grants long-term espionage access without tripping EDR alerts.

Overall Outlook

- **2025:** EDR-Freeze enters red-team adoption; LOLBIN abuse expands; DLL sideloading remains common.
- **2026:** Automation in frameworks like Brute Ratel makes evasion plug-and-play; cross-platform APIs (Linux/macOS) join the target set; identity evasion escalates.

Sectors Most at Risk (2025–2026):

1. **Finance** – High-value, highly regulated, attackers exploit ransom leverage.
2. **Healthcare** – Patient safety risks create maximum urgency to pay.
3. **Government & Municipalities** – Lean SOC teams make them soft targets.
4. **Critical Infrastructure & Cloud Providers** – Debugging API abuse will expand beyond Windows.

The next two years will mark a **transition from niche PoCs to mainstream criminal adoption**. Just as BYOVD moved from a research curiosity to a staple of ransomware, EDR-Freeze and identity evasion will become widespread. Enterprises should prepare for a world where:

- **Blinding the EDR is step one, not the exception.**
- **Trusted binaries are untrusted in attacker hands.**
- **Cross-platform evasion is standard.**
- **Cloud identity evasion is the new frontier.**

In short: 2025–2026 will be the years when **EDR evasion becomes automated, commoditized, and cross-platform**, hitting every sector from finance to healthcare to government.

8. Recommendations

The trend toward stealthy EDR evasion demands a **strategic shift in enterprise security planning**. Traditional endpoint monitoring alone is no longer enough. Attackers are proving they can blind, freeze, or sidestep EDR tools for long enough to steal credentials, move laterally, and deploy ransomware. The good news is that defenders can adapt — but it requires layered detection, behavioral approaches, proactive testing, and tighter vendor collaboration.

Build Layered Detection (Network, Identity, Endpoint)

No single security control can be fully relied upon. If the endpoint is blinded, defenders must still be able to **detect malicious behavior elsewhere in the kill chain**.

- **Endpoint Layer:** Continue monitoring for anomalies like process injection, driver loads, and thread suspensions.
- **Identity Layer:** Watch for unusual authentication attempts, anomalous SAML or OAuth activity, and lateral movement using stolen tokens.
- **Network Layer:** Maintain visibility into data exfiltration, command-and-control traffic, and ransomware staging.

Why it matters: Attackers rarely stop at a single evasion. By correlating signals across network, identity, and endpoints, organizations can still detect intrusions even when one layer goes blind.

Implement Behavioral Analytics to Spot Evasion, Not Just Malware

Most enterprises still focus detection on **malware artifacts** — file hashes, process names, or signatures. But EDR evasion techniques don't always drop malware. Instead, they rely on **legitimate tools used in illegitimate ways** (LOLBINs, debugging APIs, signed drivers).

- **Action:** Deploy behavioral analytics tuned to suspicious usage of system binaries (e.g., WerFaultSecure.exe accessing EDR processes).
- **Action:** Look for anomalies in driver loading — especially signed drivers that are out of date or rarely used.
- **Action:** Establish baselines for thread activity in critical processes so that sudden suspension events stand out.

Why it matters: EDR evasion is about blinding, not crashing. Behavioral analytics can catch the subtle patterns that signature-based tools miss.

Run Red-Team Simulations of EDR Freeze/Blind Scenarios

It is no longer enough to assume the EDR will "always be there." Security teams need to practice what happens **if it isn't.**

- **Action:** Include EDR-freeze scenarios in red-team exercises. Test what your SOC sees (or doesn't see) when your main detection layer is blinded.
- **Action:** Evaluate whether defenders notice secondary indicators, such as sudden WerFaultSecure.exe invocations or anomalous authentication flows.
- **Action:** Use purple-teaming to refine detections and ensure SOC analysts know how to respond if their primary tools stop reporting.

Why it matters: Practice is the only way to build resilience. Organizations that drill EDR-blind scenarios will respond faster and more effectively when real attackers use these techniques.

Work Closely with Vendors to Close Gaps in Watchdog Design and Driver Exposure

Vendors play a critical role in addressing systemic weaknesses. Many EDRs today still rely on **basic watchdogs** that only check whether the process is running. That leaves them vulnerable to freeze-based attacks. Similarly, outdated driver signing policies continue to enable BYOVD.

- **Action:** Push EDR vendors to implement **thread-level heartbeats and self-healing** against suspended states.
- **Action:** Require vendors to publish clear guidance on vulnerable driver detection and blocklisting.
- **Action:** Participate in **shared threat intelligence** programs to accelerate discovery of new LOLBINs or driver exploits.

