RYZOME

EDR and Agent Bypass

How attackers strike when security goes dark

Understanding how blind spots are exploited in endpoint security

Table of Contents

INTRODUCTION

EDR Bypass: When Security Goes Dark

Endpoint Detection and Response (EDR) has become one of the most widely deployed security technologies in enterprise environments. It is a core element of the modern defence stack, providing visibility into activity and the ability to detect and respond to threats in real time across endpoints, whether those are physical devices such as laptops and mobile phones, or virtual endpoints such as virtual machines and cloud workloads.

Despite its central role, EDR is increasingly being bypassed.

Attackers have developed reliable methods to disable agents, tamper with telemetry, and otherwise neutralise the protections that most organisations assume are always active. In practice, this means a security system can appear to be working as intended while, in fact, malicious activity goes completely undetected.

The challenge is not unique to EDR.

For any security solution using agents (sometimes called "sensors") such as runtime security and cloud workload protection solutions, the agent operates from inside the environment it is meant to protect. As a result, adversaries can use universally applicable techniques to undermine the security mechanisms. Ransomware groups, in particular, have operationalised these methods, embedding them in their playbooks and tooling. Once an agent is blinded, attackers gain critical time to move laterally, escalate privileges, and deploy payloads without triggering alerts.

This trend highlights a larger issue: no matter how advanced, the current layer of defence provided by EDR agents cannot provide complete protection.

Organisations that place too much confidence in agent-based solutions and are over-reliant on their capabilities risk exposing themselves to adversaries who exploit the architectural limitations of EDRs and agent-based tools to create blind spots.

333% increase in 'hunter-killer' malware

Between 2023 and 2024, for malware capable of impairing defences such as next-gen firewalls, antivirus and EDR solutions[1]

MITRE T1562: Impair Defences

The most prevalent defence evasion technique employed in malware campaigns in 2025[2]

EDR Evasion tools sold at: $350 /month $300 /bypass

Starting price for EDR evasion tools, a black market flourishing on the Dark Web[3]

New EDR killer used by at least 8 ransomware groups

Including Blacksuit, RansomHub, Medusa, Qilin, Dragonforce, Crytox, Lynx, and INC; as of August 2025[4]

01//

EDR: A Single Layer, Not a Magic Bullet

EDR delivers valuable capabilities: it monitors endpoint behaviour, flags suspicious activity, and provides forensic data to support investigations. For many security teams, it is an essential tool they turn to when responding to incidents. However, it was never designed to be a standalone solution. Moreover, it was developed in an era when threats were less sophisticated and IT environments were less complex. Today, attackers employ far more advanced techniques, and modern infrastructures are heavily virtualised and cloud-based.

Ultimately, EDR limitations are inherent to its architectural design.

Because EDR operates from within the endpoint itself, whether that endpoint is a physical device or a virtual machine, it shares the same environment as the adversaries it is meant to detect. This makes it possible for skilled attackers to interfere directly with the agent or its components. Over the past few years, this has become a standard tactic, with well-documented examples of ransomware operators and other groups disabling EDR processes, blocking communications, or using vulnerable drivers to undermine defences. The same principle applies to other categories of agent-based solutions, including runtime security and cloud workload protection platforms, which face similar risks.

The problem is compounded by the way organisations have come to view EDR as the cornerstone of security.

Extended Detection and Response (XDR) and Managed Detection and Response (MDR), for example, often depend heavily on endpoint telemetry, which means that if the agent is bypassed or its logs tampered with, the broader detection ecosystem is also affected. Dashboards may still display a “healthy” status while critical activity remains invisible.

The key takeaway is straightforward: EDR is essential, but it cannot be treated as infallible.

It should be one layer among several, not the foundation on which most of the security strategy rests. Building resilience requires assuming that agent-based solutions can and will be bypassed, and ensuring that other layers of visibility and control can compensate when they are.

