



LEADERSHIP FOR IT SECURITY & PRIVACY ACROSS HHS

HHS CYBERSECURITY PROGRAM

OFFICE OF INFORMATION SECURITY



Evasive Methods Against Healthcare

12/10/2020



- Detection Methods
- Fileless Malware
- Living off the Land
- MITRE | ATT&CK
- WMI
- Example Campaigns
- Remediation
- Summary
- References



Slides Key:



Non-Technical: Managerial, strategic and high-level (general audience)



Technical: Tactical / IOCs; requiring in-depth knowledge (sysadmins, IRT)



Signature Based

Signature-based detection relies on a preprogrammed list of known indicators of compromise (IOCs). An IOC could include malicious network attack behavior, content of email subject lines, file hashes, known byte sequences, or malicious domains. Signatures may also include alerts on network traffic, including known malicious IP addresses that are attempting to access a system.




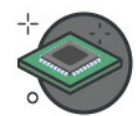




Anomaly Based

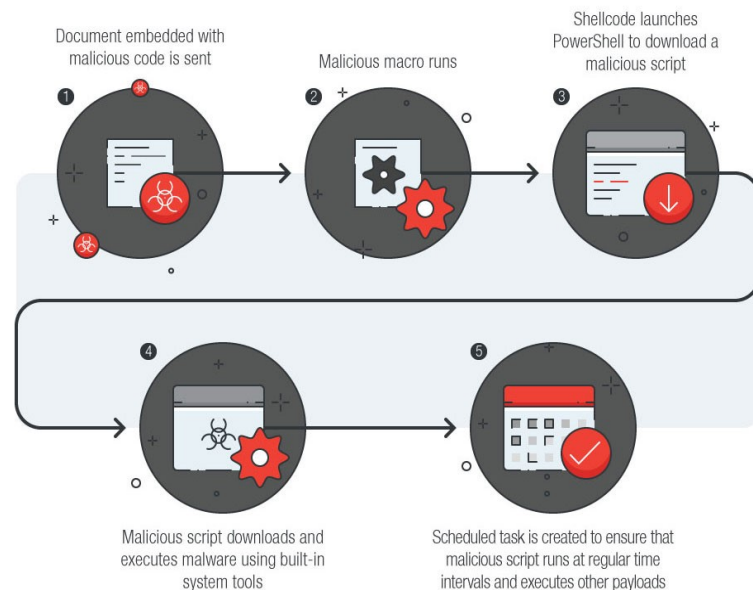
Anomaly-based detection is used for changes in behavior. Anomaly-based detection relies upon observing network occurrences and discerning anomalous traffic through heuristics and statistics.





Fileless Threats 101: Characteristics of a Fileless Attack

-  Has no identifiable code or signature and particular behavior that traditional security software detects.
-  Is a memory-based threat, resides in the computer's RAM. 
-  Takes advantage of processes in the system to facilitate an attack.
-  Could be used with other kinds of malware.
-  Could bypass whitelisting, as it takes advantage of allowed applications in the system.



LIVING OFF THE LAND

Image sources: TrendMicro



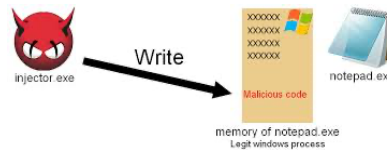
Windows Registry Manipulation

Windows registry manipulation involves the use of a malicious file or link that, when clicked on, uses a normal Windows process to write and execute fileless code into the registry.



Memory Code Injection

Memory code injection techniques involve hiding malicious code in the memory of legitimate applications. While processes that are critical to Windows activity are running, this malware distributes and reinjects itself into these processes.



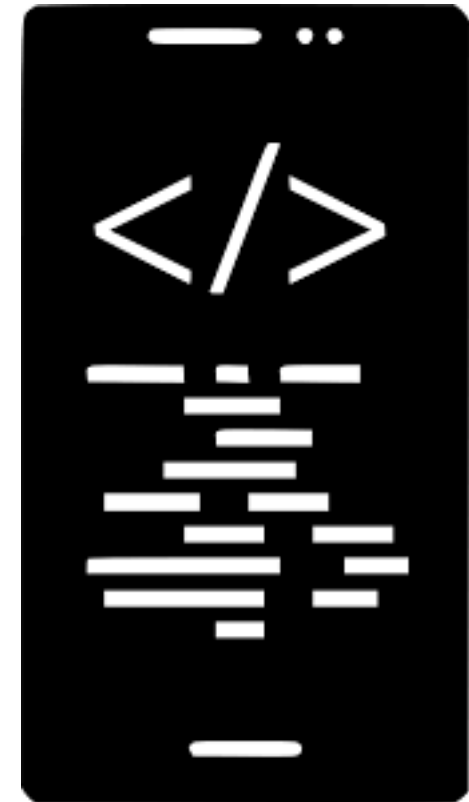
Script-Based Techniques

Scripts provide initial access, enable evasion, and facilitate lateral movements post-infection. Attackers will use scripts directly on the machine or embed them in Office documents and PDFs sent to the victims as email attachments.

