

CrowdStrike Falcon Insight XDR Detailed EDR_XDR Solutions Overview

2026, 4th edition

Controlled attack simulation with full telemetry analysis
Environment: Windows 10, Windows 11, Windows Server + Active Directory



Evaluation period: March to June 2026
Last updated: June 10, 2026

Table of contents

01	Test summary
02	The importance of telemetry detail in the context of incidents
03	Protection model evolution
04	Scope, objectives and limitations of the 2026 edition
05	Certification model
06	Security feature availability
07	Agent Configuration and Operating Mode
08	Environment configuration
09	Operational Assessment – Phase 1
10	Operational Assessment – Phase 2
11	Responding to scenarios from Phase 2
12	Telemetry, correlation and incident visibility assessment based on Phase 2
13	Test conclusions

Test Summary

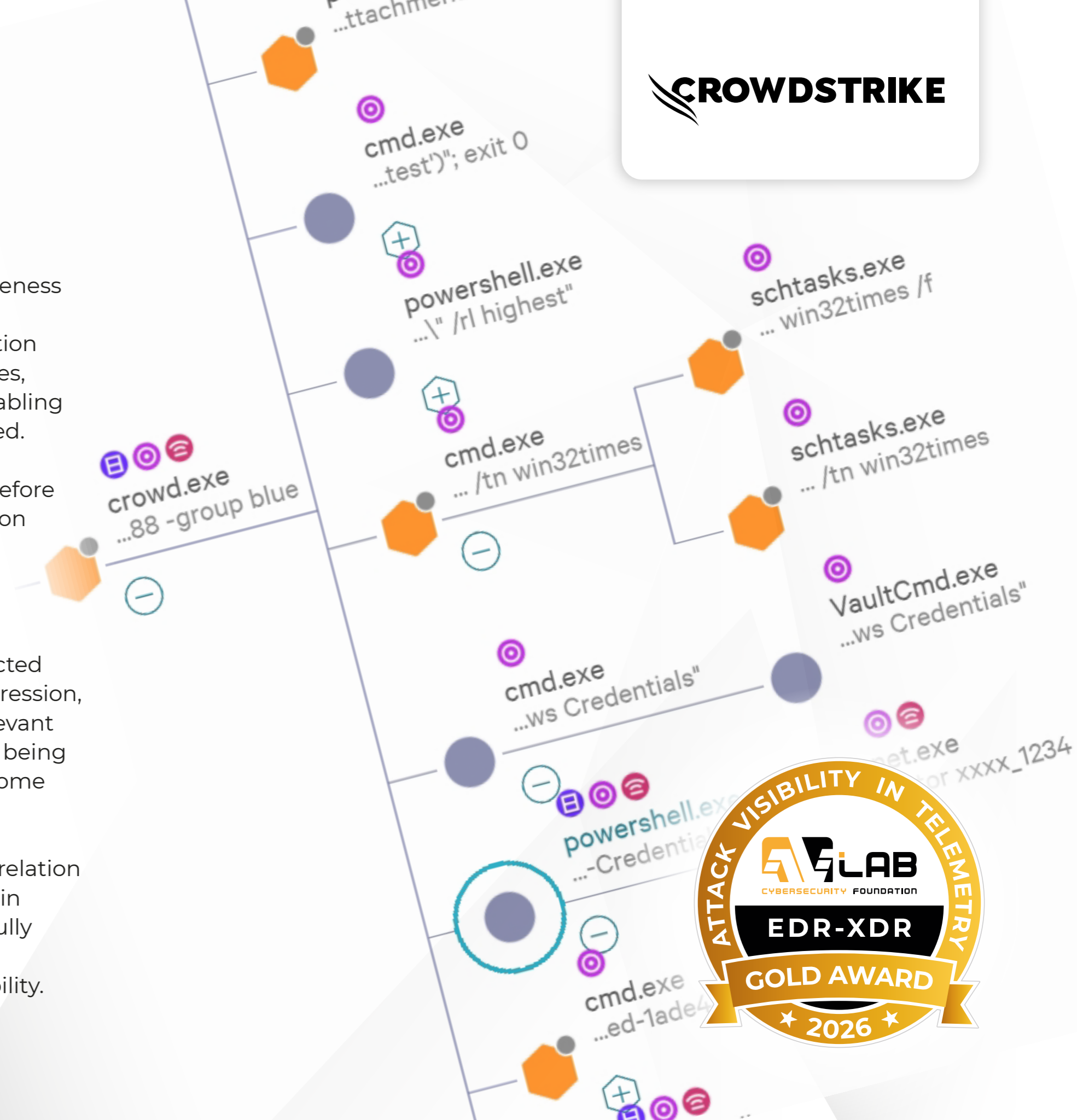
CrowdStrike Falcon Insight XDR

The solution provided a high level of visibility and detection effectiveness for most of the scenarios tested in Phase 2, including PowerShell, LOLBins, WebDAV, WMI, remote code execution, and post-exploitation activity. In many cases, detailed telemetry was available for processes, the command line, parent-child relationships, and user context, enabling analysis of the attack chain and identification of the techniques used.

The anti-malware engine effectively blocked some activities even before the payload was executed, providing good visibility into the execution chain and behavioral detections. Partial visibility into network communication and inter-host activity was also available, though cross-host correlation was not complete in all scenarios.