02//

How Attackers Evade and Disable EDR in Practice

Security teams face a broad spectrum of attacker behaviour when it comes to bypassing agent-based protections. On one end, attackers seek to avoid or circumvent detection (evasion); in the middle, they tamper with mechanisms; and on the more aggressive side, they deliberately disable or outright kill defence tools. Understanding this spectrum can help executives and technical teams see exactly where risk accumulates, and where defensive gaps emerge.

Evade

Bypass the EDR visibility without directly interacting with it

e.g.: exploiting areas without EDR monitoring, or using legitimate tools

Tamper

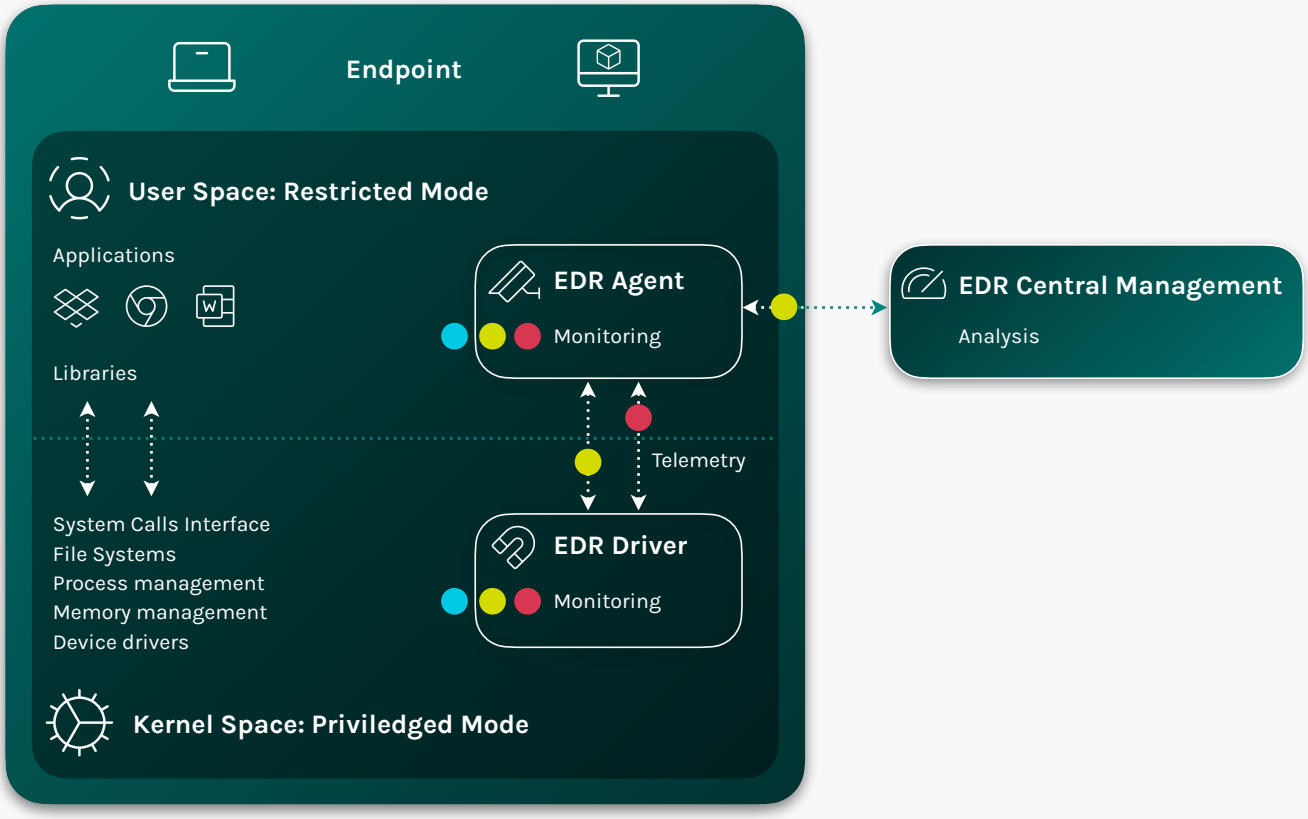
Interfere with the functioning or integrity of the EDR to degrade or manipulate its capabilities

e.g.: modifying EDR files or configurations, or removing EDR hooks used for monitoring