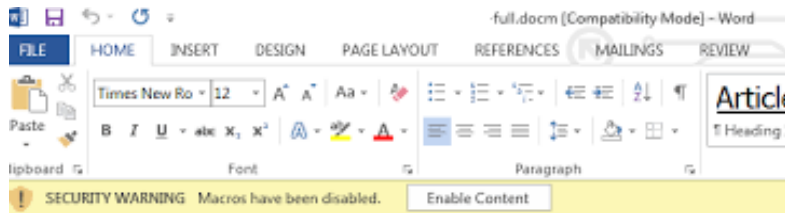
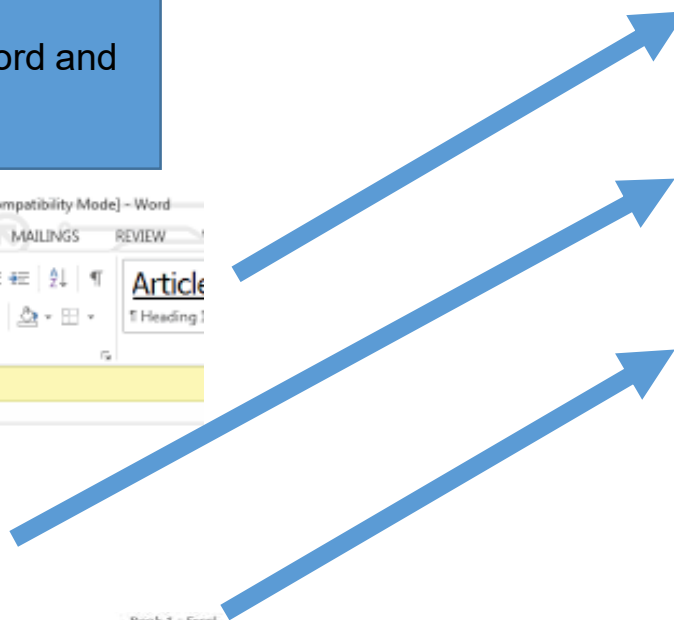




A **fileless attack** can start “traditionally” via malicious macro code (e.g. JavaScript or VBScript) embedded in archives, other seemingly normal files, and approved applications such as Office documents (e.g. Microsoft Word and Excel) and PDFs.



JScript | VBScript | Flash





...any binary supplied by the operating system that is normally used for legitimate purposes but can also be abused by malicious actors.

- powershell.exe
- bitsadmin.exe
- certutil.exe
- psexec.exe
- wmic.exe
- mshta.exe
- mofcomp.exe
- cmstp.exe
- windbg.exe
- cdb.exe
- msbuild.exe
- csc.exe
- regsvr32.exe

Most of the LoLBin and LoLBas techniques make use of PowerShell commands that execute a script directly in memory.

Can be used to easily install fake certificates for man-in-the-middle (MITM) attacks, and to download base64 or hexadecimal encoded files disguised as certificates before decoding them.

Is often used as part of the exploitation of a CVE or Office macro to download files in place of PowerShell.



Image sources: Github



GOAL

to document every binary, script and library that can be used for Living Off The Land techniques. [Github](#)

Submission Requirements

- executing code
- downloading/upload files
- bypass UAC
- compile code
- getting creds/dumping process
- surveillance (keylogger, network trace)
- evade logging/remove log entry
- side-loading/hijacking of DLL
- pass-through execution of other programs, script (via a LOLBin)
- pass-through persistence utilizing existing LOLBin
- persistence (Hide data in ADS, execute at logon etc)





Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command And Control
10 Items	31 Items	28 Items	59 Items	20 Items	17 Items	13 Items	9 Items	21 Items		
Drive-by Compromise Exploit Public-Facing Application	AppleScript CMSTP	.bash_profile and .bashrc Accessibility Features	Access Token Manipulation Accessibility Features	Access Token Manipulation Binary Padding	Account Manipulation Bash History	Account Discovery Application Window Discovery	AppleScript Application Deployment Software	Audio Capture Automated Collection	Automated Exfiltration Data Compressed	Commonly Used Port Communication Through Removable Media
Hardware Additions Replication Through Removable Media	Command-Line Interface Control Panel Items	AppCert DLLs AppCert DLLs	AppCert DLLs AppCert DLLs	BITS Jobs Bypass User Account Control	Brute Force Credential Dumping	Browser Bookmark Discovery File and Directory	Distributed Component Object Model Exploitation of Remote Services	Clipboard Data Data from Information Repositories	Data Encrypted Data Transfer Size Limits	Connection Proxy Custom Command and Control Protocol
Speerphishing Attachment Speerphishing Link Speerphishing via Service	Execution through API Execution through Module Load	Authentication Package BITS Job	Authentication Package DLL Search Order Hijacking	Bypass User Account Control DLL Search Order Hijacking	Credentials in Files Credential Dumping	File and Directory Logon Scripts	Exploitation of Remote Services Data from Network Shared Drive	Data from Local System Data from Network Shared Drive	Exfiltration Over Protocol Exfiltration Over Command and Control Channel	Custom Cryptographic Protocol Data Encoding
Supply Chain Compromise Trusted Relationship Valid Accounts	Graphical User Interface Installer Launchd Local Job Scheduling	Change Default File Association Component Object Model Hijacking	Dylib Hijacking Privilege Escalation	Dylib Hijacking Privilege Escalation	Extra Window Memory Injection File System Permissions Weakness	File System Permissions Weakness Hooking	Remote Desktop Protocol Remote File Copy Remote Services	Man in the Browser Screen Capture Video Capture	Exfiltration Over Other Network Medium Fallback Channels Multi-hop Proxy	Domain Fronting Fallback Channels Multi-Stage Channels
	LSASS Driver Mhta PowerShell Regsvcs/Regasm Regsvr32	Create Account DLL Search Order Hijacking Dylib Hijacking External Remote Services	File System Permissions Weakness Hooking Image File Execution Options Injection Launch Daemon	File System Permissions Weakness Hooking Image File Execution Options Injection Launch Daemon	File System Permissions Weakness Hooking Image File Execution Options Injection Launch Daemon	File System Permissions Weakness Hooking Image File Execution Options Injection Launch Daemon	Replication Through Removable Media Shared Webroot SSH Hijacking Taint Shared Content Third-party Software Windows Admin Shares	Screen Capture Video Capture	Scheduled Transfer	Multiband Communication Multilayer Encryption Port Knocking Remote Access Tools Remote File Copy Standard Application Layer Protocol Standard Cryptographic Protocol Standard Non-Application Layer Protocol Uncommonly Used Port Web Service



An open framework and knowledge base of adversary tactics and techniques based on real-world observations, provides a structured method to help you answer these questions.

MITRE | ATT&CK®



- Matrices
 - Tactics ▾
 - Techniques ▾
 - Mitigations ▾
 - Groups
 - Software
- Resources ▾
 - Blog
 - Contribute
 -



Enterprise	T1059	.001	Command and Scripting Interpreter: PowerShell	TAS05 has used PowerShell to download and execute malware and reconnaissance scripts. ^{[1][2][3][4]}
		.005	Command and Scripting Interpreter: Visual Basic	TAS05 has used VBS for code execution. ^{[1][2][3][4]}
		.007	Command and Scripting Interpreter: JavaScript/JScript	TAS05 has used JavaScript for code execution. ^{[1][2][3]}
		.003	Command and Scripting Interpreter: Windows Command Shell	TAS05 has executed commands using cmd.exe. ^[5]





There are many advantages of using WMI to an attacker:

- It is installed and running by default on all Windows operating systems.
- For code execution, it offers a stealthier alternative to running psexec.
- Permanent WMI event subscriptions run as SYSTEM.
- Defenders are generally unaware of WMI as a multi-purpose attack vector.
- Nearly every operating system action is capable of triggering a WMI event.
- Other than storage in the WMI repository, no payloads touch disk.