The limitations mainly concerned telemetry inconsistencies in selected cases. In one scenario involving an AI-assisted workflow, data compression, and exfiltration via HTTP, the solution did not generate alerts or relevant process, network, or command-line telemetry, despite the scenario being successfully executed. This affected the ability to fully reconstruct some activities and assess the completeness of the telemetry.

In summary, the solution demonstrated advanced visibility and correlation capabilities across most tested scenarios. However, inconsistencies in telemetry coverage, including one scenario that executed successfully without generating meaningful detections or telemetry, should be considered when assessing the overall completeness of attack visibility.

**CROWDSTRIKE**

FULL

The function works in its full scope without significant limitations.

PARTIAL

The function works, but with noticeable limitations.

LIMITED

The function is present but provides minimal visibility or analytical value.

NONE

The function is unavailable or no relevant visibility was observed.

Attack Description

Simulation of data retrieval via DNS TXT, reconstruction of the payload in %TEMP%, execution via LOLBin, and exfiltration.

PARTIAL

Copying the payload between PCs before execution, then running the RAT as an EXE and quietly capturing keystrokes.

FULL

User downloads ISO, mounts it, and executes EXE, triggering payload and C2 communication (HTTP/mTLS).

PARTIAL

Payload is delivered via WebDAV and executed by the user, establishing C2. The compromise is then extended to another endpoint through SMB file transfer and remote execution via WMI.

FULL

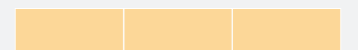
The attacker copies the payload to a remote host via SMB and executes it using PsExec as a SYSTEM service, thereby achieving remote code execution.

FULL

Using a local LLM model to dynamically select subsequent steps. After verifying SMB connectivity to the target host, remote code is executed via WMI, which creates, compiles, and runs the payload on the victim's system, generating multi-stage process telemetry.

FULL

The attack uses an external AI model as a decision-making layer to control actions on the host. The script collects system information, saves it to a file, compresses it, and sends it externally via HTTP, while the AI issues subsequent decisions based on the system's state.

NONE

Clicking a phishing link launches mshta, which downloads and executes PowerShell. The script collects system data, establishes persistence via a Scheduled Task, runs rundll32 to hide its activity, and then exfiltrates data over HTTPS (curl). The entire chain uses LOLBIN.

FULL

The importance of telemetry detail in the context of incidents

Modern attacks, including long-term activities carried out by advanced threat groups (APTs), are rarely limited to a single incident. They often begin with a seemingly harmless phishing message, which is actually only the first stage of an extensive chain of activities involving maintaining access, escalating privileges, so-called lateral movement, and data exfiltration [1].

In such scenarios, it is not only the detection of a single alert that is crucial, but also the ability to record and correlate all relevant technical artifacts. Recording even partial information about potential incidents allows for the reconstruction of events that took place in the analyzed environment. EDR-XDR solutions that monitor systems and applications, thanks to data correlation and automation mechanisms, support security teams in identifying the relationships between the stages of an attack.

Based on the collected telemetry, it is possible to determine what actions the attacker took, in what order, using what processes, applications, and user accounts. In the case of attacks spread over time, it is particularly important to maintain a consistent chronology of events and visibility of changes in user context and permissions. This information can be presented in the form of a logical or graphical reconstruction of the incident (e.g., as a process tree or a map of connections between hosts), which will certainly facilitate understanding of the full course of the operation.

Another important element of a mature EDR-XDR platform is the ability to perform advanced queries, allowing analysts to manually search raw telemetry, build their own queries, and verify investigative hypotheses. In the case of multi-stage campaigns, a ready-made alert often does not reflect the full extent of the compromise. Only in-depth analysis allows for the identification of additional traces of the breach, connections between systems, and the attacker's actual goal.

The broader the range of monitored events and the greater the depth of telemetry, the greater the organization's ability to understand the attacker's intentions and techniques, and consequently to mitigate the impact of the incident and adapt security policies to real threats.



[1] See an example of an APT attack on our editorial office in 2026:

<https://avlab.pl/przypadek-falszywego-phishingu-to-element-dlugofalowego-ataku-grupy-storm-1679/>



Protection model evolution

Product categories such as EPP, EDR, XDR, and SIEM are increasingly overlapping in terms of functionality. In practice, the differences between them no longer stem from commercial nomenclature, but from the scope of telemetry collected, the method of correlating it, the level of automation, and the ability to reconstruct multi-stage attack chains.

Many modern solutions referred to as EDR have expanded their capabilities to include integration with SaaS services, identity systems, and selected network sources, bringing them closer functionally to the XDR class. At the same time, some products positioned as XDR still rely primarily on endpoint telemetry, offering limited cross-domain correlation. This means that the product name does not always reflect its actual level of operational maturity.

The table below is for organizational and illustrative purposes only. It presents the typical characteristics of each class of solutions, assuming that specific implementations may go beyond this framework or combine elements of several categories.

EPP

Focuses primarily on prevention. Signature and reputation-based blocking with limited telemetry storage and minimal investigative context.

XDR

Enables cross-domain correlation based on endpoint telemetry. Aggregates and correlates signals from endpoints, identity providers, SaaS platforms, email, and network sources. Focuses on reconstructing attack chains across multiple systems.

EDR

Endpoint-focused detection and investigation. Provides detailed telemetry (process trees, command lines, artifacts), retrospective analysis, and host-level response actions. Correlation is primarily limited to data generated by endpoints.

