

How should we run a proof of concept for a prevention tool?

Short answer: Deploy it alongside your existing stack in a real production environment and compare what each tool sees. The most informative result isn't a lab test — it's the gap between what your current tools alert on and what actually gets intercepted.

Why deploying alongside works best

The financial services firm in our best-documented case did exactly this. They didn't remove anything. They added Cyber Crucible specifically to find out what their existing investment was missing, and had a definitive answer within 60 days.

What to measure

- **Interceptions your current stack never alerted on.** This is the core signal.
- **Time to decision.** Prevention has to complete before damage, not before a ticket is closed.
- **Business disruption.** Count downtime, reboots, and false-positive-driven interruptions.
- **Analyst hours consumed.** Autonomous prevention should reduce workload, not add a queue.

A caution about lab testing

Testing against samples pulled from malware databases often measures very little, because attackers design malware to stay dormant unless it receives a validation signal from a live command-and-control server they still operate. A sample that does nothing in your lab proves nothing about a live attack.

Revision #3

Created 2026-07-20 19:24:17 UTC by Dennis Underwood

Updated 2026-07-21 19:32:31 UTC by Dennis Underwood