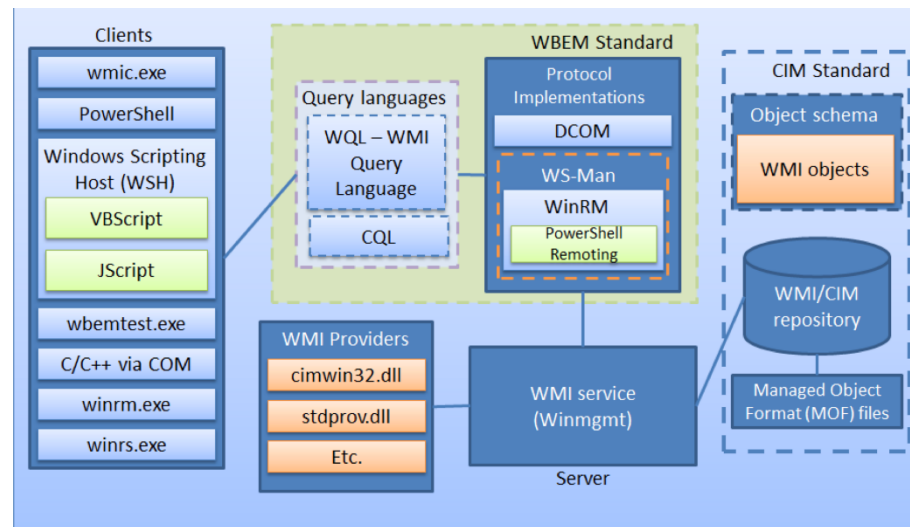


Image source:BlackHat

Great Source → <https://www.blackhat.com/docs/us-15/materials/us-15-Graeber-Abusing-Windows-Management-Instrumentation-WMI-To-Build-A-Persistent%20Asynchronous-And-Fileless-Backdoor-wp.pdf>





STUXnet

Stuxnet, one of the most sophisticated worms of 2010, affected nuclear processing facilities in Natanz, Iran and used WMI to enumerate users and spread to available network shares. It also used MOF (Managed Object Format) files, the means for creating and registering providers and events for WMI.



