WMI – The Hacker's Chocolate to Their Powershell Peanut Butter

Understanding And Mitigating Attacks Leveraging Windows Management Instrumentation (WMI)

Rico Valdez | Splunk

rico@splunk.com

October 2018 | Version 1.0

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

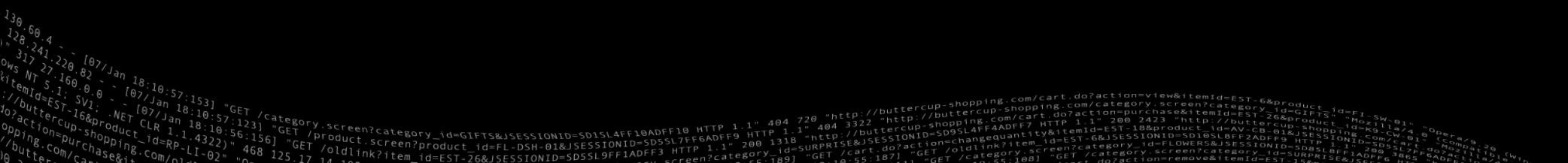
The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

splunk> .conf18

whoami

Obligatory identification

- Remote out of Albuquerque
- About 20 years in industry
  - 1st Half offensive
    - Red Teaming on DARPA and other .gov projects
    - Pen testing
  - 2nd half defensive
    - Operations work
    - Detections on endpoint data
- Enjoy really slick hacks
  - Stealthy, innovative, effective



Attacks Leveraging WMI

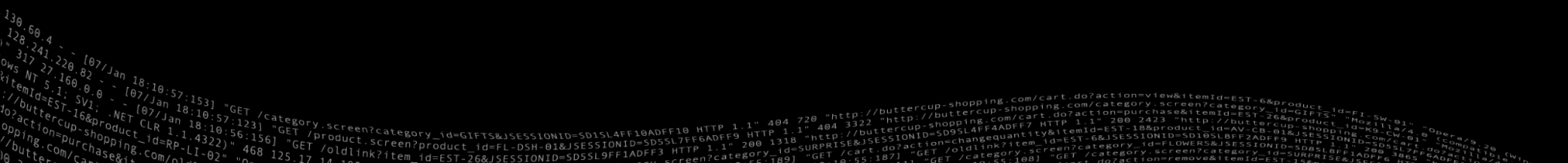
How WMI Is Being Used By Attackers

splunk> .conf18

Use in Attacks Increasing

Starts 1337 – then to the kiddies

- First Discussed @Kiwicon 2008
  - Rough for ‘The Moth’ Trojan
  - Introduced idea of using WMI Event Filters
- Stuxnet – 2010
  - WMI used to spread to network shares
    - https://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf
- TROJ_WMIGHOST - 2010
  - Watches for changes to the document’s ‘Recent’ folder
  - Run a script to prepare and upload files



Use in Attacks Increasing

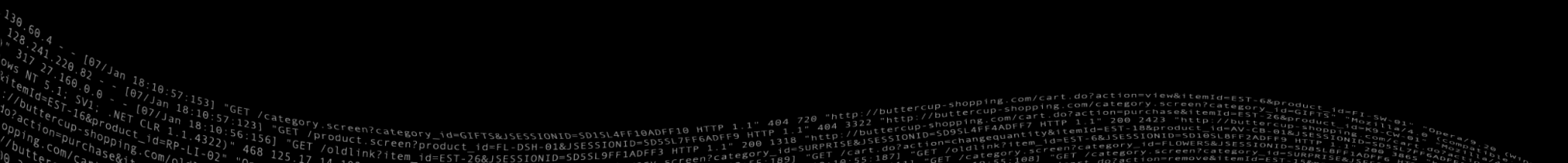
More Recent Activity

- Blackhat/Defcon 2015
  - Matt Graeber & Co Release POC backdoor and discuss the issues
- April 2017 – POSHSPY
  - WMI backdoor for backup
- August 2017 – Cryptominers
  - WMI Event Filters for persistence
  - ‘fileless’ malware
- May 2018 – Mainstream
  - SANS PR Release – “SANS to Tackle WMI-based Attacks at Boston Cyber Security Training”

Kiddie Approved!

Launch Your Own WMI Attacks!

