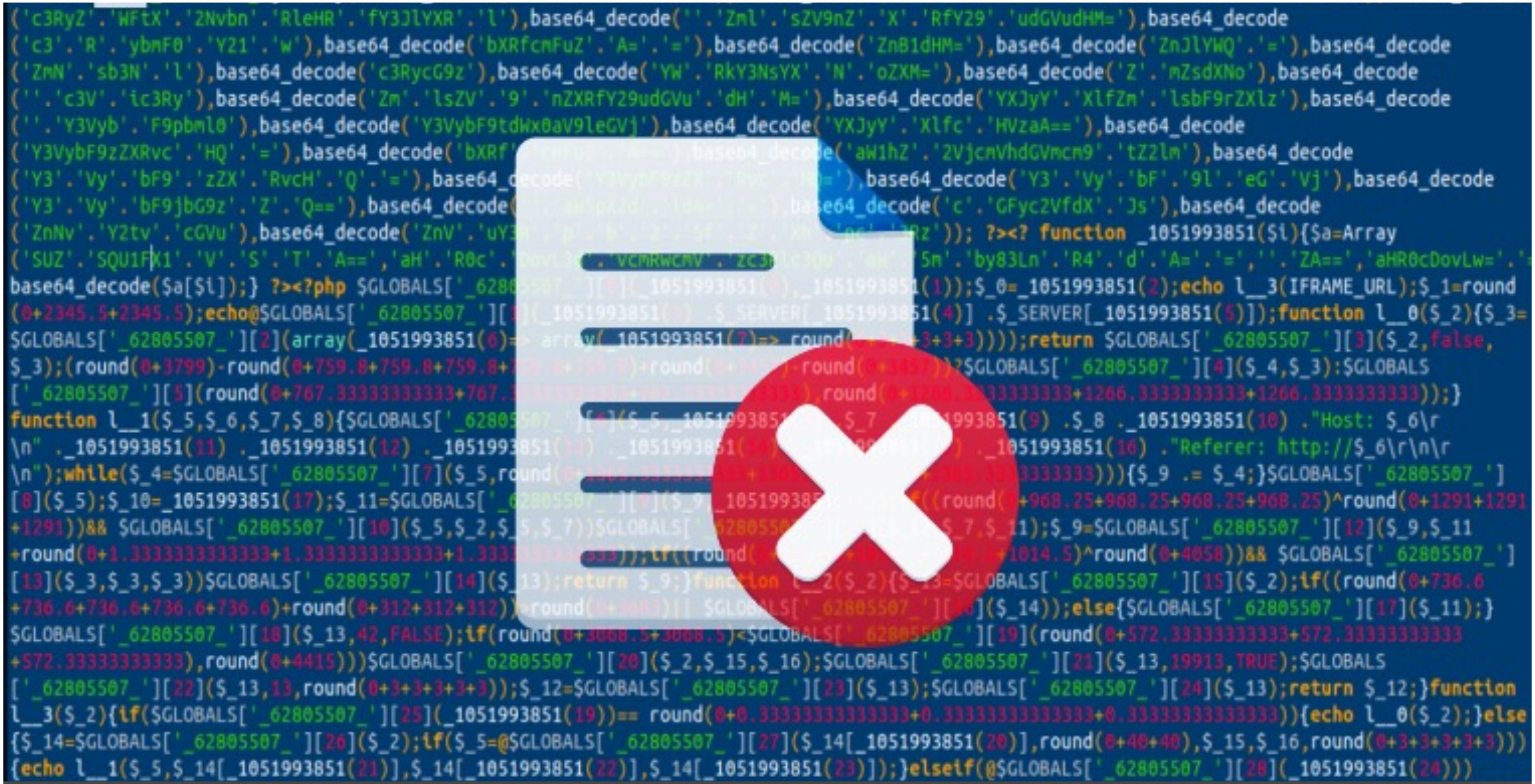
Software Reverse Engineering

COMP 272 | Spring 2022 | University of the Pacific | Jeff Shafer

Special Topics



Fileless Malware

Fileless Malware

- Every malware we've examined in this class has started with a file in the filesystem
  - `.exe` (Windows PE)
  - `.pdf` (PDF)
  - `.docx`, … (Office)
- Easy to imagine examining similar malware for Mac, Linux, Android, etc in a similar course
- **What about malware that does not need files in the filesystem to be effective?**
  - If it's not on disk, how do we find and analyze it?