<https://www.youtube.com/watch?v=0SjMgnGwpq8>

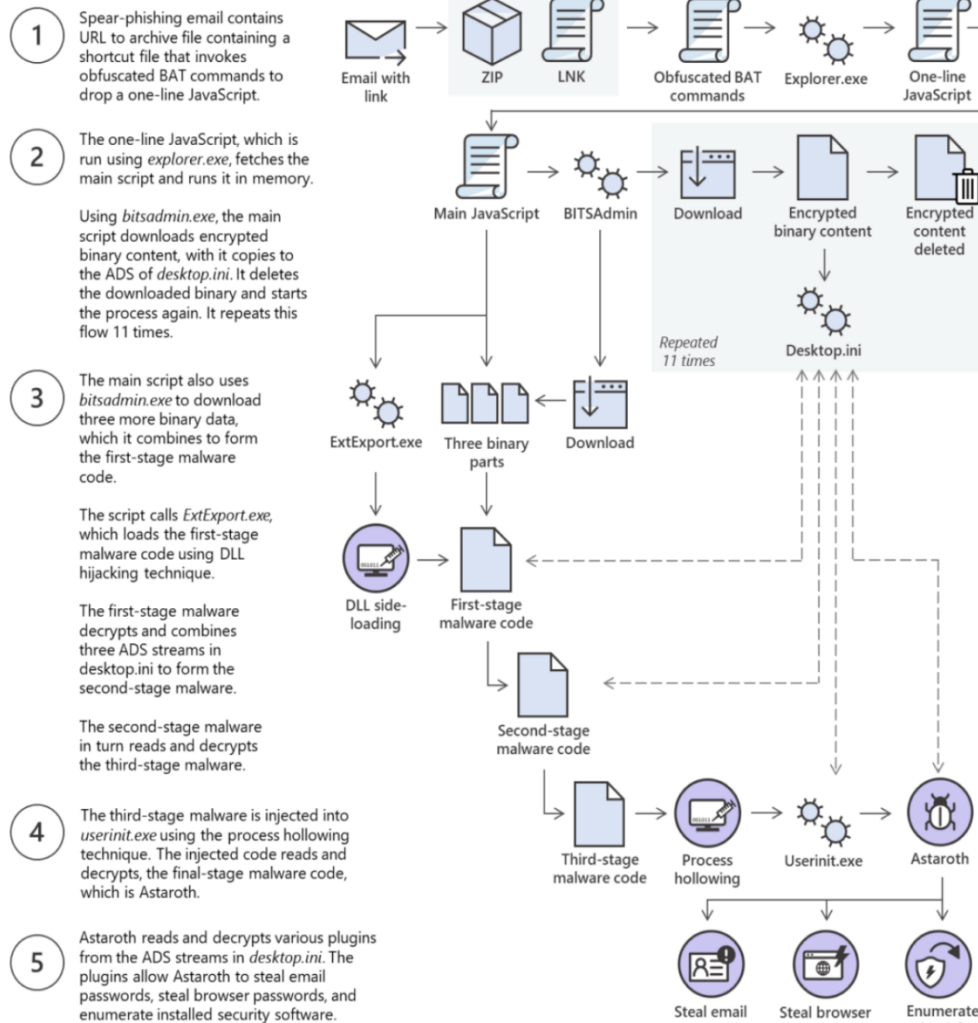




Astaroth

Fileless campaign that completely lives off the land. All of the relevant functionalities reside in scripts and shellcodes that are almost always coming in encrypted, and run while only in memory. No malicious executable is ever written to the disk.

Astaroth attack chain 2020



MITRE ATT&CK

T1192 – Spearphishing Link
 T1023 – Shortcut Modification
 T1064 – Scripting
 T1027 – Obfuscated Files or Information

T1064 – Scripting
 T1027 – Obfuscated Files or Information
 T1197 – BITS Jobs
 T1105 – Remote File Copy
 T1096 – NTFS File Attributes

TA0005 – Defense Evasion
 T1073 – DLL Side-Loading
 T1218 – Signed Binary Proxy Execution

T1129 – Execution Through Module Load
 T1140 – Deobfuscate/Decode Files or Information
 T1093 – Process Hollowing
 T1055 – Process Injection

T1503 – Credentials from Web Browsers
 T1003 – Credential Dumping

Image source: Microsoft



Nodersok

1. Victim runs an infected HTA (HTML application) file via an infected ad or download.
2. JavaScript code in the HTA file downloads a second-stage component.
3. Second-stage component launches a PowerShell command.
4. PowerShell commands download and run additional encrypted components

Nodersok Tools

Node.exe: This is an implementation of Windows' Node.js framework that is used by many web applications. This means it will go over the heads of most and slip detection.

WinDivert: Packet capture utility and would not raise any red flags regarding detection.

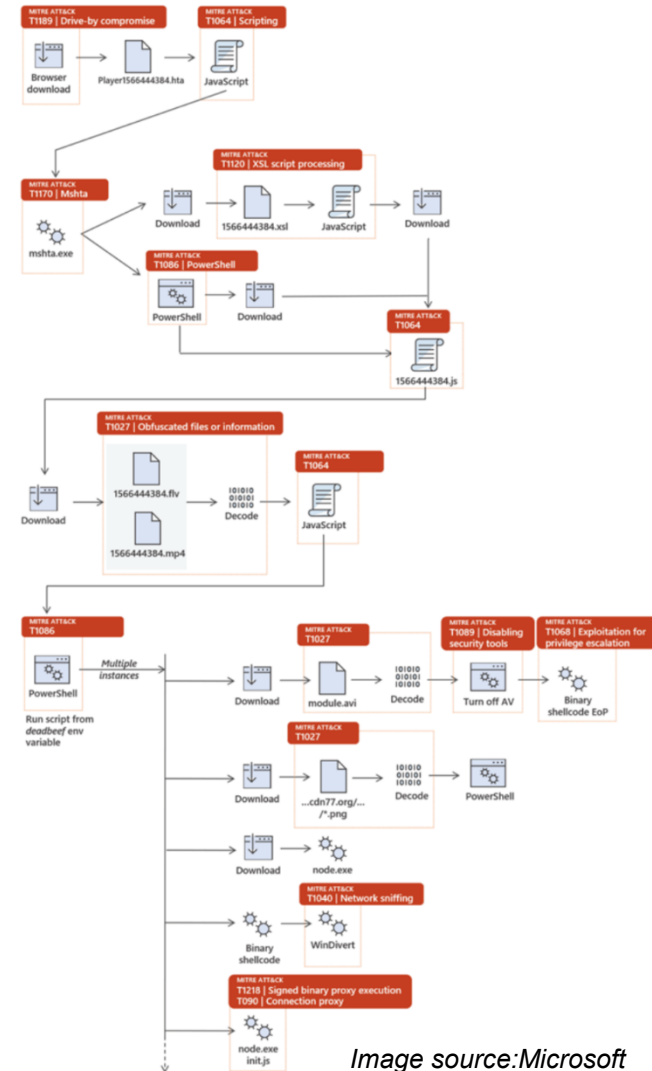


Image source: Microsoft



Zloader is spread through aggressive email campaigns where the email contains a malicious attachment with a provocative title referring to either COVID-19 or seeking a job, and invoices with links to infected Microsoft Word files. In the case of the invoice email, users will download the malware installer when they click the “Enable Content” button on the document.

