

How Cyber Crucible Compares

Straight answers to the comparison and evaluation questions buyers ask - versus EDR, XDR, MDR, antivirus, DLP, and backup-based recovery.

- [Is Cyber Crucible an EDR, XDR, MDR? Or Something Else Like Agentic AI?](#)
- [Is This a Monitoring Tool Like My Other Security Dashboards?](#)
- [Does Cyber Crucible co-exist with other security products?](#)
- [What happens if one of my security products has a conflict with Cyber Crucible?](#)
- [How is Cyber Crucible different from EDR, XDR, and MDR?](#)
- [Does Cyber Crucible replace my EDR, or work alongside it?](#)
- [Is Cyber Crucible antivirus?](#)
- [Why isn't backup and recovery enough to handle ransomware?](#)
- [Why is capturing ransomware encryption keys no longer a valid defense?](#)
- [Why do "cloud collector" security tools struggle against modern attacks?](#)
- [Does Cyber Crucible slow down my systems?](#)

Is Cyber Crucible an EDR, XDR, MDR? Or Something Else Like Agentic AI?

Short answer: It has elements of both EDR and XDR, but the category question is the wrong one. What matters is *where the decision is made*. Cyber Crucible's response logic runs entirely on the endpoint using only information available there at the moment of attack — because remote analytic servers introduce two fatal weaknesses: latency and fragility.

The honest answer to the label question

- **As an EDR:** if an EDR uses endpoint telemetry to make decisions, then Cyber Crucible's automated response is an EDR. It clamps down on extortion behaviour in milliseconds, using only local information.
- **As an XDR:** telemetry is also sent to a database (customer appliance or central) for correlation, threat hunting, insider-threat work, and IT audits. By the XDR marketing definition, that qualifies.

The team is comfortable with "EDR for extortion defense," even though edge computing is seen as last-generation in some circles. The reasoning below is why that view is backwards.

Why remote analytics became a liability — latency

The "X" in XDR represents moving analytical computing power to remote servers. That buys power and costs time, and attackers built their tradecraft around the gap:

- **Speed:** attacks were accelerated so irreversible actions complete before an analytic server can respond. This is especially visible with small, high-value items like passwords.
- **Parallelism:** many endpoint tools inspect one program, wait, then move to the next. Attackers run multiple extortion programs at once — 5,000 files accessed in parallel instead of 500.
- **Distribution:** extortion runs across many machines simultaneously. Cyber Crucible has observed roughly 75 machines at once. A tool then faces 50 programs across 75 machines — 3,750 programs to inspect.

- **Commander processes:** some attackers run a local controller that instantly re-spawns any extortion tool that gets killed.

Detection-and-response strategies wait, by their nature, for the attack to be underway. The better analogy is stopping bank robbers at the door rather than after a certain amount of cash has left the safe.

Why remote analytics became a liability — fragility

Around 80% of EDR and XDR solutions need access to remote analytic servers to function optimally, and are barely functional without that cloud "brain."

Attackers exploit this directly: gain enough access to change firewall rules, block the endpoint tool from reaching its analytic servers, and the tool loses its ability to analyze or respond. It remains unexploited, installed, running — and almost completely ineffective.

What Cyber Crucible does instead

Because latency and fragility make cloud dependency untenable for extortion defense, Cyber Crucible invented a detection and response capability whose behavioral analytics use **only information available on the endpoint at the time of attack**. Interdiction happens locally in typically under 200 milliseconds, with no cloud round trip in the critical path.

The telemetry that does flow to the database supports investigation rather than protection — threat hunting, insider threat detection, IT audits — and an open API lets it feed open XDR platforms, SOAR, or RPA tooling. That work is valuable, but it is never what stands between you and an attack in progress.

Is This a Monitoring Tool Like My Other Security Dashboards?

The answer is, it depends on how you and your organization wants to use the tool, but probably "no".

It is important to note to product design strategies with Cyber Crucible:

1. Protection should be hyper-automated, and not require constant care and feeding (alert attention), nor knob-turning (configuration) by the users.
2. For more resource-rich security teams, the valuable telemetry should be available to them, for advanced threat hunting activities, or to support forensics and incident response activities.
3. Action #2 above, should in no way degrade the protection in place for action #1.

The most common feedback we receive from IT leadership is that their employees did not know Cyber Crucible had been protecting them for weeks already.