Fileless Malware History

- Code Red worm (July 2001)
  - Attacked hosts running vulnerable Microsoft IIS web server (buffer overflow)
  - Defaced website
  - Attempted to scan Internet and spread
  - **Existed only in memory of infected host**
- SQL Slammer worm (January 2003)
  - Attacked servers running vulnerable Microsoft SQL Server
  - Attempted to scan Internet (fire-and-forget UDP packets) and spread
  - **Existed only in memory of infected host**

https://zeltser.com/fileless-malware-beyond-buzzword/

Fileless Malware History

- Banker Trojan (March 2012)
  - Malware loaded via JavaScript served via web advertising agency (used by Russian news sites)
  - JavaScript exploited Java vulnerability CVE-2011-3544 (for Windows and MacOS)
  - **Existed only in memory of infected host in the javaw.exe process**
  - Malware used to bootstrap Lurk banking trojan

https://zeltser.com/fileless-malware-beyond-buzzword/

Definitions

- Arguing over definitions…
  - Must fileless malware strictly not write *anything* to disk at all?
    - Examples: Code Red, SQL Slammer, Java Banker Trojan
    - Restarting computer will *temporarily* remove malware from system
  - What about storing some data in the Registry?
    - Technically the Registry is written to disk…
    - Examples: Poweliks, Phase Bot, …
    - Can be used to achieve persistence

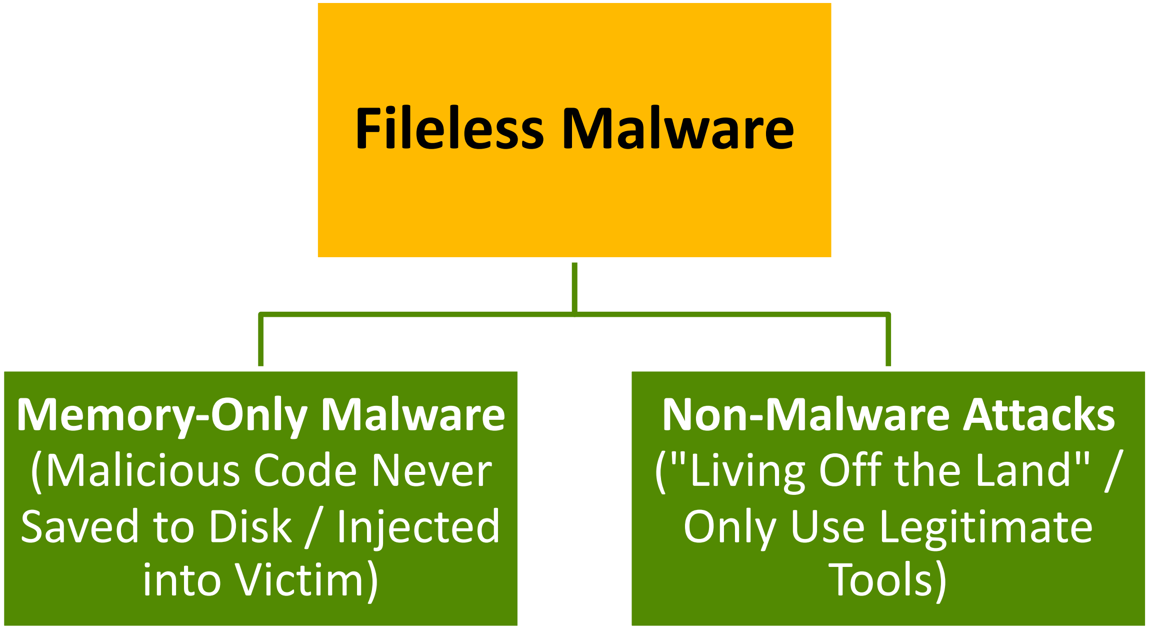
Fileless Malware History

- Poweliks (2014)
  - Spread via document malware (Microsoft Word), but document not needed after infection
  - Deployed with PowerShell, JavaScript, and shellcode
  - Persistence achieved via Registry (which stores malware)
  - Malware will persist after a reboot
  - Before/after snapshots of the filesystem will not reveal any new files

https://zeltser.com/fileless-malware-beyond-buzzword/

Fileless Malware History

- Many other fun examples described at https://zeltser.com/fileless-malware-beyond-buzzword/



