DEEPARMOR®
by SPARKCOGNITION

Dispelling the Myths on Fileless Attacks

Fileless attacks are cybersecurity's new boogeyman. This type of cyber attack is increasingly dominating the conversation, with industry leaders and vendors hailing it as the new threat to watch for.

While there are increasing amounts of information being written about fileless attacks, a surprising percentage of it is misleading or incorrect. Many vendors are peddling misinformation on what fileless attacks are, how great a threat they pose, and how to protect against them. This paper will be dispelling those myths and laying out what is fact about fileless attacks and what is fiction.

Identifying Fileless Attacks

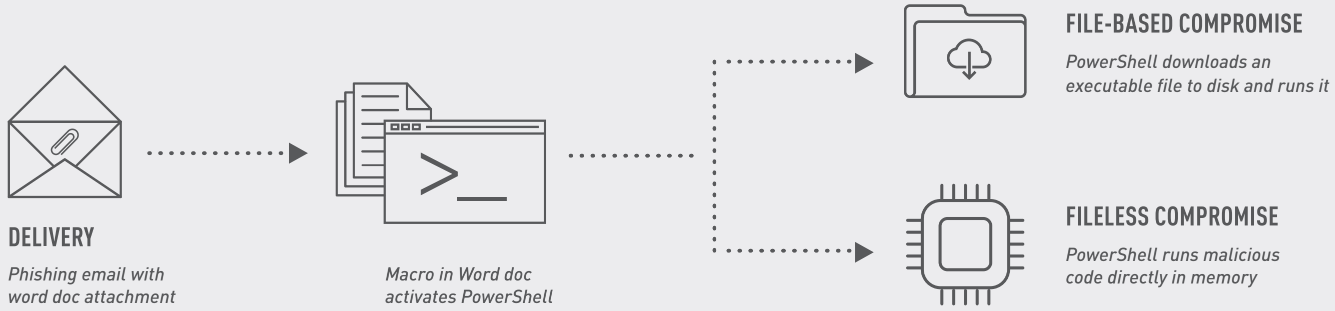
What are fileless attacks? This simple question is perhaps the greatest source of misinformation.

Fileless attacks are best characterized as "diskless," or attacks that never write to the disk. Instead, they're typically injected directly into memory and run from there. Because of this, they can be surprisingly short-lived. They exist only until the system is rebooted and then, because they never write their activity to the hard drive, they disappear. Hackers can use workarounds to ensure the longevity of the attack. Regardless, the defining feature of fileless attacks is that no file ever hits the disk—hence the name.

Many sources online miss this crucial trait and end up defining fileless attacks in inaccurate, self-contradictory ways. It's often claimed that fileless attacks work specifically by hijacking legitimate system tools or applications such as documents. However, with most document-based threats, a document is still written to the disk. This process contradicts the definition of fileless attacks, which exist exclusively as volatile, memory-based artifacts.

We've cataloged here legitimate varieties of fileless attacks:

- ***In-memory:*** *This is the most basic, fundamental form of fileless attacks. They're loaded and executed directly in system memory using exploits and code injection techniques.*
- ***Registry-resident:*** *The other main form of fileless attacks, this refers to malicious scripts stored in the registry rather than memory. Autorun registry entries are one technique used to continue running scripts on a system, even after a reboot. This is also a type of "living off the land" technique.*
- ***DLL injection:*** *Some fileless attacks use DLL injection as their attack vector. This attack runs malicious code within the address space of another process using a dynamic-link library. DLL injections can hijack legitimate processes and force them to behave in unintended ways.*
- ***SQL injection:*** *SQL injection is an attack vector that targets databases using the SQL programming language. It inserts malicious SQL statements into an entry field for execution.*
- ***Script-based:*** *Script-based attacks can be file-based or fileless. Script-based fileless attacks use scripts to insert themselves into otherwise benign sites or applications to execute post-exploitation activities.*
- ***Macro-based:*** *Like script-based attacks, macro-based attacks can be file-based or fileless. Macro-based fileless attacks embed malicious code into Microsoft Office documents, which is then run by PowerShell directly in memory.*



DELIVERY
Phishing email with word doc attachment

Macro in Word doc activates PowerShell

FILE-BASED COMPROMISE
PowerShell downloads an executable file to disk and runs it

FILELESS COMPROMISE
PowerShell runs malicious code directly in memory

How Common are Fileless Attacks?