We have many users that leverage the tool in three capacities:

Set and Forget (Like Your Smoke Alarms)

Many organizations lack the resources to have threat hunters diving into Cyber Crucible (or other tool) telemetry, but they want the risk of data extortion “off of their plate”. That is perfectly OK, and Cyber Crucible’s Data Extortion Prevention is designed to do exactly that.

These users lack the resources to spend time looking at the Process Injection, Credential Protection, and Process Creation data feeds.

Hence, we find, unless they are responding to a rare automated response by Cyber Crucible, or are performing inventory management, like when a new machine needs Cyber Crucible deployed to it...

...they simply do not login to the Cyber Crucible web portal, and leave the automation to just do its job.

That is perfectly OK!

Advanced Threat Hunting, + Automated Protection

The [process injection](#), credential use monitoring, and [process creation](#) capabilities provide a rich set of data, captured via kernel-level behavioral analytics, to conduct threat hunting.

The automated protection kicks in once data theft, credential (token, password) theft, or ransomware encryption begins. For everything else - this is a great resource.

The most common finding thus far is unmanaged (at least by the current IT admins) remote management tools installed by users. Yes, they do sometimes result in an automated extortion protection (aka, the unmanaged RMM is used by a bad actor), but it is always better to know more about your environment beforehand, regardless!

Here is a quick video of a type of attack we saw in use by attackers, in multiple client environments, that was used to successfully evade the customers' EDR solutions (we don't like to names...there were 3 products in four enterprises).

[rr-com-dll.mp4](#)

Post-Incident Forensic Analysis & Remediation

There are a couple types of use cases for Cyber Crucible data.

The first is so that a security person at a company, or an IT person, can investigate root cause for a program to be suspended. For some customers, who typically install our software, then only login again when there are new employees to onboard, they have never used any Cyber Crucible functionality except for the Manage Agents page. We're glad to help them learn about [root cause analysis](#) - which is painless and simple for technical and semi-technical users.

The second type of customer for post-incident forensic analysis are DFIR firms that are called in to investigate an issue at a client, sometimes at the cyber insurance company's request, and Cyber Crucible is installed after an issue. (Hey, we can't always be there before the attacker, as much as we want to be...otherwise every company in the world would have us, and extortion would be history!)

The best way to describe a roll-out after an extortion attack is, “messy”. Not because we are difficult to install or start protecting. Quite the opposite! In that instance, the attackers have normally moved around the system, and are embedded in operating systems, network gear, Active Directory, and are operating in memory. Multiple machines end up having processes suspended, and sometimes the attackers even try to regain control. It is a bit like taking medicine that makes you sick in the beginning, but you need it to ever get better.

Clients who are already stressed out, sometimes react emotionally to “more” noise, as the attackers are finally rooted out and defeated. There is a lot of education as to why or how their in-place security stacks didn’t find the attackers. At the end, though, the business regains control over their IT infrastructure. They will not become one of the 80-90% of companies that are re-extorted a year or two later, in what we call, “the extortion subscription economy”.

The third type of customer we have is an MSSP that, oftentimes is a partner of Cyber Crucible, that is responding to an automated protection from Cyber Crucible. They need to find out the way the attacker achieved a temporary foothold, and close any holes in security. The good news is that HIPAA or other compliance reporting typically reports little or no breach of confidentiality, due to the speed of the automated response. Many times, the MSSP reports, “nothing to report”, which makes everyone involved happy. Well, except for the criminal.

Does Cyber Crucible co-exist with other security products?

Yes!

Cyber Crucible is currently deployed and co-existing with every security vendor on Gartner's Magic Quadrant.

Our software automatically discovers and configures itself to be aware of the other security tools' activities.

In the rare event there is a potential conflict, please see [this knowledgebase article](#).

What happens if one of my security products has a conflict with Cyber Crucible?

Cyber Crucible currently co-exists with every major security vendor, if we use Gartner's Magic Quadrant reports as a metric of industry incumbents.

Cyber Crucible's endpoint software is reliably more resilient than other software, and that includes other endpoint security tools. Additionally, the other security software typically does not even have full visibility into Cyber Crucible's software.

Another security tool attempts to shut off Cyber Crucible

Summary:

The other tool tries to shut off part of Cyber Crucible. They cannot. The other vendors' tool has not tested what happens when another tool (aka, Cyber Crucible), says, "no, I'm not shutting off". The other tool crashes or enters an error state.