Why it matters: Enterprises cannot close these gaps alone. Collaboration with vendors ensures systemic improvements that benefit the entire ecosystem.

Executive Guidance

For leaders outside the SOC, the message is straightforward:

- **Resilience over reliance.** Assume that your EDR can be blinded. Ensure your organization has **network and identity visibility** as backup layers.
- **Invest in readiness.** Fund red-team simulations and purple-team collaborations. Preparedness is cheaper than paying ransoms.
- **Demand accountability.** Vendors must be pressed to close watchdog and driver gaps, not just sell features.

EDR evasion is here to stay — and it will only get stealthier. But organizations are not powerless. By building **layered defenses**, embracing **behavioral analytics**, **testing resilience through simulation**, and **working with vendors to harden core weaknesses**, defenders can stay ahead of attackers.

The key is to shift the mindset: instead of asking "How do we stop ransomware?" we must ask **"How do we continue detecting when our primary tools are blinded?"** Enterprises that answer this question today will be far better prepared for the evasion-dominated threat landscape of 2025–2026.

9. Our Perspective: How Ransomwared Helps

Attackers are getting better at hiding in plain sight. Rather than overwhelming security controls with noise, they now focus on silencing them altogether. The latest wave of EDR evasion techniques, from driver abuse to DLL sideloading and the recent EDR-Freeze proof of concept, shows how adversaries can suspend or blind defenses without ever triggering the alarms that watchdogs depend on. For organizations, this creates dangerous invisibility windows where attackers can operate freely.

Ransomwared was designed with this new reality in mind. We begin with the assumption that your EDR will, at some point, be blinded. Instead of treating this as a remote possibility, we see it as a certainty. The question is not whether attackers will try to disable or freeze endpoint defenses, but how quickly you will know when it happens. That is a gap Ransomwared closes.

Where traditional watchdogs check only if a process is running, Ransomwared goes deeper. We look not just at whether the agent is alive, but whether it is actually working. We monitor thread responsiveness, detect frozen states, and recognize when a security tool has been quietly suspended. In practice, this means you gain awareness at the very moment your defenses stop defending, rather than only after ransomware has taken hold.

This vigilance extends to the way attackers misuse trusted tools. Living-off-the-land binaries such as WerFaultSecure.exe, rundll32, and msiexec are legitimate components of Windows. In normal circumstances they support error reporting, software execution, and system administration. In the hands of attackers, however, they become weapons used to blind, sidestep, or disable defenses. Ransomwared establishes a baseline of how these tools behave in your environment and identifies when their behavior shifts in ways that indicate misuse. This approach dramatically reduces noise for analysts while highlighting the moments when trusted binaries turn hostile.

Even when EDR is frozen or distracted, ransomware itself still leaves traces. File encryption, the deletion of backups, attempts to spread laterally through stolen credentials—these behaviors are hard to disguise. Ransomwared monitors for them independently, ensuring that even if attackers succeed in creating a short invisibility window, their actual malicious actions are still observed. The difference is that instead of finding out only after thousands of files have been encrypted, you are warned when the attempt to blind your defenses begins.

Our philosophy is not to replace the investments organizations have already made in EDR, antivirus, or SIEM. These tools remain essential, but in a world where adversaries target them directly, they cannot be the only line of defense. Ransomwared adds a safety net beneath them. It provides independent visibility and layered protection, ensuring that even short periods of blindness are not truly invisible.

Consider the difference this makes in practice. A ransomware operator launches an EDR-Freeze attack in a hospital network. For ninety seconds the EDR agent is suspended, unseen by its own watchdog. During that time the attacker extracts credentials, moves laterally to a file server, and prepares to begin encryption. Without additional safeguards, the first sign of trouble is a wave of encrypted files. With Ransomwared in place, the freeze attempt itself is identified, the misuse of WerFaultSecure.exe is flagged, and thread-level health checks raise alarms. Instead of reacting to a live ransomware incident, the hospital's SOC has a chance to intervene before patients are affected.

This is the role Ransomwared plays. Where attackers seek to blind defenses, we keep watch. We bring early warning to blinding attempts, visibility into the abuse of trusted system tools, and resilience against ransomware behaviors themselves. **In a landscape where invisibility has become a weapon, Ransomwared ensures your organization never operates in the dark.**