- POCs available!
  - Mattefestation
    - Matt Graber's POC from 2015
  - WMIOps / WMImplant
    - https://github.com/FortyNorthSecurity/WMIOps
    - https://github.com/FortyNorthSecurity/WMImplant
- Metasploit modules!
  - WbemExec
  - WMI Event Subscription Persistence



What is WMI Anyway?

Framework Overview

splunk> .conf18

Windows Management Instrumentation (WMI)

Obligatory Stuff You Probably Don't Care About

Windows Management Instrumentation

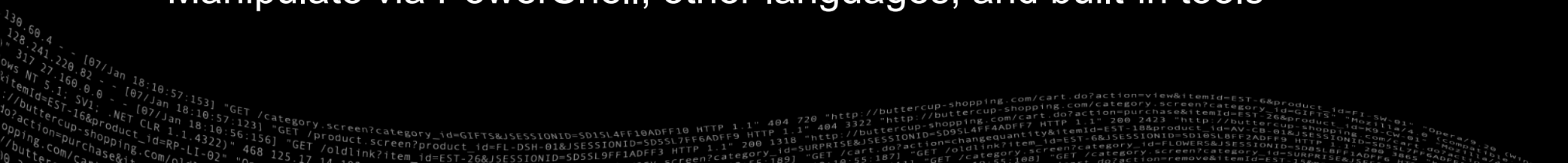
"Windows Management Instrumentation (WMI) is Microsoft's implementation of Web-Based Enterprise Management (WBEM), which is an industry initiative to develop a standard technology for accessing management information in an enterprise environment. WMI uses the Common Information Model (CIM) industry standard to represent systems, applications, networks, devices, and other managed components. CIM is developed and maintained by the Distributed Management Task Force (DMTF). "

- Microsoft

Windows Management Instrumentation (WMI)