Cyber Crucible's response:

The response can be two fold. The first is to whitelist Cyber Crucible software with the other vendor. This will cause the other tool to ignore Cyber Crucible software, which will (hopefully) prevent the other tool from entering an error state.

If that doesn't work, it is up to our team to come up with a solution. We've learned that other vendors move a lot more slowly (several months or never) to fix a bug on their part.

Historically, in the few times this has happened, we've essentially created an environment where the other tool thinks it succeeded. A non-tech analogy may be letting your young family member or child beat you at checkers.

Another security tool attempts to run code as Cyber Crucible

Summary:

This happens very rarely, but there have been instances where a security tool attempts to inject its code into Cyber Crucible's running program, and execute code posing as us.

This is no different than a hacker posing as a security tool, and there is no way to differentiate if a hacker has taken over the "trusted" security (or any other) security program.

Cyber Crucible's response:

There is not a scenario where Cyber Crucible will allow another company (or hacker) to run code posing as our product. The only solution at this point would be to whitelist Cyber Crucible software with the other vendor's tool. There is no mitigation Cyber Crucible can provide for this, to prevent the other tool from attempting this behavior.

Fortunately, this is a very rare event.

Cyber Crucible is disabled by another tool

There were various instances where security tools with highly privileged accesses would repeatedly attempt to disable Cyber Crucible software. This included repeatedly attempting to remove our driver, or kill our service over and over.

While those issues never affected extortion defense, it did negatively impact user experience and system performance.

All of that is historical.

If another tool has the ability, legitimate or not, to disrupt Cyber Crucible - then a hacker could as well.

If you suspect or observe Cyber Crucible software diminished in any way, regardless of the software that caused it, please contact us immediately.

We are usually last tool standing between an extortionist and your critical data. There is no room for weakness in that circumstance.

How is Cyber Crucible different from EDR, XDR, and MDR?

Short answer: EDR, XDR, and MDR are detection-and-response tools: they observe an attack, generate an alert, and rely on a human or a cloud service to decide what to do. Cyber Crucible is a prevention tool: it decides autonomously on the endpoint and stops the malicious process in under 200 milliseconds, before damage occurs.

The practical differences

	EDR / XDR / MDR	Cyber Crucible
Response time	Minutes to hours	Under 200 milliseconds
Detection method	Signatures and historical data	Behavioral intent, no signatures
Zero-day coverage	Partial	Built for unknown threats
Human dependency	High — analysts and SOCs	None — fully autonomous
Where decisions happen	Cloud analytics	Locally, in the kernel

Why the gap matters

Detection tools were designed for an era when attacks unfolded over hours or days. Automated attacks now complete in seconds. A tool that produces an accurate alert after the data has been exfiltrated has documented the loss, not prevented it.

This isn't a claim that detection has no value — forensics and investigation matter. It's that detection alone cannot stop a machine-speed attack.

Does Cyber Crucible replace my EDR, or work alongside it?

Short answer: Either. Cyber Crucible is designed to run alongside existing EDR, XDR, and MDR tools without conflict, and many customers deploy it that way. It also works as a standalone prevention layer if you decide to reduce your stack.

Running alongside your current stack

Cyber Crucible does not require you to remove anything. It adds autonomous prevention beneath the detection layer you already own, and it works with or without EDR and MDR present.

Customers frequently start this way deliberately — deploying Cyber Crucible to find out what their existing stack is missing. In one financial services deployment, that answer arrived within 60 days: nearly 10,000 malicious processes intercepted that three EDRs and an outsourced SOC never alerted on.

Reducing the stack

Once teams see how often their detection tools don't engage, some choose to consolidate. There's no vendor lock-in and no forced integrations — you can replace your EDR entirely or simply cut spend where it isn't earning its keep.

Is Cyber Crucible antivirus?

Short answer: No. Antivirus identifies known-bad files using signatures. Cyber Crucible uses no signatures at all — it evaluates what a running program is actually doing and stops malicious behavior, including threats no one has seen before.

Why signatures are no longer sufficient

A signature can only describe an attack someone has already analyzed. That model fails in two common situations:

- **Zero-day attacks**, where no signature exists yet.
- **Fileless attacks**, where there is no file to match a signature against.