Disable

Outright aims to stop the EDR from running or functioning at all, like turning it off

e.g.: forcibly terminating EDR processes and services, or uninstalling the EDR agent



RYZOME

02.01//

Top Bypass Techniques

The following are five of the most common techniques adversaries use to bypass, tamper with, or disable security agents. These have been repeatedly documented and observed across ransomware campaigns and targeted attacks.

Living-off-the-land with legitimate tools

Attackers use native system utilities to execute malicious activity without introducing new binaries. PowerShell, WMI, and command-line tools can be used to stop services, alter configurations, or disable defences. For example, the LockBit ransomware group has abused PowerShell and `sc.exe` commands to bypass EDR services before deploying their payloads, while Conti operators have been observed using tools like WMIC and `taskkill` to terminate or uninstall security software, and weaken defences without introducing external binaries. Because these actions use trusted binaries, they often evade basic detection.

Process injection and unhooking

To avoid scrutiny, adversaries inject their code into legitimate processes such as `svchost.exe` or `explorer.exe`, blending in with normal system activity. They may also bypass or remove the hooks placed by EDRs. By restoring original code or using direct system calls, attackers sidestep monitoring logic. Threat actors like LockBit affiliates have leveraged these methods to run ransomware payloads within trusted processes, reducing the chance of triggering alerts.

Bring Your Own Vulnerable Driver (BYOVD)

Signed but vulnerable drivers are loaded to gain kernel-level privileges, which can then be used to disable or tamper with security software. RansomHub has employed this technique using the tool EDRKillShifter, while AvosLocker affiliates have abused an Avast anti-rootkit driver to shut down EDR functions. Because the drivers are signed, the operating system grants them high trust, making this approach difficult to block without additional controls.

Disrupting telemetry and communication

Some attackers block or corrupt the communication channels between agents and their management servers. This can be achieved by modifying policies, changing registry keys, or altering network rules. Rhysida ransomware, for example, has been observed running PowerShell scripts (such as SilentKill) to terminate security services and prevent telemetry from being sent, leaving consoles unaware of the compromise.

Purpose-built tools to disable agents

Beyond general evasion, adversaries now rely on utilities specifically designed to kill or uninstall security agents. These may terminate processes, delete drivers, or uninstall software packages altogether. Black Basta uses a custom tool called Backstab for this purpose, while LockBit affiliates deploy utilities such as Defender Control, ProcessHacker, and GMER to remove endpoint protections before launching ransomware. These are called: EDR killers.

Research Example

HookChain: Advanced EDR Evasion Technique

HookChain is a sophisticated EDR evasion method leveraging Import Address Table (IAT) hooking combined with dynamic System Service Number (SSN) resolution and indirect system calls.

By invisibly rerouting Windows subsystem execution flows, it bypasses traditional EDR monitoring at the ntdll.dll level without modifying any source code.

HookChain achieved an **88% success rate** in evading detection across evaluated EDR solutions, rendering many defenses ineffective and highlighting advanced risks in process injection and unhooking tactics.

Source: Helvio Carvalho Junior. 2024. HookChain: A new perspective for Bypassing EDR Solutions. Curitiba, PR, BRAZIL, 50 pages. https://arxiv.org/abs/2404.16856

02.02//

The Rise of EDR Killers

While many evasion techniques rely on misusing legitimate tools or exploiting vulnerabilities, a more aggressive trend has emerged: the use of EDR killers. These are dedicated tools developed with the explicit goal of targeting and neutralising security agents. Their appearance, in particular in the toolkits of major ransomware operations, shows a shift from passive evasion to active suppression of defences.

EDR killers are binaries or scripts built to identify, disable, or remove endpoint security processes and services.