SIEM

Log aggregation and correlation engine based on rules or behaviors. Data normalization, long-term storage, compliance-related use cases, and configurable detection logic. Detection quality depends largely on log quality, integration level, and rule maturity.



The models are simplified and do not fully reflect market dynamics. The historical evolution from EPP through EDR to XDR shows a shift in emphasis from signature-based prevention to deep telemetry, event correlation, and incident reconstruction. Of course, the protection of workstations and servers remains the unchanging core, but it is the range of data sources, the quality of correlation, and the possibility of multi-system analysis that today determine the real effectiveness of IT environment protection.

Scope, objectives and limitations of the 2026 edition

The purpose of the test is to evaluate the actual capabilities of EDR-XDR solutions in detecting, recording, and correlating multi-stage attacks under controlled laboratory conditions.

The analysis is not limited to the generation of alerts. The depth of telemetry, the quality of event correlation, the ability to reconstruct the attack chain, and the operational usefulness of data from the perspective of a SOC analyst are also evaluated.

Methodological assumptions

The test is carried out in a structured environment simulating a realistic attack scenario covering the stages from Initial Access to Exfiltration and Impact, in accordance with selected MITRE ATT&CK techniques.

Each stage is performed in a controlled and repeatable manner, with accurate time recording and predefined expected technical artifacts.

In the first phase, the test can be performed in “report-only” mode to assess visibility and correlation without interrupting the scenario. In subsequent stages, the effectiveness of response and automatic prevention mechanisms can be analyzed.

What exactly are we evaluating?

The test answers the following questions:

01

Does the solution generate a clear alert for the techniques used?

?

02

Does it provide complete and detailed event telemetry?

?

03

Does it enable correlation of events within a host and between systems?

?

04

Does it allow reconstruction of the attack chain in the logical context of the incident?

?

05

Is the data provided operationally useful from the SOC team's perspective?

?

Test limitations

The test is conducted in a controlled environment and does not reflect the full complexity of production environments involving hundreds or thousands of endpoints, non-standard configurations, integrations with external systems, and actual load.

The attack scenarios, while realistic, are selected examples and do not cover all possible threat variants.

The results should be interpreted as an assessment of the technical capabilities of the solution under precisely defined test conditions. The test ensures repeatability and comparability of results but does not constitute a complete simulation of a large-scale production environment.



The 2026 edition introduces a clear distinction between:

- ✓ technology detection alone,
- ✓ full telemetry visibility,
- ✓ event correlation,
- ✓ attack chain reconstruction.

In the previous edition, these elements were evaluated together. The updated methodology separates detection effectiveness from the quality of the analytical context and the operational capabilities of the tested solution.

The comparison criteria have also been standardized, including:

- ✓ telemetry completeness,
- ✓ local and inter-host correlation,
- ✓ incident presentation consistency in the console,
- ✓ coverage of tested techniques.

The purpose of the changes is to increase the transparency of the methodology and reduce discretionary elements in the final assessment.

Certification model



Level 1 – Core Telemetry Visibility

CERTIFIED 2026

Awarded to solutions that provide full technical visibility of events within tested scenarios, including:

- 1 Generation of unambiguous security alerts,
- 2 Access to full telemetry (command-line, process relationships, file and registry changes, user context),
- 3 Consistent and accurate event chronology,
- 4 Automatic correlation of events within a single host.

Level 1 confirms that the solution provides sufficient technical visibility of security events at the host level, including clear alerts, key execution artifacts and contextual information required to analyze an incident within a single system.

Level 2 – Full Attack Chain Correlation

GOLD AWARD 2026

Level 2 is awarded to solutions that, in addition to meeting all Level 1 requirements, demonstrate the capability to automatically correlate events across multiple hosts and reconstruct a multi-stage attack chain within a single logical incident.








Solutions that do not provide basic attack visibility in the tested scenarios do not receive certification.

Certification is a summary of a technical assessment based on a defined methodology and uniform comparative criteria. The level awarded confirms that specific requirements for visibility, telemetry, and event correlation have been met.







Security features availability




1. Detection & Telemetry depth

Basic attack visibility	FULL 
Full attack telemetry	FULL 
Process tree visibility	FULL 
Command-line visibility	FULL 
Network - file - registry telemetry	FULL 
MITRE technique mapping	FULL 
Offline detection capability	FULL 






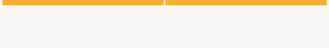
2. Event Correlation & Attack Context

Single-host correlation	FULL 
Cross-host correlation	FULL 
Full attack-chain reconstruction	FULL 
Graphical attack visualization	FULL 




3. Threat Intelligence & Enrichment

Suspicious object intelligence (IP, URL, SHA)	FULL 
External enrichment (reputation feeds, threat intelligence, VirusTotal-like services)	FULL 
Approximate file reputation scoring	FULL 



4. Incident Response Capabilities

Workstation isolation	FULL 
File & process containment (quarantine, kill, blocking, isolation)	FULL 
Sandbox or deep file analysis (manual or automated detonation; local or cloud)	FULL 
Proposed remediation guidance	FULL 
Data recovery capability (rollback or backup)	FULL 
Automation or SOAR (native or external)	FULL 






5. Investigation & Hunting

Advanced query capability	FULL 
Raw telemetry access	PARTIAL 
Timeline analysis	FULL 

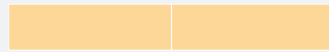
6. Security Posture & Exposure Visibility

Graphical security posture visualization (vulnerabilities, weak passwords, misconfiguration)	FULL 
Agent configuration validation	FULL 

7. Platform & Administrative Controls

Updates management	FULL 
Granular administrative access control	FULL 
Admin console protection (MFA, SSO, audit log)	FULL 
API availability	FULL 
Multi-tenant console for MSSP (multi-company management)	FULL 

8. AI-Assisted Operations

AI assistance in console (alert summarization, query generation, recommendations etc.)	NONE 
---	--

Agent Configuration and Operating Mode

The test covered EDR-XDR solution operating in a Windows environment, in accordance with a predefined attack scenario and a uniform assessment methodology.