Because fileless attacks are often misidentified, reports on their prevalence vary wildly. But in general, fileless attacks are on the rise. While they're not the most common type of threat, organizations should be on guard.

According to the Ponemon Institute, 29% of attacks that organizations experienced in 2017 had a fileless component, representing a 20% increase from 2016. Chances are good that this number will continue to steadily rise, particularly as fileless attacks are dangerously effective when they do appear.

Protecting Against Fileless Attacks

While reports of the frequency of fileless attacks may sometimes be exaggerated, the difficulties they pose for antivirus are not. Fileless attacks are estimated to be roughly ten times more likely to succeed than their file-based counterparts. The numbers bear out this claim: Despite accounting for about a third of all attacks in 2017, fileless attacks represented 77% of successful attacks.

This effectiveness is in how tricky fileless attacks are to detect and block. Because there's no file written to the hard drive, there's no signature for antivirus to detect. System memory—the only place on the computer where the threat resides—cannot be scanned using heuristics.

The static file-scanning approach used against malware by legacy and next-generation antivirus is insufficient against fileless attacks. In addition, legacy antivirus solutions use rudimentary signatures, heuristics, and predefined YARA rules, none of which are effective at detecting new fileless attacks. Next-generation antivirus are typically hybrids of legacy approaches and artificial intelligence (AI), but they also rely on signatures, heuristics, and rules. This evolved platform may catch fileless attacks they have seen before but won't help against novel threats. Someone has to write those rules, and the time that takes creates a protection gap between when a new threat is launched—fileless or otherwise—and when rules- or heuristics-based approaches can detect it.

TECHNIQUES USED AGAINST FILELESS ATTACKS	
Legacy Antivirus	**Indicators of compromise (IOC)** *Reactive. Ineffective against unknown fileless threats*
Next-Generation Antivirus	**Heuristics** **Behavior analysis** *Same problems as legacy antivirus.* *Even vendors that otherwise use AI employ legacy techniques against fileless attacks*
Cutting-Edge Antivirus	**Machine learning** *Reliably detects and blocks unknown fileless attacks before execution*

Cybersecurity built entirely from AI is best positioned to identify and detect new fileless attacks. AI and machine learning platforms don't employ signatures or heuristics. Instead, they generalize what a threat looks like and continue to learn and update themselves over time. This capability makes it the best match for more difficult cyber threats such as fileless attacks.

Despite this advantage, many next-generation vendors that do understand the unmatched power of AI and employ it against malware fall back on legacy techniques for fileless threats. When considering cybersecurity for your organization, ask vendors specific questions on their approach to fileless attacks. Do they actually use their AI technology against fileless attacks? Or are they relying on heuristics, rules, and signatures?

Legacy and many forms of next-generation protection are ineffective at detecting new fileless attacks because they're reactive and capable onlyof discovering threats that are known and understood. This shortcoming even extends to relatively modern approaches such as behavior analysis, which employs machine learning, but only post-infection. This won't help keep systems from being infected by fileless attacks in the first place. By contrast, approaches built completely from AI and machine learning take a predictive approach to detecting and preventing attacks, thus enabling them to protect against unknown threats. This type of protection can analyze code injected into the memory space, allowing it to uncover fileless attacks before they breach a system.

Any trend must be analyzed critically, and the recent wave of press about fileless attacks is no different. Organizations should protect themselves against fileless attacks, but that doesn't mean they should believe everything they read.