They may combine service termination, registry tampering, driver exploitation, and uninstallation routines into a single package. Unlike traditional evasion techniques, which focus on stealth, EDR killers aim to ensure that the agent is no longer functional at all.

The adoption of EDR killers means adversaries are not only seeking to avoid detection but are actively targeting and dismantling the very tools defenders rely on.

This accelerates attack timelines and reduces opportunities for defenders to intervene. In practice, once an EDR killer succeeds, visibility is lost, telemetry is disrupted, and incident responders are forced to operate without critical data.

Threat reports continue to show that these tools are spreading across different ransomware groups and are being updated over time.

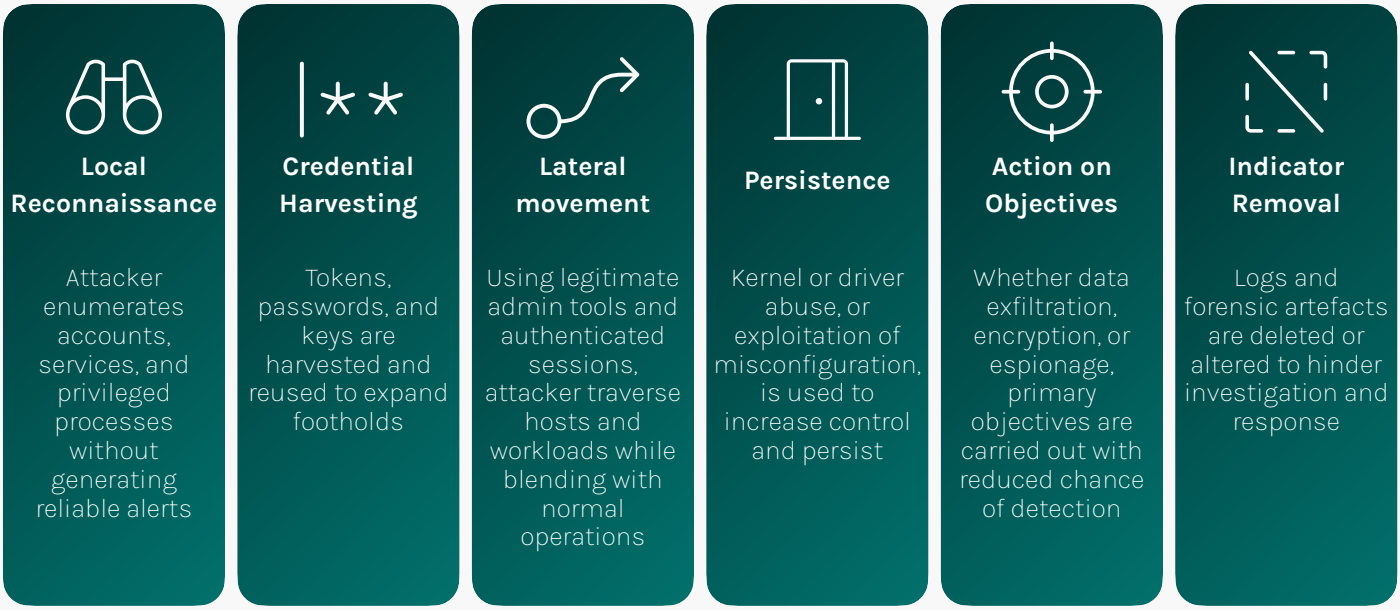
EDR Killers and Evasion Tools in the Wild