Most of the tested EDR-XDR platforms include an integrated antivirus module. This module remained active so that the environment configuration would reflect a real production deployment that an administrator might use in an organization.

In the first phase of testing, the solutions could be run in report-only mode, without automatic blocking and remediation. This was done to:

- ✓ enable full execution of the attack scenario,
- ✓ assess telemetry visibility,
- ✓ analyze the quality of correlation and incident reconstruction,
- ✓ avoid interrupting the attack chain at an early stage.

The configuration of agent policies was based on default settings, with a possible extension of the scope of telemetry collected. For solutions requiring manual policy configuration, settings were used to maximize event visibility and technical artifact logging, while maintaining compliance with the manufacturer's official documentation.

The aim of the study was not to test non-standard experimental configurations, but to evaluate the real capabilities of the platform in a production scenario.

Environment configuration

The tests were conducted in a controlled virtual environment, including a separate attack infrastructure and victim systems with EDR-XDR agents installed.

The environment included:

01

a separate server simulating the attack infrastructure (Command-and-Control)

02

virtual machines running Windows 10 and Windows 11 with the tested agents installed

03

optionally, a domain controller (Active Directory) to simulate lateral movement and identity abuse scenarios

The victim systems had a standard operating system configuration with up-to-date security patches and full network access in accordance with the test scenario.

The attack scenarios were carried out using controlled simulation techniques that mirrored selected MITRE ATT&CK techniques. Depending on the test phase, adversary emulation tools and native system mechanisms were used to replicate the attacker's behavior as realistically as possible.

The test did not include social engineering elements (e.g., a real phishing campaign) because the goal was to technically replicate behavior at the host and infrastructure level, not to test user susceptibility to manipulation.

The attacks were carried out in a controlled and repeatable manner, without conducting full campaigns from start to finish. Each stage was performed according to a predefined scenario and documented in terms of expected technical artifacts.



Attacker Infrastructure
(Isolated Zone)



C2 Server - Kali Linux
(Command & Control)



Attack Tools & Frameworks

- Atomic Red Team
- Custom Scripts
- Other Simulations



Payload Delivery
(HTTP, SMB, RDP, etc.)



Internal Network
(Lab Environment)



Active Directory



DC - Windows Server
EDR/XDR Agent



Host A - Windows 10
EDR/XDR Agent



Host B - Windows 11
EDR/XDR Agent



Internet Acces
(Allowed)

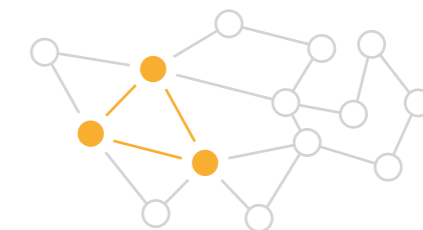


EDR/XDR Console & Evaluation



Central Console
(Alerting, Investigation, Hunting)

MITRE ATTACK Mapping



Attack Chain Reconstruction

Attack Stages

Initial Access

Executions

Persistence

Defense Evasion

Credential Access

Lateral Movement

Exfiltration

Operational Assessment – Phase 1

Settings applied: Telemetry Only (Detect Mode)

Attack Scenario – Adversary Emulation (MITRE ATT&CK via Caldera Framework)

FULL



The function works in its full scope without significant limitations.

PARTIAL






The function works, but with noticeable limitations.



LIMITED



The function is present but provides minimal visibility or analytical value.

Initial Access	T1566.001	Download Macro-Enabled Phishing Attachment	FULL 	Single host	<ul style="list-style-type: none"> ✓ Security alert generated ✓ Full network telemetry ✓ Including parent-child relationships ✓ Full telemetry event visibility ✓ Command line visibility ✓ User context visibility ✓ Host visibility ✓ ATT&CK mapping correctly
Execution	T1106	Execute process via Win32 API (Process.Start)	FULL 	Single host	<ul style="list-style-type: none"> ✗ No security alert generated ✗ No ATT&CK mapping ✓ Full network telemetry ✓ Including parent-child relationships ✓ Full telemetry event visibility ✓ Command line visibility ✓ User context visibility ✓ Host visibility