Altered integers with ASCII characters to comprise a script to download malicious DLLs.

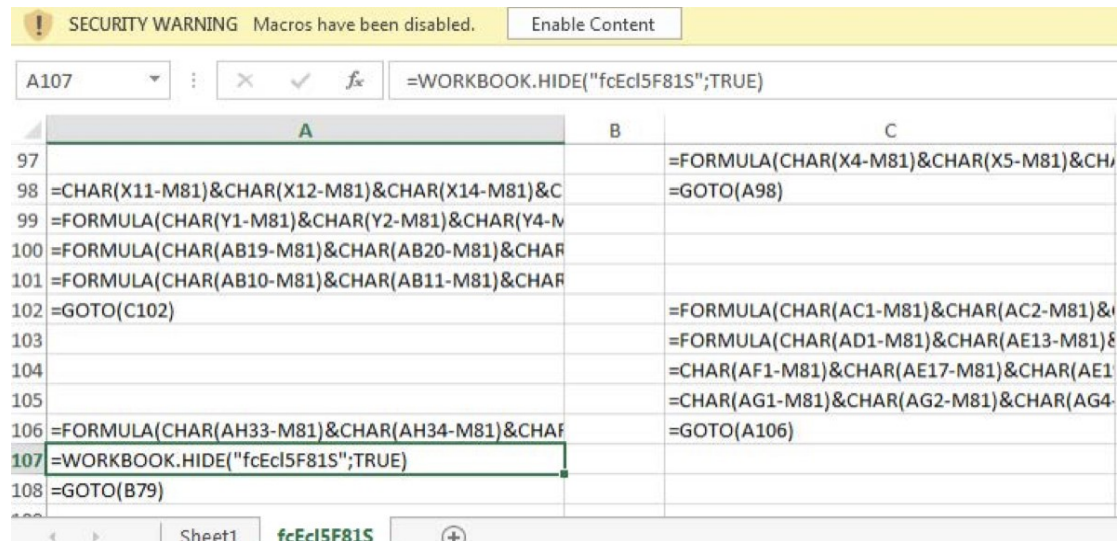
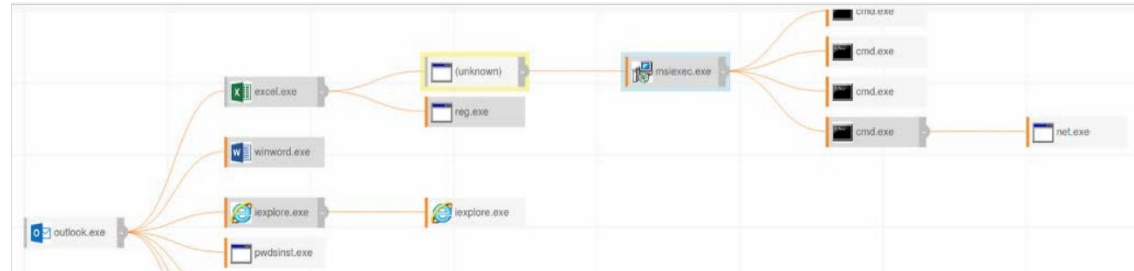


Image sources: eSentire and VMWare Carbon Black



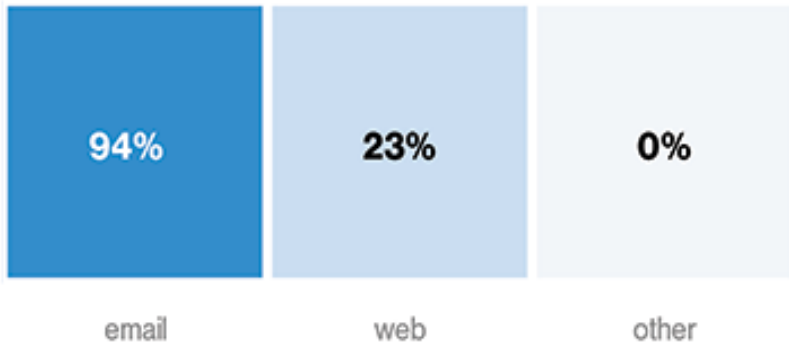
Signature + Anomaly (Behavior) Detection