Tool Name	Threat Actors Usage	Technical Description
AuKill	FIN7, Black Basta, LockBit, Medusa Locker affiliates	BYOVD-based tool that leverages a vulnerable Process Explorer driver to terminate protected processes and disable endpoint protections.
Backstab	LockBit affiliates and other ransomware groups	Purpose-built utility used by affiliates to stop and uninstall EDR components, alter configurations, and ensure agents remain inactive.
EDRKillShifter / EDR Killer (Evolution)	RansomHub, BlackSuit, Medusa, Qilin, DragonForce, Crytox, Lynx, INC	BYOVD-style kit that installs legitimately signed but vulnerable drivers and uses a user-mode orchestrator to trigger kernel flaws and terminate or corrupt EDR/AV components.
EDRSandblast (red team tool, abused)	Undisclosed	Toolkit that automates termination and disabling of security processes and may leverage vulnerable drivers or privileged utilities to blind agents.
EDRSilencer (red team tool, abused)	Undisclosed	Silencing utility that interferes with agent telemetry and communications and implements network/registry modifications to prevent reporting.
GMER (legitimate tool, abused)	BlackSuit, Play Ransomware, LockBit, and other ransomware groups	Legitimate rootkit-detection/removal utility that provides deep kernel access and can be repurposed by attackers to remove drivers and terminate security components.
IOBit Uninstaller (legitimate tool, abused)	Play Ransomware, and others threat actors	Legitimate uninstaller abused in scripted workflows to remove agent installations and drivers without normal uninstall protections.
MS4Killer	Embargo Ransomware	Custom Rust-based EDR killer that targets specific EDR products by terminating services, removing drivers, and corrupting agent components.
PCHunter (legitimate tool, abused)	Play Ransomware, and others ransomware affiliates	Diagnostic and driver-management utility that allows stopping kernel components, removing drivers, and altering low-level system state, enabling EDR disablement.
Poortry / Stonestop	BlackCat, Cuba, LockBit, Scattered Spider, RansomHub	Combined scripts and driver-abuse techniques that stop security processes, modify registry and policy, and tamper with telemetry to create blind spots.
PowerTool	BlackSuit, Play Ransomware, LockBit	Collection of scripts/utilities that exploit drivers and administrative tooling to terminate agent processes, change Group Policy, and disable tamper protections.
ProcessHacker (legitimate tool, abused)	DoppelPaymer, Play Ransomware, and other ransomware groups	Process inspection and manipulation tool abused to suspend, terminate, or replace EDR processes and to delete forensic artefacts.
TDSSKiller	LockBit, RansomHub, and other ransomware affiliates	Toolkit that leverages rootkit-removal drivers and kernel abuse to forcibly terminate protected processes and remove agent files.
Terminator	Various ransomware groups, promoted by threat actor 'Spyboy'	EDR killer kit combining driver abuse and service/process termination to neutralise agent protections at kernel level.
TrueSightKiller	Various threat actors exploiting legacy drivers	EDR killer that targets legacy or widely-deployed drivers to achieve kernel-level process termination and supports many driver variants.

03//

What Happens in the Dark

When an agent-based defence is disabled or blinded, security teams and tools lose an essential source of telemetry and control — and much can silently unfold once visibility is removed or unreliable.



Empirical Demonstration:

How an adversary can disable defences and operate undetected in virtual machines

While much of the current reporting on EDR evasion highlights Windows environments as they are heavily targeted and widely studied, the same principles apply to Linux. Bypassing or disabling agents in Linux is just as feasible, and just as damaging.

In the next infographic, we demonstrate how, on a Linux host provisioned with an up-to-date version of a well-known runtime security monitoring tool, an adversary with local access and a working privilege escalation exploit can disable defences to exfiltrate sensitive files while remaining undetected.