Fileless Malware

Memory-Only Malware (Malicious Code Never Saved to Disk / Injected into Victim)

Non-Malware Attacks ("Living Off the Land" / Only Use Legitimate Tools)

Resources

- **Living off the land and fileless attack techniques**
  - Symantec Internet Security Threat Report
  - July 2017
  - Topics:
    - Living off the land, Defining fileless attack methods, Prevalence of dual-use tools, Dual-use tools in targeted attacks
  - https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-living-off-the-land-and-fileless-attack-techniques-en.pdf

Living Off the Land

- *Tactic* for malware authors – "**Living Off the Land**"
- Strategy
  - Use whatever tools are already installed on the targeted system
  - Drop few or no files on disk to avoid detection
  - Only use clean system tools that will have "known good" hashes

Fileless Attack – Persistence via Windows Registry

- Traditional use of Registry
  - Set the `/Run` key to point to your `.exe`
  - We did this in fake-malware lab
- Powerliks use of Registry
  - The `/Run` key points to `rundll32.exe` (legitimate program)
  - Normal usage
    - `rundll32.exe <dll-name>,<entry point> <opt args>`
  - Malicious usage
    - `rundll32.exe javascript:"\..\mshtml,RunHTMLApplication";<JS payload>;`
    - `rundll32.exe` will use `LoadLibrary` to search for matches for this "DLL" and eventually load `mshtml.dll` as a match
    - Entry point in `mshtml.dll` is `RunHTMLApplication`
    - JavaScript handler is used in `RunHTMLApplication`, which can execute code
    - Code will load payload from another registry entry and decrypt/run it

https://www.symantec.com/connect/blogs/poweliks-click-fraud-malware-goes-fileless-attempt-prevent-removal

https://blog.trendmicro.com/trendlabs-security-intelligence/poweliks-malware-hides-in-windows-registry/

Fileless Attack – Peristence via Services

- Windows Services are defined in registry too
  - Start a PowerShell script as a service?
- Command-line tool (`sc.exe`) can create a service for you

-
  ```
  sc create Payloadservice binpath= "C:\Windows\
  system32\cmd.exe /c start /b /min powershell.exe -
  nop -w hidden [MALWARE]" start= auto
  ```

Fileless Attack – Peristance via File Extensions

- Malware defines new file extension in registry
  - Instead of `.doc`, perhaps add `.notevil`
- Registry defines an action that is taken when running files with `.notevil` extension
  - Perhaps using `rundll32.exe` to execute a malicious script?
- Malware dumps some files with new extension in startup folder and/or a batch file listed in registry `/Run` key
- But there is nothing malicious *inside* these new files
  - Looks like random software cruft, AV says "clean"
  - All the malware logic is hidden in the Registry

Persistence Mechanism – Windows Management Instrumentation (WMI)

- Enterprise management tool: Windows Management Instrumentation (WMI)
- Query system settings, start/stop processes, execute scripts on local or remote machines
- Data stored in central WMI repository in encoded format
- Attacker can create periodic events in WMI that trigger their malicious PowerShell scripts

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

Dual-Use Tools

- Tool that could be used for *good* (by Sysadmin) or *evil* (by hackers)
  - `net user /add [username] [password]`
  - `query user >> %s`
  - `net view /domain >> %s`
  - `tasklist /svc >> %s`

- Legitimate tools may escape application whitelisting or some security tools
  - Would need to examine command-line arguments to determine if intent is good or evil