Signature-based detection also requires constant updating from threat intelligence feeds, which means your protection is always trailing the attacker.

What replaces it

Cyber Crucible models behavior — memory activity, process activity, and file access — and assesses intent in real time. That's why it has stopped zero-day attacks on customer networks as much as 90 days before any other vendor publicly reported or responded to them.

Why isn't backup and recovery enough to handle ransomware?

Short answer: Backups address only the final stage of a ransomware attack — encryption. By the time encryption starts, attackers have usually already stolen credentials and exfiltrated your data. Restoring from backup recovers your files; it does nothing about the data already in the attacker's hands.

Ransomware is the last step, not the first

A typical intrusion runs in three stages:

1. **Identity theft** — credentials are compromised to gain access.
2. **Data theft** — intellectual property, customer records, and other sensitive data are exfiltrated.
3. **Encryption** — files are locked and a ransom is demanded.

Recovery strategies engage only at stage three. The confidentiality breach and loss of operational control have already happened.

The real cost

The largest business risk from ransomware usually isn't the ransom itself — it's revenue lost to a prolonged outage and the reputational damage that follows. Preventing the initial compromise protects both, which is why prevention has to come before recovery planning rather than instead of it.

Why is capturing ransomware encryption keys no longer a valid defense?

Short answer: Key capture stopped being technically reliable in 2021. Modern ransomware avoids the operating system's standard encryption libraries, so there are no keys for a defensive tool to intercept — and even when keys are obtained, they often can't be applied.

How modern ransomware bypasses key capture

- **Custom libraries:** encryption code is compiled directly into the malware instead of calling OS libraries, so keys are generated and used entirely inside the attacker's own process — invisible to tools watching standard calls.
- **Unique implementations:** some strains use non-standard, per-execution encryption. Even a captured key can't be applied, because the algorithm itself is bespoke.
- **Streaming ciphers:** these are built on a fundamentally different concept from AES and are simply immune to AES key capture.

Cyber Crucible originally developed and patented key-capture technology, then presented the cryptanalysis showing its limits at BSides Pittsburgh in 2021. We moved on because the evidence said to.

The compliance problem too

Key capture requires storing encryption keys centrally — creating a single point of failure and a high-value target. A government could also compel access through a classified subpoena or national security letter, without your knowledge. That's a serious data-sovereignty risk on top of a defense that no longer works.

Why do "cloud collector" security tools struggle against modern attacks?

Short answer: A cloud collector is a lightweight endpoint agent that does little local analysis and instead ships raw telemetry to a cloud platform for processing. That round trip adds delay an automated attack doesn't give you, and the agent depends on the very OS components attackers now compromise.

Why the industry built it this way

Kernel-level engineering is difficult and expensive. Shipping telemetry to the cloud let vendors build cheaper endpoint software, monetize cloud storage, and market "cloud AI" capabilities. The model was optimized for ease of development, not for defense.

The two failure modes

1. **Too slow.** Automated smash-and-grab attacks complete in milliseconds. Cloud analytics cannot issue a response in time.
2. **Blind when it matters most.** Because these agents depend on native OS libraries to function and collect data, an attacker who compromises those libraries can silence the agent. It keeps running and reports nothing — a dashboard saying all is well while data leaves.

Cyber Crucible processes locally in the kernel and makes its own decisions, so there is no cloud round trip and no dependency on a potentially compromised OS.

Does Cyber Crucible slow down my systems?

Short answer: No. Typical CPU usage is around 1% or lower, and the architecture deliberately avoids the technique that causes most endpoint-agent slowdowns — inserting hooks into Windows system libraries.

Why other agents cause lag

Legacy security tools insert "hooks" into highly optimized system libraries to watch for bad behavior. Those libraries were never designed to be modified on the fly. When a security tool and an attacker are both manipulating them at once, the result is significant system degradation.

The pattern that follows is predictable: the user blames the security tool for the slowdown and removes it. The vendor can't reproduce the problem in a lab, unaware that a root-level adversary is manipulating the system. Attackers have learned to exploit this deliberately — degrading performance until victims disable their own defenses.

A different approach

Cyber Crucible operates through independent subsystems inside the kernel rather than hooking fragile Windows components, so it doesn't compete with the operating system — or with an attacker — for the same fragile code paths.