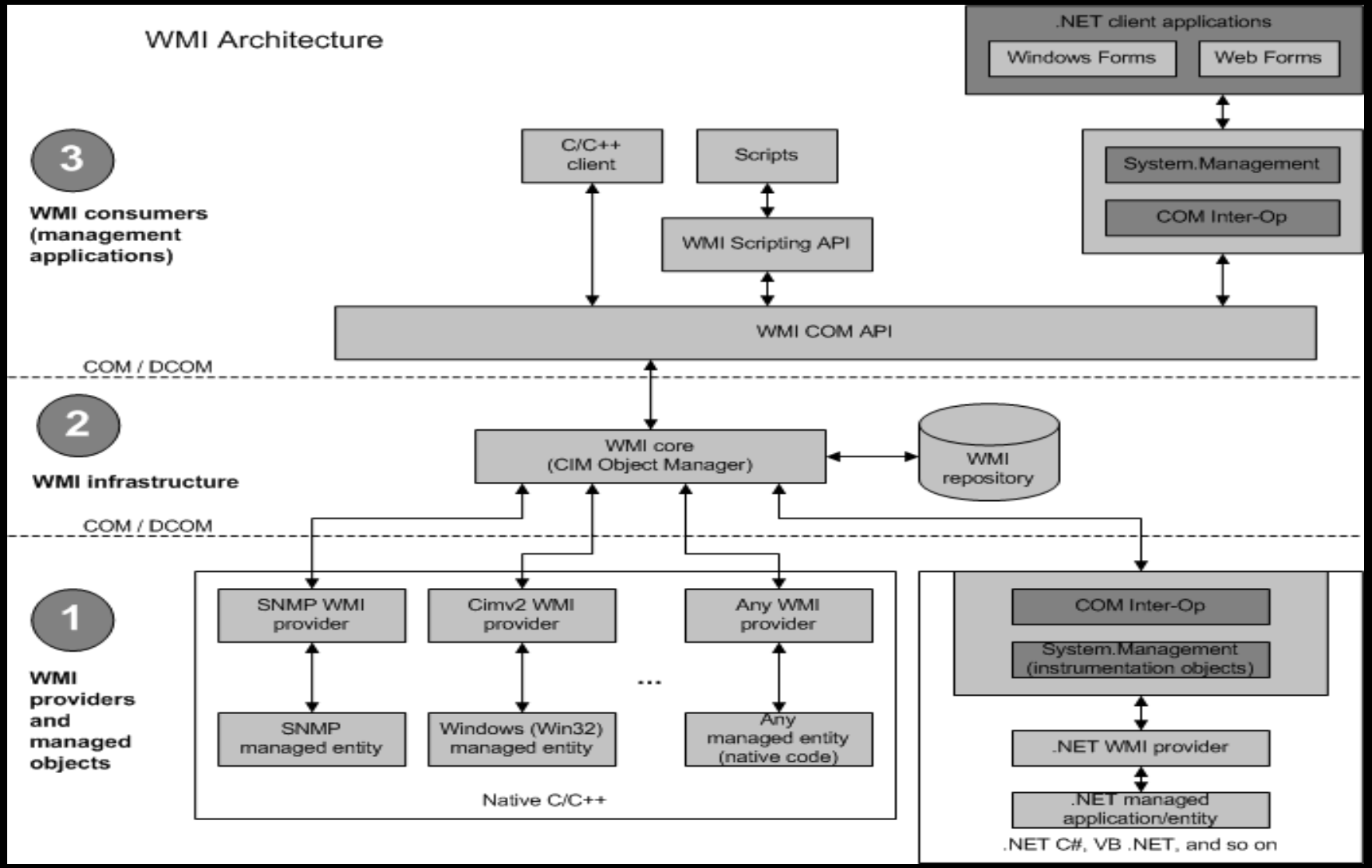
The Skinny

- WMI is architected in a provider/consumer model
  - Providers provide for management of system resources
  - Consumers interface with providers
- Extensible
  - Providers can be easily added for further functionality
- Powerful
  - Providers exist to control just about anything on the system
- Convenient
  - Part of all Windows Operating Systems
  - Powerful Remoting Capabilities
  - Easy to get data with WMI Query Language (WQL)
  - Manipulate via PowerShell, other languages, and built-in tools



Windows Management Instrumentation (WMI)

A picture says what?



Windows Management Instrumentation (WMI)

Key Concepts

- Managed Object/Component
- WMI Provider
  - COM-based Dll file (code) and mof file (description)
  - Registry Provider, Win32 - %windir%\system32\Wbem
- WMI Object/Class
  - Construct that describes the resources that can be managed
  - Object is just an instantiation of the class – used interchangeably
- WMI Infrastructure
  - Core and repository
  - Repository organized by WMI *namespaces*
    - *root\cimv2*
- WMI Consumers
  - Applications, scripts

Windows Management Instrumentation (WMI)

Some Example Providers

- Active Directory Provider
- BitLocker Drive Encryption Provider
- BizTalk Provider
- Boot Configuration Data (BCD) Provider
- DNS Provider
- Event Log Provider
- Hyper-V WMI Provider
- IIS Provider
- Ping Provider
- Security Provider
- Shadow Copy Provider
- System Registry Provider
- Trusted Platform Module (TPM) Provider
- Windows Driver Model (WDM) Provider

And many more…

splunk> .conf18

WMI Attack Details

Framework Overview

splunk> .conf18

Windows Management Instrumentation (WMI)

Why Attackers Likey

- Administrator access required for most
  - This is all ‘right of boom’
- Part of every windows version since Windows 2000
  - And available for earlier versions as well
- Allows for inspection/modification of system parameters
  - Pretty much everything
- Scriptable! Use with PowerShell is common in attack scenarios
  - Especially for Persistence
- Interface via WMI Command-line (wmic.exe) or winrm.exe
  - Already present on the system

Windows Management Instrumentation (WMI)

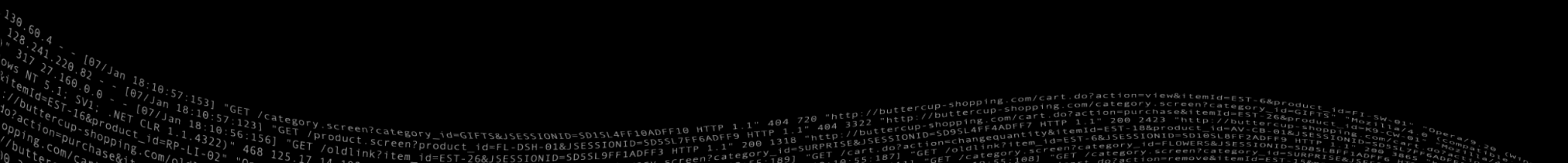
There's more…

- Able to be used in remote attacks
  - Works over DCOM and SOAP transports
- Also allows for establishing 'triggers'
  - Allows execution of tasks based upon some condition
- Poorly understood by many
  - Especially the potential for abuse
- Typically not actively monitored
  - Poor understanding of threat
- Great for 'fileless' attacks
  - Use existing functionality or embed script in repository