- **Endpoint hardening:** Since fileless attacks originate at the endpoint, it's important to have an advanced endpoint security solution that provides essential defenses like vulnerability assessment, exploit/memory protection, desktop firewall, and URL filtering.
- **Machine learning:** Security analytic solution using advanced, adaptive, and state-of-the-art machine learning, deep learning, and artificial intelligence techniques.
- **Application containment:** Blacklisting solution that blocks unauthorized applications and code from running on servers, desktops, and fixed-function devices.
- **Behavior monitoring:** Anomaly detection and customized rules.
- **Interactive threat hunting:** Endpoint detection and response (EDR) tool that automatically and proactively investigates and responds to abnormal behavior on endpoints and searches for fileless attack footholds.
- **Single-console centralized management:** Security management through platform that provides control, visibility, reporting, and actionable dashboards across hundreds and even thousands of nodes enterprise-wide.
- **Integration with partner technologies:** Third-party partners who offer additional advanced technologies, helping you gain the advantage over adversaries.



According to 2019 Data Breach Investigations Report, it takes on average of 6 months to identify a compromise.

Delivery Method



File Type

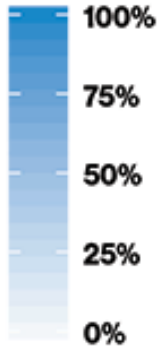
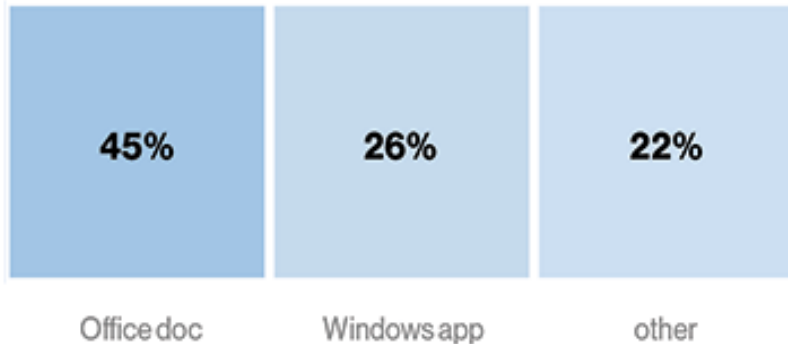


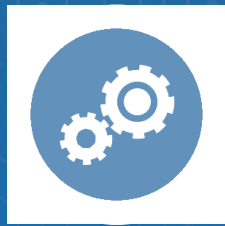
Image sources: Bluefin.com





Stealth is one of every threat actor's primary objectives, and fileless malware, LOLBins, and WMI functions provide perfect camouflage for malware that wants to hide in plain sight. This leaves plenty of time for an attacker to do their worst and maximize damage to the target network. We would do ourselves a disservice to think that these techniques aren't being utilized in the Healthcare sector. In order to truly protect our systems we have to utilize both signature and behavior detection methods.





Reference Materials



- Bring your own LOLBin: Multi-stage, fileless Nodersok campaign delivers rare Node.js-based malware
 - <https://www.microsoft.com/security/blog/2019/09/26/bring-your-own-lolbin-multi-stage-fileless-nodersok-campaign-delivers-rare-node-js-based-malware/>
- Dismantling a fileless campaign: Microsoft Defender ATP's Antivirus exposes Astaroth attack
 - <https://www.microsoft.com/security/blog/2019/07/08/dismantling-a-fileless-campaign-microsoft-defender-atp-next-gen-protection-exposes-astaroth-attack/>
- InfoStealers Weaponizing COVID-19
 - <https://www.lastline.com/labsblog/infostealers-weaponizing-covid-19/>
- New TA505 Phishing Campaign Using LOLBins to Distribute Backdoor Malware
 - <https://securityintelligence.com/news/new-ta505-phishing-campaign-using-lolbins-to-distribute-backdoor-malware/>
- What Are Living Off the Land Attacks?
 - <https://logrhythm.com/blog/what-are-living-off-the-land-attacks/>
- Living Off The Land Binaries and Scripts (and now also Libraries)
 - [ht https://github.com/api0cradle/LOLBAS](https://github.com/api0cradle/LOLBAS)
- What Are LOLBins and How Do Attackers Use Them in Fileless Attacks?
 - <https://www.cynet.com/attack-techniques-hands-on/what-are-lolbins-and-how-do-attackers-use-them-in-fileless-attacks/>
- Hunting for LoLBins
 - <https://blog.talosintelligence.com/2019/11/hunting-for-lolbins.html>
- How Do Attackers Use LOLBins In Fileless Attacks?
 - <https://www.sentinelone.com/blog/how-do-attackers-use-lolbins-in-fileless-attacks/>
- Cyber researchers confirm Russian government hack of Democratic National Committee
 - https://www.washingtonpost.com/world/national-security/cyber-researchers-confirm-russian-government-hack-of-democratic-national-committee/2016/06/20/e7375bc0-3719-11e6-9ccd-d6005beac8b3_story.html