<p>Persistence</p>	<p>T1566.001</p>	<p>Creating persistent access by Scheduled Task after reboot</p>	<p>FULL</p> 	<p>Single host</p>	<ul style="list-style-type: none"> ✗ No security alert generated ✗ No ATT&CK mapping ✓ Full network telemetry ✓ Including parent-child relationships ✓ Full telemetry event visibility ✓ Command line visibility ✓ User context visibility ✓ Host visibility
<p>Defense Evasion</p>	<p>T1036.004</p>	<p>Masquerading via scheduled task name (win32times)</p>	<p>FULL</p> 	<p>Single host</p>	<ul style="list-style-type: none"> ✓ Security alert generated ✓ Including parent-child relationships ✓ Full telemetry event visibility ✓ Command line visibility ✓ User context visibility ✓ Host visibility ✓ ATT&CK mapping correctly
<p>Credential Access</p>	<p>T1555.004</p>	<p>Enumerate stored credentials (Windows Credential Manager)</p>	<p>FULL</p> 	<p>Single host</p>	<ul style="list-style-type: none"> ✗ No security alert generated ✗ No ATT&CK mapping ✓ Including parent-child relationships ✓ Full telemetry event visibility ✓ Command line visibility ✓ User context visibility ✓ Host visibility

<p>Lateral Movement</p>	<p>T1021.002</p>	<p>SMB mount, file transfer, and remote execution via PowerShell</p>	<p>FULL</p> 	<p>Full cross-host</p>	<ul style="list-style-type: none"> ✗ No security alert generated ✓ Network telemetry ✓ Including parent-child relationships ✓ Lateral movement visibility ✓ Full telemetry event visibility ✗ No ATT&CK mapping ✓ Command line visibility ✓ User context visibility
<p>Exfiltration</p>	<p>T1048.002</p>	<p>Upload file via HTTPS using curl to external service (file.io)</p>	<p>FULL</p> 	<p>Single host</p>	<ul style="list-style-type: none"> ✓ Security alert generated ✓ Network telemetry ✓ Including parent-child relationships ✓ Full telemetry event visibility ◐ ATT&CK mapping ✓ Command line visibility ✓ User context visibility ✓ Host visibility ✓ Exfiltration visibility

Operational Assessment – Phase 2

Settings applied: Default Protection (Windows policy)

Methodological assumptions

The results of each attack scenario are interpreted from two operational perspectives: the attacker and the Security Operation Center. This approach helps to determine not only whether the attack was technically successful, but also how clearly it was visible and understandable from the defender's point of view.



Attacker's perspective

From the attacker's point of view, the key issue is whether the individual techniques were executed according to the planned scenario and whether the chain of attacks could be carried out in whole or in part.

- 1 Was the payload successfully executed?
- 2 Was communication or control of the host established (if applicable)?
- 3 Was it possible to maintain or extend access (e.g., lateral movement or exfiltration)?
- 4 At what stage was the attack chain interrupted if it was stopped by defensive mechanisms?

This perspective helps determine whether the solution actively disrupts the attack or merely logs the activity.



SOC-Defender

From a security team perspective, the key factors are the quality and completeness of the information presented in the security console and whether the incident can be quickly understood and investigated. From this perspective, we assess:

- 1 Was a clear security alert generated?
- 2 Is there sufficient telemetry data available to analyze the event?
- 3 Were related events automatically correlated?
- 4 Can the attack chain be reconstructed in the context of a single incident?
- 5 Does the analysis require manual correlation of events?

The tables below summarize the operational results of each attack scenario using simplified visual indicators for both perspectives.

Responding to scenarios from Phase 2

Custom Attacks Scenarios

FULL



The function works in its full scope without significant limitations.

PARTIAL



The function works, but with noticeable limitations.

LIMITED



The function is present but provides minimal visibility or analytical value.



NONE


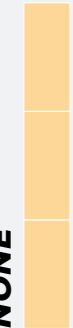



The function is unavailable or no relevant visibility was observed.

Attack Description	Used tools	Stages	Attack's MITRE ID	Attacker's Perspective	Security Operation Center Perspective	Assessment
Simulation of data retrieval via DNS TXT, reconstruction of the payload in %TEMP%, execution via LOLBin, and exfiltration.	Custom + PowerShell script	<ol style="list-style-type: none"> Initial Access Execution Defense Evasion Command & Control Exfiltration 	T1059.001 T1071.004 T1105 T1218.009 T1036.005 T1564.001 T1041	Downloading DNS chunks, delivering the payload, execution, and successful HTTP communication (status code 200).	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Security alert generated <input checked="" type="checkbox"/> Automatic blocking <input type="checkbox"/> Network telemetry <input checked="" type="checkbox"/> Including parent-child relationships <input type="checkbox"/> Partial telemetry event visibility <input type="checkbox"/> Partial ATT&CK mapping <input type="checkbox"/> Partial command line visibility <input checked="" type="checkbox"/> User context visibility 	PARTIAL

Attack Description	Used tools	Stages	Attack's MITRE ID	Attacker's Perspective	Security Operation Center Perspective	Assessment
Copying the payload between PCs before execution, then running the RAT as an EXE and quietly capturing keystrokes.	DuplexSpy	<ol style="list-style-type: none"> Initial Access Lateral Movement Execution Persistence Credential Access Collection Command and Control Exfiltration 	<p>T1204.002</p> <p>T1056.001</p> <p>T1113</p> <p>T1071.001</p> <p>T1041</p> <p>T1547.001</p> <p>T1082</p>	Execute the RAT as an EXE to quietly capture keystrokes. There are no exploits, it has a low profile and there is periodic exfiltration to C2.	<ul style="list-style-type: none"> ✔ Security alert generated ✔ Automatic blocking 🟡 Network telemetry ✔ Including parent-child relationships ✔ Full telemetry event visibility 🟡 Partial ATT&CK mapping ✔ Command line visibility ✔ User context visibility ✔ Host visibility ✔ Lateral movement visibility 	<p>FULL</p>
User downloads ISO, mounts it, and executes EXE, triggering payload and C2 communication (HTTP/mTLS).	Sliver Server	<ol style="list-style-type: none"> Initial Access Execution Defense Evasion Command and Control Lateral Movement Discovery Collection Exfiltration 	<p>T1566.001</p> <p>T1204.002</p> <p>T1036.005</p> <p>T1071.001</p> <p>T1021.002</p> <p>T1041</p>	The attacker delivers ISO, victim mounts it and runs EXE, establishing payload execution and C2.	<ul style="list-style-type: none"> 🟡 Security alert generated 🟡 Automatic blocking ✔ Network telemetry ✔ Including parent-child relationships 🟡 Partial telemetry event visibility 🟡 Partial ATT&CK mapping 🟡 Command line visibility ✔ User context visibility ✔ Host visibility ✔ C2 visibility 🟡 Lack of full response action visibility 	<p>PARTIAL</p>

