Your Windows Sensor Checklist

Securing Windows workloads in the cloud is different from securing laptops or desktops. Many solutions simply repurpose EDR agents designed for endpoint use cases — but EDR wasn't built for cloud-specific attack scenarios, which are layered, identity-driven, and often involve lateral movement.

Relying on rule engines and binary signatures means you'll frequently miss attacks where legitimate Windows utilities are abused for malicious purposes. When the activity looks normal but the behavior is not, visibility gaps form — and attackers take advantage.

This cheat sheet outlines what your Windows sensor must do to catch, investigate, and stop threats across the full Windows attack surface, including:

1. **What Your Windows Sensor Should Look At**
2. **Creating a Behavioral Baseline of Your Windows Environment with Anomaly Detection**
3. **Cross Correlating Cloud Signals with Sensor Data**

Use this as a checklist when evaluating sensors or briefing red teams.

1 What Your Windows Sensor Should Look At

Below are core Windows surfaces and exactly what your sensor should monitor, in addition to processes, files, and net flows.

☑ Registry

Description	How Attackers Use It
The registry is the central configuration database that defines service behavior, components, and system policies, which attackers can exploit.	Persistence (Run keys), service redirection, disabling security features, execution redirection via Run/RunOnce, scheduled-run markers.

☑ Signatures

Description	How Attackers Use It
Signatures include code hashes, certificates, and curated behavioral signatures.	Reuse signed-but-abused binaries, living-off-the-land binaries, or slightly modified tools to evade simple hash checks.

☑ DLL Anomalies

Description	How Attackers Use It
How Windows loads modules can be abused to run attacker code in trusted processes.	Search order hijacking, side-loading, file-less injection.

☑ Task Scheduler and Service Abuse

Description	How Attackers Use It
Scheduled tasks and services provide stealthy persistence and recurring execution.	Create tasks to run payloads on schedule, modify service ImagePath to point to malicious binary.

☑ PowerShell and Script Execution	
Description	**How Attackers Use It**
PowerShell is a powerful automation tool and a common attacker vector.	Encoded payloads, unrestricted execution policy, living-off-the-land scripts, lateral movement.

☑ Credentials & Hive Access (SAM, SECURITY)	
Description	**How Attackers Use It**
SAM and SECURITY hives contain sensitive hashes and secrets.	Export SAM/SECURITY hives to disk for offline cracking or lateral use.

2 Creating a Behavioral Baseline of Your Windows Environment with Anomaly Detection

Once you're monitoring the full attack surface, you need to understand what "normal" looks like. Attackers often bypass known detections by using legitimate tools to disable logs, inject threads into trusted processes, or create persistence in obscure registry paths.

By establishing a behavioral baseline for every host, user, and process, your sensor should detect anomalies even when no MITRE technique matches.

- Detects zero-day or previously unseen techniques.
- Flag deviations from standard execution flow or registry activity.
- Reduce false positives by filtering out expected system behavior.

This baseline-driven layer ensures you're protected from both known and unknown attack vectors.

3 Cross Correlating Cloud Signals with Sensor Data

Windows telemetry by itself tells you what happened on the machine but not who did it or what other assets were affected. Your solution should be able to correlate local system data with cloud-level context:

- **Identities:** Which user or service account triggered the process?
- **APIs & workloads:** What cloud resource or container was involved?
- **Network flow:** Did the process communicate with an external IP or internal API endpoint?

By linking OS signals with your broader CNAPP telemetry – identity, CDR, cloud logs, and Layer 7 API data – you gain end-to-end understanding of the incident:

- What identity initiated the attack?
- How did they move across your environment?
- What did they attempt to access or modify?

Try Sweet Security's Windows sensor and achieve a 2-5 min MTTR.

About Sweet

Sweet Security is redefining enterprise cloud protection. As the leading provider of Runtime CNAPP solutions and a pioneer in AI Security, Sweet unifies runtime context with advanced AI intelligence to protect the modern enterprise across applications, workloads, and cloud infrastructure. Its platform delivers real-time detection and response, vulnerability and posture management, identity threat protection, and API security–powered by patent-pending, LLM-driven detection, reducing alert noise to just 0.04%. By bridging cloud and AI security, Sweet enables organizations to accelerate innovation, reduce operational risk, and achieve industry-leading MTTR times.