Threact Actor

Initial Access

Linux VM

What Happens on the Virtual Machine	What is Seen with an Agent-Based Runtime Security Tool
01  **Privilege Escalation** T1068 Exploitation for Privilege Escalation Exploit system vulnerabilities to get root privileges	**Alert: None** Why? Lack of alert rule on the mounting of a filesystem without `nosuid` option set
02  **EDR Silencing** T1562.006 Impair Defences: Indicator blocking Kernel patched to bypass security solution's eBPF hooks, blocking telemetry about the kernel activities	**Alert: None** >> Lack of alerts simply appears as an absence of events
03  **Access Persistence** T1098.004 Account Manipulation: SSH Authorised Keys Attacker's private key added to local account's list of SSH authorised keys	**Alert: None** Why? Communication channel remains intact but no event are generated as event hooking is bypassed.
04  **Data Exfiltration** T1048.002 Exfiltration Over Asymmetric Encrypted Non-C2 Protocol Sensitive file uploaded to HTTPS server	**Alert: None** Why? Accessing remote HTTPS server appears as normal activity
05  **Evidence Erasure** T1070.002 Indicator Removal: Clear Linux System Logs All attack-related entries are deleted from the syslog	**Alert: None** Why? Communication channel remains intact but no event are generated as event hooking is bypassed. >> Forensic trails are lost
06  **Evidence Erasure** T1070.003 Indicator Removal: Clear Command History Local account's shell command history deleted	**Alert: None** Why? Communication channel remains intact but no event are generated as event hooking is bypassed. >> Forensic trails are lost

What would you see with an additional layer of defence?

Detect previously unrevealed threat activity with **Ryzome Security Monitor,** our agentless, hypervisor-based threat detection solution.

Request a live demo to see the differences in threat detection with and without visibility from Ryzome.

04//

Consequences of Inaction

Treating EDR bypass, or agent-based security bypass in general, as unlikely or peripheral creates structural weaknesses that cascade across detection, response, and overall security posture. Below are the core impacts organisations must anticipate.

Detection and response become delayed or simply ineffective

When agents are the primary source of detection, their compromise leaves security teams blind. With reduced visibility, attackers can remain longer in the environment before detection. Investigations are slower because crucial process, event, and telemetry data are missing or untrustworthy. This forces responders into manual discovery at scale that is slower, more error-prone, and often too late to prevent damage; especially considering how short attack time frames have become.

Forensics and root-cause analysis degrade in quality

If essential logs and traces are absent or tampered with, root-cause analysis is compromised. It slows remediation and increases the likelihood of recurring compromise because teams lack reliable evidence to identify how persistence or lateral movement was achieved.

Escalation of impact

EDR killers and targeted tampering extends the window of opportunity between breach and objective. This increases the chance of large-scale encryption, broader data theft, or deeper infrastructure compromise before defenders can react.

Architectural fragility and single-point dependence

Over-reliance on agent telemetry creates a single point of failure. When that point is attacked, upstream systems that depend on the agent feed (e.g., XDR, SIEM, MDR workflows, automated playbooks) produce misleading signals. The effective control plane is weakened and automated containment actions can fail.

Hidden risk across virtual and cloud workloads

Virtual machines, containers, and cloud workloads are not immune. Agent bypass in cloud-native contexts can enable lateral movement across tenant boundaries or permit actions against orchestration layers. That amplifies risk in environments that assume platform isolation as sufficient protection.

Strategic and operational consequences

If organisations continue to treat agent coverage as equivalent to adequate security coverage, investments and operational attention remain misaligned. This perpetuates reactive patching and rule-based tuning rather than investing in compensating, independent visibility and controls.

Clear implication for defenders

The practical consequence of inaction is simple: detection and response become largely dependent on an attackable data source. The defensible response is to assume agents will fail at some point and to design compensating capabilities that restore independent, trustworthy visibility and control across devices, virtual machines, and cloud workloads.

> **Treat agent bypass as inevitable.**
>
> **Architect for visibility and detection that remain intact — even when attackers succeed in throwing endpoint agents into the dark.**

Why does it matter?

Attack timelines are shrinking
Attackers are moving faster, giving less time for defenders to detect and respond

17 hours
Average time-to-ransom (TTR); some groups operate even faster, deploying ransomware in <1hour[5]

<5 hours
For attackers to exfiltrate data in 25% of incidents; in one in five cases, data theft occurred in <1 hour[6]

48 minutes
Average eCrime breakout time; with the fastest breakout time they observed being 51 seconds[7]

Data breaches are costly
But identifying breaches faster and by internal security teams minimise damages

USD 4.44M
The global average cost of a data breach[8]

Nearly 5%
Decline in average costs for data breaches with a lifecycle under 200 days[8]

Around 18%
In cost reduction when breaches are detected internally versus when disclosed by a third party or attackers[8]

05//

Designing for Resilience

Continuous Visibility as the New Rule of Engagement

The evidence is clear: attackers can and do bypass agent-based defences.