Attack Description	Used tools	Stages	Attack's MITRE ID	Attacker's Perspective	Security Operation Center Perspective	Assessment
<p>Payload is delivered via WebDAV and executed by the user, establishing C2. The compromise is then extended to another endpoint through SMB file transfer and remote execution via WMI.</p>	Sliver Server	<ol style="list-style-type: none"> Initial Access Execution Command and Control Lateral Movement Exfiltration 	<p>T1105 T1204.002 T1071.001 T1021.002 T1047 T1041</p>	<p>The attacker delivers payload via WebDAV, gains C2, then uses SMB and WMI to execute payload on another endpoint.</p>	<ul style="list-style-type: none"> ✔ Security alert generated ✔ Automatic blocking ✔ Including parent-child relationships ✔ Command line visibility ✔ User context visibility ✔ Host visibility ✔ Lateral movement visibility 	<p>FULL</p> 
<p>The attacker copies the payload to a remote host via SMB and executes it remotely using WMI (Win32_Process), thereby achieving remote code execution.</p>	Custom + PsExec + Atomic Red Team	<ol style="list-style-type: none"> Lateral Movement Execution 	<p>T1021.002 T1047 T1570</p>	<p>The attacker uses administrator credentials to access the administrative share, transfers the payload to a remote host, and executes it remotely via WMI (Win32_Process), achieving remote code execution.</p>	<ul style="list-style-type: none"> ✔ Security alert generated ✔ Automatic blocking ✔ Network telemetry ✔ Including parent-child relationships ✔ User context visibility ✔ SMB activity visibility ✔ Cross-host correlation visibility 	<p>FULL</p> 

Attack Description	Used tools	Stages	Attack's MITRE ID	Attacker's Perspective	Security Operation Center Perspective	Assessment
Using a local LLM model to dynamically select subsequent steps. After verifying SMB connectivity to the target host, remote code is executed via WMI, which creates, compiles, and runs the payload on the victim's system, generating multi-stage process telemetry.	Ollama AI + Custom PowerShell	1. Discovery 2. Execution 3. Lateral movement	T1059.001 T1046 T1021.002 T1047	Collect the host's context and pass it to a local LLM, which selects the most effective lateral movement step. Then use SMB to gain access to the target system and remotely execute code via WMI.	<ul style="list-style-type: none"> ✔ Security alert generated ✔ Automatic blocking ✔ Including parent-child relationships ✔ Full telemetry event visibility ✔ Command line visibility ✔ User context visibility ✔ Lateral movement visibility ✔ Cross-host correlation ✔ Remote execution visibility 	FULL 
The attack uses an external AI model as a decision-making layer to control actions on the host. The script collects system information, saves it to a file, compresses it, and sends it externally via HTTP, while the AI issues subsequent decisions based on the system's state.	OpenAI (API) + PowerShell	1. Reconnaissance 2. Command & Control (AI) 3. Collection 4. Staging 5. Exfiltration	T1082 T1518 T1071.001 T1005 T1560 T1041 T1567	The attacker launches a simple loader that communicates with the AI model and executes its commands. It delegates the analysis of the environment and the selection of actions to LLM, which decides on the next steps based on the data it receives.	<ul style="list-style-type: none"> ✘ Detection ✘ Prevention ✘ Telemetry visibility ✘ Exfiltration visibility ✘ Command line visibility <p>The scenario was executed successfully, but it did not appear in either the detections or the RAW telemetry (Advanced Event Search).</p>	NONE 
Clicking a phishing link launches mshta, which downloads and executes PowerShell. The script collects system data, establishes persistence via a Scheduled Task, runs rundll32 to hide its activity, and then exfiltrates data over HTTPS (curl). The entire chain uses LOLBIN.	Kali Linux + browser + powershell + schtasks + rundll32 + curl	1. Initial Access 2. Execution (LOLBIN) Execution (PowerShell) 3. Collection 4. Persistence 5. Defense Evasion 6. Execution (LOLBIN) 7. Exfiltration	T1566.002 T1218 T1059.001 T1005 T1053.005 T1036.005 T1218 T1041	I use phishing to launch mshta and bypass standard detection mechanisms. The HTA loads PowerShell in the background, which collects data and maintains access via a Scheduled Task. I use only legitimate system tools (LOLBIN) to minimize detection, and I send the data externally via HTTPS.	<ul style="list-style-type: none"> ✔ Security alert generated ✔ Automatic blocking ✔ Including parent-child relationships ✔ Full telemetry event visibility ✔ ATT&CK mapping ✔ Command line visibility ✔ User context visibility ✔ Host visibility ✔ LOLBin visibility 	FULL 

Telemetry, correlation and incident visibility assessment based on Phase 2

The table summarizes the observations derived from all executed attack scenarios in phase 2 and presents the overall assessment of telemetry visibility, event correlation and incident reconstruction capabilities of the evaluated solution.