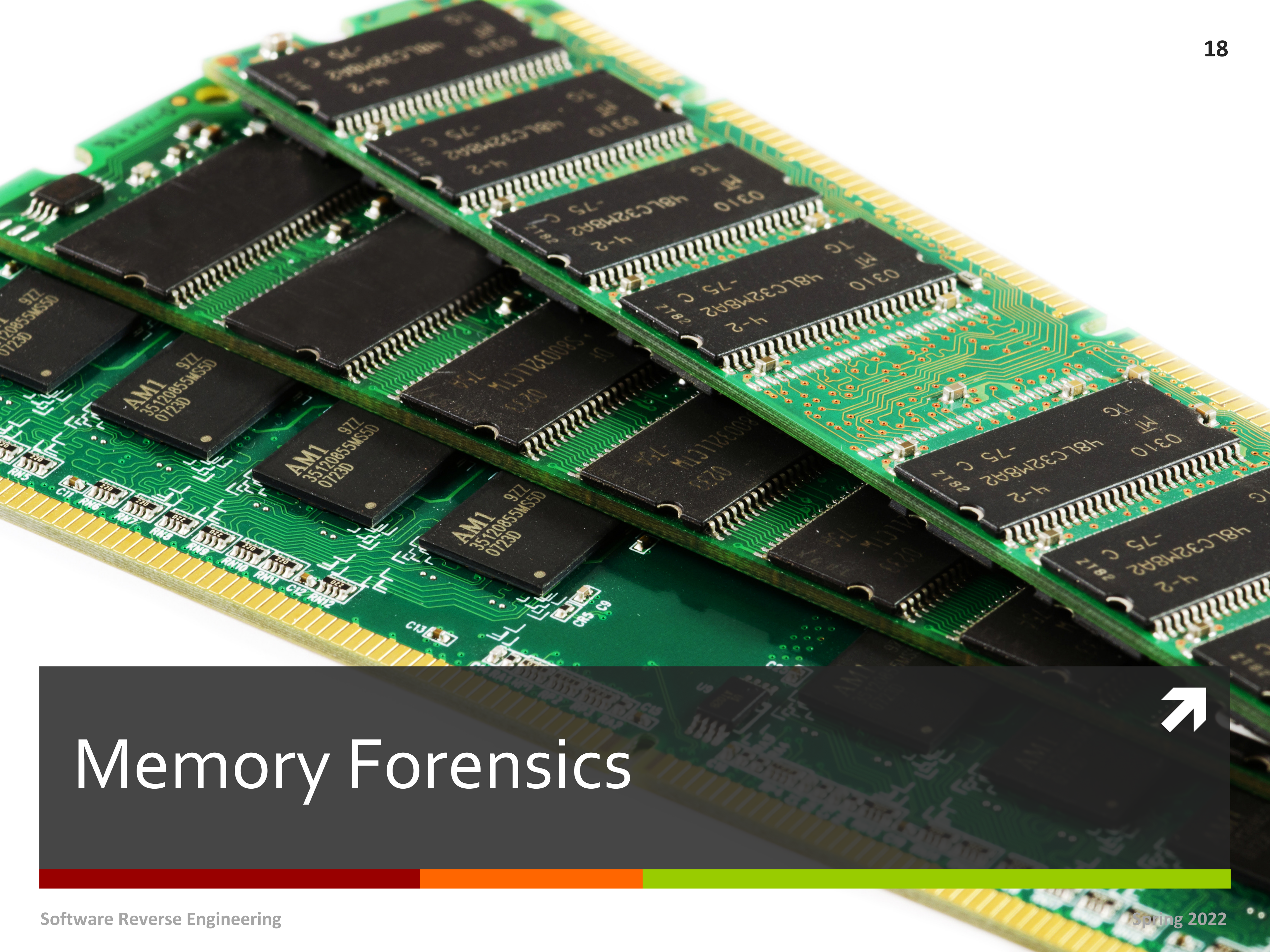
Dual-Use Tools

Activity	Purpose	Dual-Use Tools
Internal network reconnaissance	Enumerate information about a target environment	net (net user, net start, net view), systeminfo, whoami, hostname, quser, ipconfig
Credential harvesting	Obtain legitimate user credentials to gain access to target systems for malicious purposes	Mimkatz, Windows Credentials Editor (WCE), pwdump
Lateral movement	Gain deeper access into target network	RDP, PsExec, PowerShell
Data exfiltration	Send data back to attackers	FTP, RAR, ZIP, iExplorer, PuTTY, PowerShell, rdpclip
Fallback backdoor	Enables a backdoor that can be used, should the main backdoor be removed	Net User, RDP, Telnet server

https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-living-off-the-land-and-fileless-attack-techniques-en.pdf

Dual-Use Tools

- Note that sharing MD5 hashes of these tools is useless as an IOC
  - It’s not the *tool* that’s malicious
  - It’s how the tool is *being used* that is malicious
- `notepad.exe` *could* be malicious
  - Could be used to overwrite or modify contents of any file user has access to
  - Uploading the MD5 of notepad.exe to VirusTotal won’t help you