Techniques once considered advanced are now mainstream, packaged into ransomware toolkits and shared across groups. The result is that disabling or blinding endpoint agents has become a predictable step in modern attack chains. And this reality should serve as a warning shot.

The strategic lesson is straightforward. Security architectures built on the assumption that endpoint agents will always be present and reliable are fragile by design. Once that assumption fails, detection gaps emerge, response timelines lengthen, and attacker dwell time increases.

Designing for resilience means embracing a new rule of engagement: continuous visibility must not depend on the components attackers target.

Independent sources of truth are required so that defenders retain visibility even when agents are disabled, tampered with, or operating in degraded states. This shift moves security from a posture of trust in a single control, to a posture of layered assurance where attackers must overcome multiple, diverse barriers to remain undetected.

06//

Resilience and Stealth in Virtualised Environments

06.01//

The Virtual Machine Introspection Advantage

A resilient architecture requires visibility and security controls that adversaries cannot manipulate from within.

Establishing independent vantage points that remain reliable even when traditional agent-based solutions are compromised is essential. If a compromised system cannot be trusted, defenders need independent, outside-in visibility to maintain continuous coverage and close gaps in their security posture.

Virtualisation technology makes that possible.

In virtualised and cloud environments, the hypervisor layer offers a unique position to monitor workloads without residing inside them.

Virtual Machine Introspection (VMI) is a technique that enables the monitoring and analysis of virtual machines and cloud workloads from the hypervisor layer. Hypervisor-based monitoring solutions that leverage the introspection capabilities of the hypervisor are out-of-band and untouchable from within the guest operating system. Unlike an agent, it cannot be uninstalled, terminated, or tricked into silence, making it resistant to bypass techniques and the targeted “EDR killer” tools seen in the wild.

By anchoring detection and monitoring in the hypervisor, organisations add a resilient and stealth security layer that closes a critical gap.

This line of defence provides a reliable, independent source of truth, even when virtual machines and cloud workloads are compromised.

In times where adversaries assume they can blind endpoint defences, such solutions offer defenders a way to keep the lights on and continue watching without being seen.

06.02//

The Ryzome Advantage

Ryzome is built for stealth, precision, and deep observability to enhance security in virtualised environments. We instrument the hypervisor and leverage its introspection capabilities to monitor guest activity in real-time and in great detail, without any agent or footprint inside your virtual machines.

What sets us apart

Continuous, real-time introspection

Ryzome is the only solution leveraging live, in-vivo introspection for security monitoring and threat detection in production environments. This technology enables continuous, real-time visibility into workloads, which eliminates the blind spots created by snapshot-based or periodic approaches and allows defenders to detect and respond to attacks as they happen.

Broad coverage of malicious behaviour

While some solutions are tuned to detect activities related to very specific categories of threats such as cryptominers or rootkits, Ryzome maps and detects a growing range of adversary TTPs (Tactics, Techniques, and Procedures) to detect malicious activity regardless of threat type.

Independent of infrastructure providers

Our solution can be deployed anywhere there is direct hypervisor access: on-premises, public or private cloud, or custom data centres. Unlike approaches tied to a single public cloud or infrastructure provider, Ryzome extends protection across heterogeneous and hybrid environments.

