What is a WMI event subscription?

A WMI event subscription is a persistence technique that uses Windows Management Instrumentation to run a payload when a chosen event occurs, a logon, a process start, or a time trigger. It is built from three WMI objects: an event filter (the trigger), a consumer (the action, often a command), and a binding that links them. Stored in the WMI repository as SYSTEM, it leaves no file in a normal autostart location, which makes it stealthy. It needs admin and maps to MITRE T1546.003.

HOW IT WORKS

01 The technique and payload

With admin, the attacker registers all three objects (via PowerShell or `wmic`):

- An `__EventFilter` querying for a trigger (logon, an interval, a process start).
- A `CommandLineEventConsumer` whose `CommandLineTemplate` runs the payload.
- A `__FilterToConsumerBinding` joining them.

WMI then runs the payload as SYSTEM on each trigger, with nothing in the usual autostart spots. This stealth and SYSTEM execution are why it is a favored advanced persistence. Documented for defensive context.

> NO FILE IN THE USUAL SPOTS
>
> *WMI subscriptions persist in the WMI repository, not run keys or Startup, and execute as SYSTEM. That stealth is exactly why defenders must enumerate WMI subscriptions specifically, not just the obvious autostart locations.*

HOW TO DEFEND

- Enumerate permanent WMI subscriptions regularly (filters, consumers, bindings) and baseline the legitimate ones.
- Enable WMI activity logging and alert on `CommandLineEventConsumer`/-`ActiveScriptEventConsumer` creation.
- Use Sysmon (events 19/20/21) to catch WMI subscription creation.
- Limit local admin, required to register subscriptions.
- Hunt for consumers running scripts or binaries from user-writable paths.

SOURCES

[1] MITRE ATT&CK: Event Triggered Execution (T1546)

[2] Microsoft: WMI events

[3] MITRE ATT&CK: Persistence (TA0003)

Find the backdoors an attacker would leave behind.

securelayer7.net/learn/persistence/what-is-a-wmi-event-subscription

Open online