Capabilities - Discovery

Understand the environment

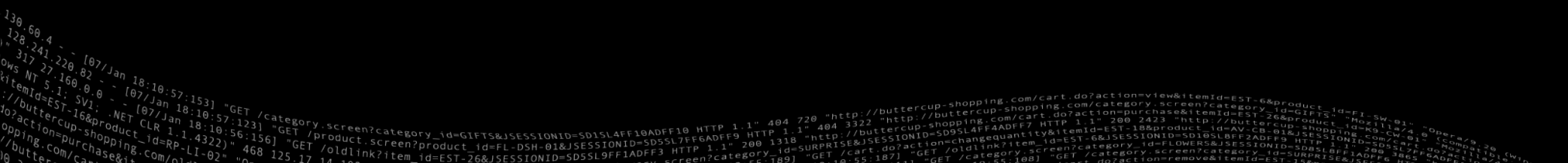
- Query system for various info
  - Files, processes, disk, registry, services, events, users, shares, patches, and more
  - Can help understand if running in Sandbox/VM
  - Profile the environment
- Use AntiVirus provider
  - Determine what AV is on the system
    - SELECT * FROM AntiVirusProduct



Capabilities – Lateral Movement

Spread throughout the target

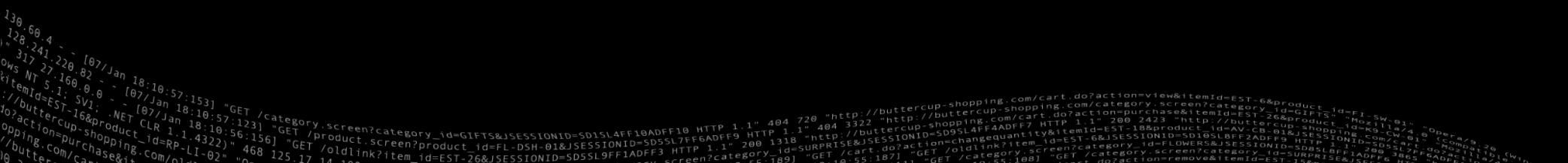
- Remote process execution
  - Wmic.exe
    - wmic.exe process call create
  - Winrm.exe
    - Requires the WinRM service to be listening
  - Powershell
    - Invoke-WmiMethod –Class Win32_Process –Name Create –ArugmentList ‘calc.exe’ –ComputerName 10.10.1.111
- Remotely Establish Persistence
  - Remote creation of event filters



Capabilities – Persistence

And go for the kill

- Event Filters
  - Temporary
    - Good for the session
  - Permanent
    - Stored in the object repository
- How to win –
  - Create an event filter
  - Create an action
  - Tie them together
  - ???
  - Profit!



Leveraging WMI

Getting to the capabilities

- Interact with WMI
  - Malware
    - C/C++ via COM API
    - .Net via System.Mangement classes
  - Command line tools
    - WMI Command Line – wmic.exe
    - Windows Remote Management – winrm.exe
  - Scripts (vbscript/javascript)
  - PowerShell
    - Typically best attack option

Step by Step

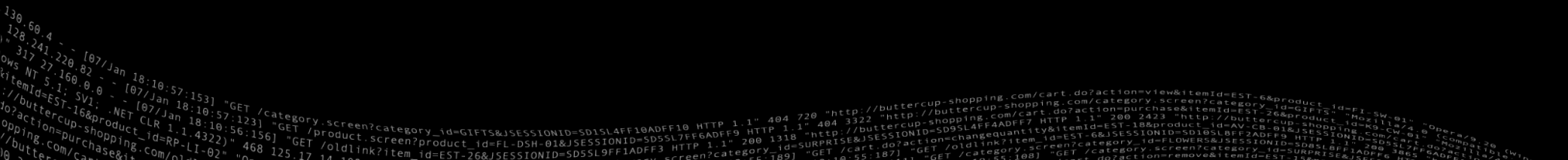
This is how they pwn us

- Query/Manipulate WMI Providers
  - Easily done with powershell cmdlets
    - Get-WmiObject –Namespace root\cimv2 –Class Win32_OperatingSystem
    - Get-WmiObject –Namespace root\default –Class StdRegProv –Push-Location HKLM:SOFTWARE\Microsoft\Windows\CurrentVersion\Run
  - Or wmic.exe
    - Wmic.exe /NODE:10.10.1.111 /USER:”TEST\Administrator” /PASSWORD:”password” process call create “cmd.exe /c calc.exe”