✓ requirement for obtaining Level 1 or Level 2 Certification

✗ no requirements

FULL



The function works in its full scope without significant limitations.

PARTIAL








The function works, but with noticeable limitations.






LIMITED



The function is present but provides minimal visibility or analytical value.

Attack Description	Required for at least Level 1 Certification	Required for Level 2 Certification	Assessment	Comment
Basic attack visibility	✓	✓	PARTIAL 	The platform detected and blocked most tested attack scenarios in Phase 2, including PowerShell abuse, LOLBins, WebDAV delivery, remote execution, and multiple post-exploitation techniques. However, at least one scenario involving AI-assisted workflow, data collection, compression, and HTTP exfiltration was not detected and produced no meaningful telemetry or RAW event visibility, which prevented consistent attack visibility across all tested scenarios.
Full attack telemetry	✗	✗	PARTIAL 	The platform provided detailed telemetry for most scenarios, including command-line operations, parent-child relationships, user context, and process and PowerShell telemetry. However, in some scenarios, the telemetry was incomplete or unavailable, which limited the ability to fully reconstruct the attack.
Parent-Child Process Visibility	✓	✓	PARTIAL 	The platform provided visibility into parent-child process relationships for a variety of scenarios involving PowerShell, mshta, WMI/DCOM, and LOLBins. However, in some scenarios, telemetry for processes and execution chain relationships was incomplete or unavailable.

Attack Description	Required for at least Level 1 Certification	Required for Level 2 Certification	Assessment	Comment
Command-line visibility	✓	✓	PARTIAL 	<p>The platform provided command-line visibility for many scenarios. However, in some scenarios, command-line telemetry was incomplete or completely unavailable.</p>
User context visibility	✓	✓	FULL 	<p>The platform consistently provided visibility into the user context, including usernames, domains, and login types for the tested scenarios.</p>
Timestamp integrity	✓	✓	FULL 	<p>The platform provided consistent and detailed timestamps for processes, detections, and telemetry events, enabling an analysis of the attack timeline.</p>
Single-host correlation	✓	✓	FULL 	<p>The platform enabled the correlation of activity on a single host, including processes, PowerShell, LOLBins, parent-child relationships, and behavioral detections.</p>
Cross-host correlation	✗	✓	PARTIAL 	<p>Activity visibility between hosts was available for some scenarios involving SMB, WMI/DCOM, and network communication. However, there was no full source-to-destination correlation or automatic linking of attack chains between hosts.</p>

Attack Description	Required for at least Level 1 Certification	Required for Level 2 Certification	Assessment	Comment
Network telemetry	✘	✘	PARTIAL 	Partial network telemetry was available, covering selected connections and C2 activity. However, in some scenarios, there was a lack of full visibility into network communication and exfiltration activity.
File & registry telemetry	✘	✘	PARTIAL 	File operation visibility was available for some scenarios, including file creation, compression, and quarantine. However, file telemetry was inconsistent and did not cover all tested operations.
Remediation guidance	✘	✘	LIMITED 	Basic information about the security measures taken was available, but the platform did not provide detailed guidance on analysis, remediation, or follow-up response actions.
Graphical attack visualization	✘	✘	PARTIAL 	A partial visualization of the process chain and parent-child relationships was available. However, a complete graphical representation of the attack chain between hosts and scenarios was missing.
Advanced query capability	✘	✘	PARTIAL 	Advanced Event Search and RAW telemetry were available for investigation and threat hunting. However, telemetry was not consistently available across all tested scenarios, limiting analysis in selected cases.

Test conclusions

In this edition, the solution demonstrated a high level of effectiveness in detecting and blocking techniques used in multi-stage attack scenarios. Execution chain visibility, process telemetry, command-line visibility, and analysis of activity related to PowerShell, LOLBins, and remote code execution were particularly well-rated. In many scenarios, behavioral detections and detailed information enabling incident analysis were available.

Visibility into cross-host activity and network telemetry was partially available and allowed identification of selected elements of lateral movement and C2 communication. Limitations primarily concerned telemetry consistency across all tested scenarios. In selected post-exploitation and exfiltration scenarios, some activities were not recorded or were not readily accessible through detections, telemetry, or RAW events, reducing the ability to fully reconstruct attacker activity.

Overall, the solution provides a high level of visibility and protection for most tested techniques. However, the assessment identified gaps in telemetry coverage and incident correlation for specific attack workflows, indicating that visibility may vary depending on the techniques used and the nature of the attacker activity.