>>

If you're interested in learning more about this technology and how Ryzome can help with the EDR and agents bypass risks outlined in this material:

Visit ryzome.com

Contact us for a demo

REFERENCES

1. “Red Report 2024” (Picus Security, 2024); available at: Link
2. “Red Report 2025” (Picus Security, 2025); available at: Link
3. “An Inside Look at the Black Market for EDR Killers on the Dark Web” (ExtraHop, 2024); available at Link
4. “Advisory on New Endpoint Detection and Response (EDR) Killer Tool Used by Multiple Ransomware Groups” (Cyber Security Agency of Singapore, 2025); available at: Link
5. “2025 Cyber Report” (Huntress, 2025); available at: Link
6. “Global Incident Response Report 2025” (Palo Alto, 2025); available at: Link
7. “2025 Global Threat Report” (CrowdStrike, 2025); available at: Link
8. “Cost of a Data Breach Report 2025” (IBM, 2025); available at: Link

BONUS RESOURCE

A Buyer's Mini-Checklist of 5 Essential Terms to Clarify

To counter the problem explored in this material, you may start (re)evaluating your endpoints, virtual machines, and cloud workloads security technologies. In that process, you'll often encounter similar language from different vendors, but what they mean, and how they're delivered, can vary significantly.

This checklist is designed to help you move past 5 key buzzwords and get to the substance of what's being offered. You can use this guide in conversations with your current vendors or when assessing new ones. The aim is simple: ensure that when a vendor says "X," you know what questions to ask and what it should really mean in practice.

Buzzword	What vendors usually mean	What you should ask	What it *should* mean
"100% Tamper-Resistant" or "Tamper-Proof"	Anti-tampering mechanisms (controls or policies) that make it harder to disable the agent. It may achieve "100%" scores in comparative tests, but only against tested techniques. If it uses an agent, the risk is architectural, so it will always be there.	*"Are your security controls running inside the same OS they protect, or outside of it?* *If they run inside, how can you credibly claim to be 100% tamper-proof against attackers with admin/root or kernel access?"*	**Tamper-resistant by design:** Monitoring outside the workload/OS; cannot be touched from inside.
"Agentless-First with Lightweight Sensors"	"Lightweight sensors" are essentially agents under another name. The truth is: some features (such as extending visibility into runtime) still rely on agents.	*"Does the solution require anything inside the endpoint, VM, or workload to deliver runtime visibility?* *Which other capabilities are dependent on agents?"*	**True agentless:** No agents footprint, nothing attackers can directly access or tamper with, and core capabilities are not extensively limited by the lack of agents.

Buzzword	What vendors usually mean	What you should ask	What it *should* mean
"Real-time visibility"	Agent-based tools usually deliver continuous monitoring. Agentless tools may or may not: some stream selected data continuously, others use periodic polling or snapshots. For hypervisor-based solutions, only in-vivo introspection provides real-time visibility.	*"Is this continuous, live monitoring, or periodic snapshots?* *How often is telemetry collected?* *What's the delay between activity and detection?"*	**Continuous real-time**: Live, uninterrupted monitoring of workload activity, not delayed or sampled.
"Next-Generation" or "Next-Gen"	Often means the current version with incremental updates or features.	*"What's the actual innovation here?* *Is there a genuine technology shift, or just an iteration?"*	**Meaningful innovation**: A real departure from legacy approaches that break from legacy limitations, not just rebranding or added features.
"AI-Powered"	Could be anything from pattern matching to standard ML models, or just a wrapper around existing tools (or, let's be real, just a prefix for the sake of checking the AI-box...)	*"What AI/ML methods do you use, and for which tasks?* *What data was it trained on? How is accuracy validated?"*	**Substantive AI/ML:** Clearly defined role in detection or response, transparently explained.

RYZOME

About Ryzome

Ryzome empowers organisations to secure their virtualised environments against advanced threats. Ryzome's "outside-looking-in" approach provides a new line of sight that traditional solutions cannot match observing virtual machines and workloads from the outside to detect threats that slip past conventional defences.

Powered by proprietary hypervisor-based introspection, Ryzome Security Monitor eliminates the need for agents and delivers continuous visibility at runtime, real-time threat detection, detailed forensic trails, and actionable intelligence, enabling security teams to detect hidden threats, accelerate investigations, and reduce business risks.

Ryzome makes security in virtualised environments more reliable and harder to evade.

Learn more at ryzome.com