Memory Forensics

Memory Forensics

- Examine malware that has been denotated
  - Similar to behavioral analysis
- Rather than examining malware on running system, you examine a memory snapshot (complete contents of physical memory)
- Available artifacts
  - Similar to behavioral analysis – may find interesting ephemeral evidence
  - Active processes and their data (Encryption keys? Logins?), network connections, Registry, …

Memory Forensics

- **How to obtain a snapshot of physical memory? (and potentially pages in *swap* memory too)**
- Apps running within target system
  - WinPMEM - https://github.com/google/rekall/tree/master/tools/windows/winpmem
  - Comae Memory Toolkit - https://www.comae.com/
  - BelkaSoft Live RAM - https://belkasoft.com/ram-capturer
- Drawbacks
  - Malware may detect capture applications
  - Capture applications may evict malware data from memory as they work

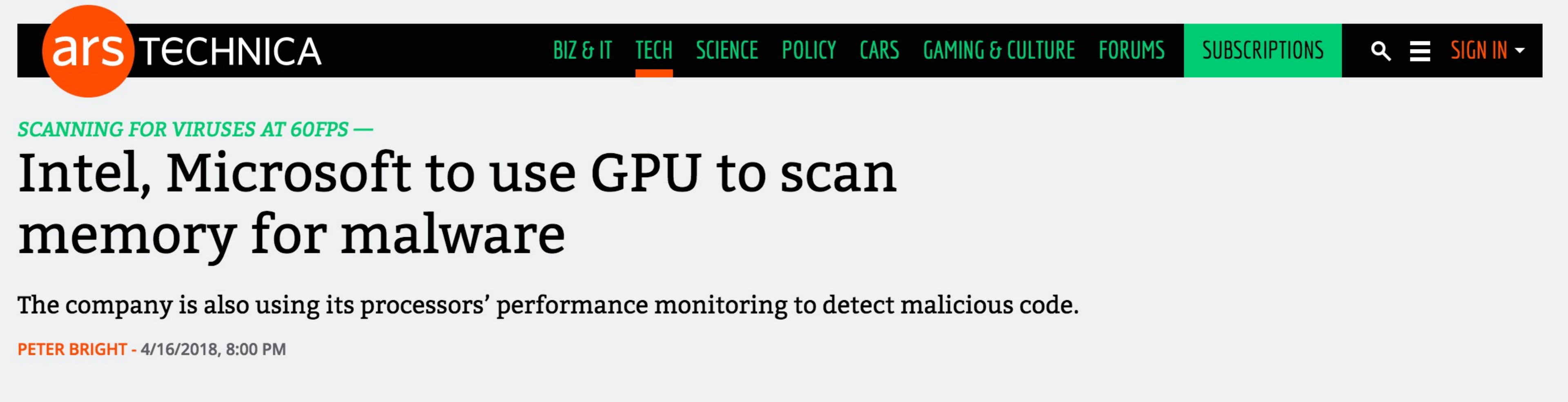
Memory Forensics

- **How to obtain a snapshot of physical memory? (and potentially pages in *swap* memory too)**
- Windows hibernation file
- Virtual machine snapshot file
  - Avoids running analysis tool *inside* target machine
- External hardware with Direct Memory Access (DMA)
  - Advantage: Try to detect *this*, malware authors!
  - Disadvantage: $$, operator skill

Memory Forensics Tools

- Volatility Framework
  - http://www.volatilityfoundation.org/
  - https://github.com/volatilityfoundation/volatility
- Rekall Forensics
  - http://www.rekall-forensic.com/
  - https://github.com/google/rekall
- FireEye Redline
  - https://www.fireeye.com/services/freeware/redline.html

Volatility Demo



SCANNING FOR VIRUSES AT 60FPS —

Intel, Microsoft to use GPU to scan memory for malware

The company is also using its processors' performance monitoring to detect malicious code.

PETER BRIGHT - 4/16/2018, 8:00 PM

- CPU scanning of RAM for malware artifacts is slow
  - "20% increase in processor load" – Intel
- The GPU has direct memory access (DMA) to main system memory without involving CPU
- The GPU has compute capabilities and memory of its own to save data
- Why not have the GPU scan main memory for malware periodically?
  - "Cuts processor load to 2%" – Intel

https://arstechnica.com/gadgets/2018/04/intel-microsoft-to-use-gpu-to-scan-memory-for-malware/