4.exe

Blocked Actions

Disk operations

DLL / Library Load 3

Filename	Time
\Device\HarddiskVolume3\Windows\System32\ntdll.dll	May 12, 2026 15:52:52
\Device\Mup\;WebDavRedirector\;Z:0000000000d452d\192.168.108.132@8005\DavWWWRoot\4.exe	May 12, 2026 15:52:52
\Device\Mup\192.168.108.132@8005\DavWWWRoot\4.exe	May 12, 2026 15:52:52

3 results (1-3 shown) Items per page 50 Page 1 of 1

Related entities

Devices 0 of 1 shown

Host	IP address	Platform
DESKTOP-V285N8Q	46.148.157.110	Windows

Detection - high

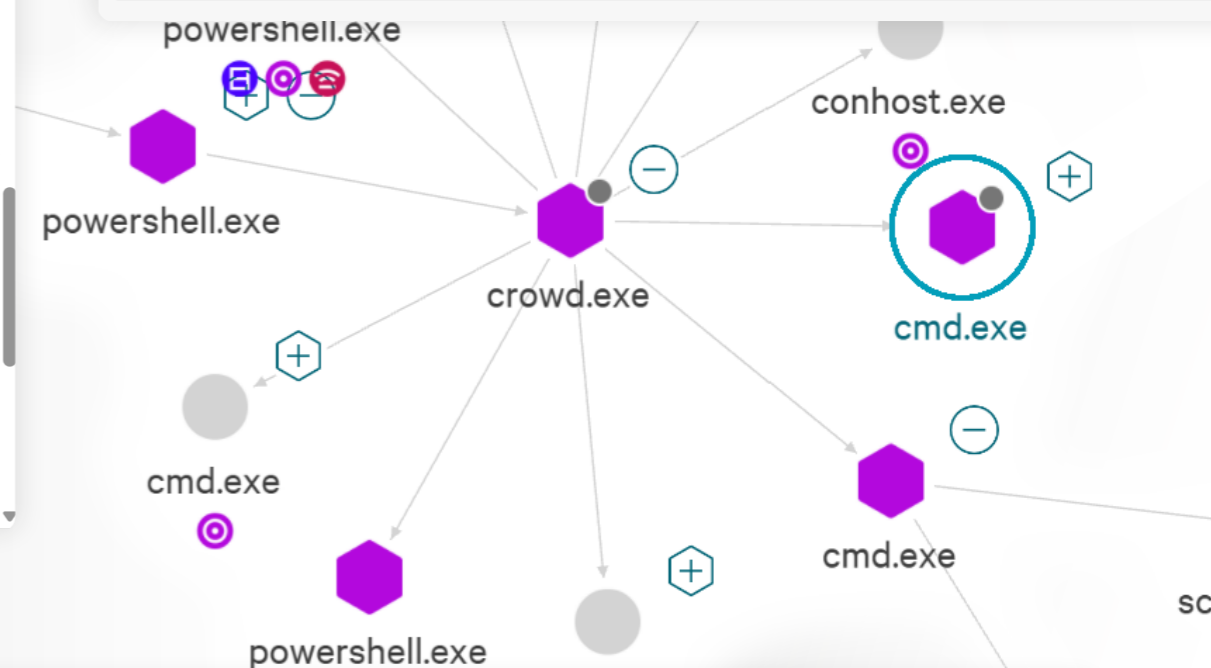
Command line

Show decoded On

```
powershell.exe -NoProfile -ExecutionPolicy Bypass -EncodedCommand [I0.File]:W('C:\Users\Public\6.cs', [Text.Encoding]::UTF8.GetString([Convert]::FromBase64String('U3lzdGVtOyB1c2l1ZyBTeXN0ZW0uRGlhZ25vc3RyY3M7IHVzaW5nIFN5c3R1bS5JTzsgdXNpbmcgU3GluZzsgY2xhc3MgUHJvZ3JhbSB7IHNOYXRpYyB2b2lkIE1haW4oKSB7IEZpbGUuV3JpdGVbbGxUZXhNcUHVibGljXDZfb2sudHh0IiwicmFuIik7IFByb2Nlc3MuU3RhcncQoImNhbGMuZXh1Iik7IFRocmVhbkTsgfSB9'))); & 'C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe' /target:C:\Users\Public\6.exe C:\Users\Public\6.cs; Start-Process 'C:\Users\Public\6.exe
```

File path

\Device\HarddiskVolume3\Windows\System32\WindowsPowerShell\v1.0\powershell.exe





To learn more about the collaboration, please visit the Attack Visibility in EDR-XDR Telemetry page, where you can also track the results of recent editions.

[CHECK OUR WEBSITE](#)

We also conduct tests of other types:



Advanced In-The-Wild Malware Test

We evaluate the effectiveness of protection software against threats in the wild, and reveal the true ability of the product to neutralize threats.



Test of modules to protect online banking

Antivirus solutions are subjected to scenarios of payment details theft, information manipulation by banking Trojans and other malware.



Cyber Transparency Audit

We verify information from Endpoint Protection software vendors for compliance with specific standards and the idea of transparency.

For more information, please visit our website...



The AVLab Cybersecurity Foundation is an independent organization dedicated to protecting privacy and security on the Internet. We are a member of AMTSO (Anti-Malware Testing Standards Organization), which works to improve the transparency, objectivity and quality of testing. Also, we are a member of MVI (Microsoft Virus Initiative) in this matter as well.

We build awareness of users in the field of digital protection. We issue opinions, technical analysis and tests of IT solutions in the field of cybersecurity. Our strongest assets include thorough and detailed reviews, preparation of reports related to privacy and endpoint protection, and in particular, security tests that make us recognizable all over the world as one of the most trusted and popular testing laboratories.

To learn more about other opportunities for cooperation, please refer to our full offer and contact us: kontakt@avlab.pl



www.avlab.pl