Step by Step

This is how they pwn us

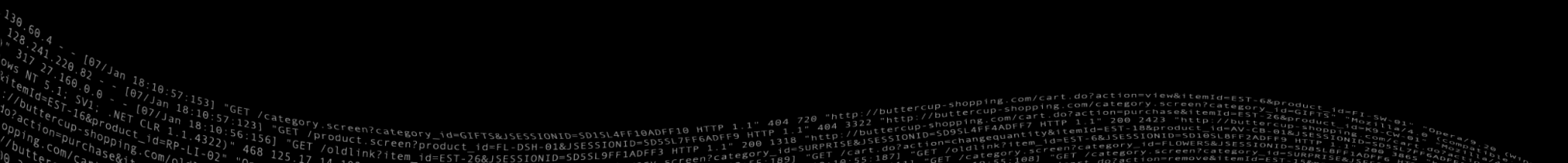
- Use Eventfilters to trigger actions
  - OS Commands
  - Execute code
- Create your own WMI Classes
  - Mofcomp.exe
  - Provide new interfaces/methods

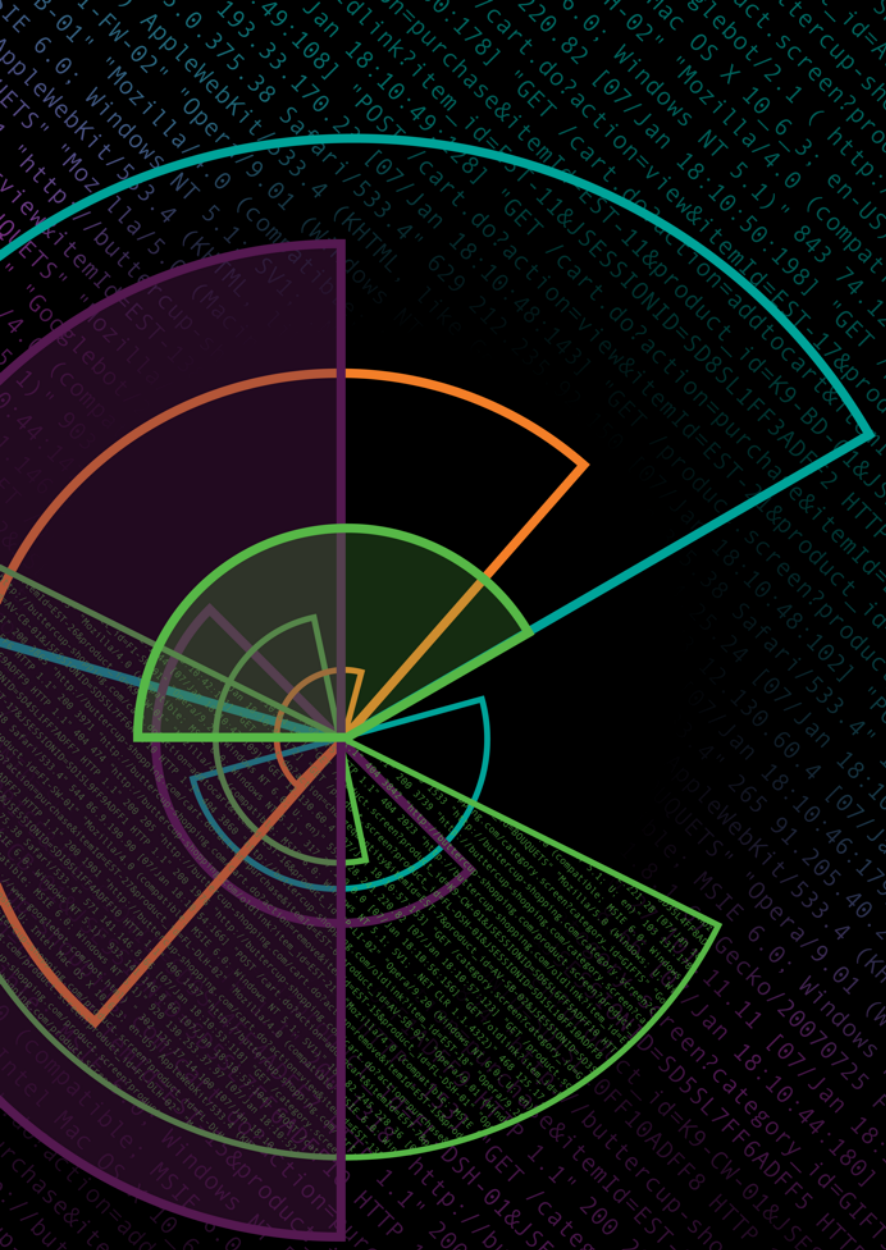


Recap – Why it's Dangerous

There be dragons here

- Prevalent
  - Already there – no need for malware execution
    - Will not trigger A/V or many endpoint products
      - WMI is part of the OS
  - Can use with powershell
- Capabilities
  - Sky's the limit
  - Remote capbilities
- Stealthy
  - Most not looking
  - Frustrates forensics





Detecting WMI Abuse

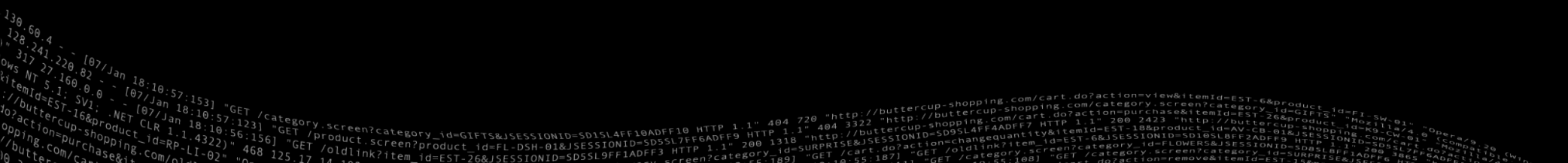
Framework Overview

splunk> .conf18

WMI Logging Data

Here's the 0xb33f

- WMI Logging
  - Windows Events
    - Better Logging in Current Versions
  - Sysmon –
    - WMI logging and events
  - Extended logging available
    - Event Tracing for Windows (ETW)
    - Typically not necessary unless troubleshooting



Windows Events

Default Windows Logging

- Some WMI Activity logged by default
  - Shows up in Event Log
  - Navigate to – Applications and Services\Microsoft\Windows\WMI Activity

- Enable collection by modifying inputs.conf
  - Add a new stanza –

```
[WinEventLog://Microsoft-Windows-WMI-Activity/Operational]
disabled = 0
renderXml = 1  (optional)
```

Windows Events

Default Windows Logging