- What Is Fileless Malware?
 - <https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware/what-is-fileless-malware.html>
- Monitoring with PowerShell: Preventing PowerShell based attacks (LoLBas)
 - <https://www.cyberdrain.com/monitoring-with-powershell-preventing-powershell-based-attacks-lolbas/>
- Keep An Eye on LOLBins
 - <https://isc.sans.edu/forums/diary/Keep+An+Eye+on+LOLBins/26502/>
- Abusing Microsoft Windows Utilities to Deliver Malware for Fun and Profit
 - <https://www.blackhat.com/docs/us-15/materials/us-15-Graeber-Abusing-Windows-Management-Instrumentation-WMI-To-Build-A-Persistent%20Asynchronous-And-Fileless-Backdoor-wp.pdf>
- An intro into abusing and identifying WMI Event Subscriptions for persistence
 - <https://in.security/an-intro-into-abusing-and-identifying-wmi-event-subscriptions-for-persistence/>
- Abusing Windows Management Instrumentation (WMI) to Build a Persistent, Asynchronous, and Fileless Backdoor
 - [ht https://github.com/api0cradle/LOLBAS](https://github.com/api0cradle/LOLBAS)
- ZLoader: What it is, how it works and how to prevent it | Malware spotlight
 - <https://resources.infosecinstitute.com/topic/zloader-what-it-is-how-it-works-and-how-to-prevent-it-malware-spotlight/>
- TA505
 - <https://attack.mitre.org/groups/G0092/>
- Latest Astaroth living-off-the-land attacks are even more invisible but not less observable
 - <https://www.microsoft.com/security/blog/2020/03/23/latest-astaroth-living-off-the-land-attacks-are-even-more-invisible-but-not-less-observable/>
- A survey of emerging threats in cybersecurity
 - <https://www.sciencedirect.com/science/article/pii/S0022000014000178>



- Threat Intel Versus Threat Hunting, What's the Difference?
 - <https://www.activecountermeasures.com/threat-intel-versus-threat-hunting-whats-the-difference/>
- Abusing Windows Management Instrumentation (WMI)
 - <https://www.youtube.com/watch?v=0SjMgnGwpq8>
- Malware spotlight: Nodersok
 - <https://resources.infosecinstitute.com/topic/malware-spotlight-nodersok/>
- What Is Endpoint Detection and Response (EDR)?
 - <https://www.mcafee.com/enterprise/en-us/security-awareness/endpoint/what-is-endpoint-detection-and-response.html>
- Attack Detection Fundamentals: Discovery and Lateral Movement - Lab #5
 - <https://labs.f-secure.com/blog/attack-detection-fundamentals-discovery-and-lateral-movement-lab-5/>
- How Do Attackers Use LOLBins In Fileless Attacks?
 - <https://www.sentinelone.com/blog/how-do-attackers-use-lolbins-in-fileless-attacks/>
- What Is Stuxnet?
 - <https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware/what-is-stuxnet.html>
- A Decade of WMI Abuse – an Overview of Techniques in Modern Malware
 - <https://www.bitdefender.com/files/News/CaseStudies/study/377/Bitdefender-Whitepaper-WMI-creat4871-en-EN-GenericUse.pdf>



Upcoming Briefs

Securing RFID In Healthcare

Product Evaluations

Recipients of this and other Healthcare Sector Cybersecurity Coordination Center products are highly encouraged to provide feedback to HC3@HHS.GOV.

Requests for Information

Need information on a specific cybersecurity topic? Send your request for information (RFI) to HC3@HHS.GOV or call us Monday-Friday, between 9am-5pm (EST), at **(202) 691-2110**.





Questions