Event Triggered Execution: Component Object Model Hijacking, Sub-technique T1546.015 - Enterprise

Archived: 2026-04-05 18:32:52 UTC

Adversaries may establish persistence by executing malicious content triggered by hijacked references to Component Object Model (COM) objects. COM is a system within Windows to enable interaction between software components through the operating system.[1] References to various COM objects are stored in the Registry.

Adversaries may use the COM system to insert malicious code that can be executed in place of legitimate software through hijacking the COM references and relationships as a means for persistence. Hijacking a COM object requires a change in the Registry to replace a reference to a legitimate system component which may cause that component to not work when executed. When that system component is executed through normal system operation the adversary's code will be executed instead.[2] An adversary is likely to hijack objects that are used frequently enough to maintain a consistent level of persistence, but are unlikely to break noticeable functionality within the system as to avoid system instability that could lead to detection.

One variation of COM hijacking involves abusing Type Libraries (TypeLibs), which provide metadata about COM objects, such as their interfaces and methods. Adversaries may modify Registry keys associated with TypeLibs to redirect legitimate COM object functionality to malicious scripts or payloads. Unlike traditional COM hijacking, which commonly uses local DLLs, this variation may leverage the "script:" moniker to execute remote scripts hosted on external servers.[3] This approach enables stealthy execution of code while maintaining persistence, as the remote payload would be automatically downloaded whenever the hijacked COM object is accessed.

Source: https://attack.mitre.org/techniques/T1546/015