- Current Versions of Windows - Default events logged
  - Event ID 5857 – Operation_StartedOperational
    - Provider loading
  - Event ID 5858 – Operation_ClientFailure
    - Typically query errors
  - Event ID 5859 – Operation_EssStarted
    - Permanent Event Filter Started
  - Event ID 5860 – Operation_TemporaryEssStarted
    - Temporary Event Consumer Registered/Started
  - Event ID 5861 – Operation_ESStoConsumerBinding
    - Permanent Event Consumer Binding

Sysmon

Get WMI Events Flowing

- Excellent tool for endpoint monitoring
  - Can install as a driver and send events to Splunk
- Captures most events you could think of
  - Network connections, file modifications, command lines, module loads, cross process activity
- Configuration file allows for fine-tuning events
- Various configs available
  - SwiftOnSecurity
    - https://github.com/SwiftOnSecurity/sysmon-config/blob/master/sysmonconfig-export.xml
- WMI Events off by default
  - <WmiEvent onmatch="exclude"></WmiEvent>
    - No entries – exclude nothing (get everything)

Sysmon

Here's your endpoint data

- WMI Event Monitoring Introduced in Version 6.1
  - Event ID 19 – WmiEventFilter activity detected
    - Event filter registration
    - Includes details such as WMI namespace, filter name, and filter expression
  - Event ID 20 – WmiEventConsumer activity detected
    - Registration of WMI Event Consumers
    - Inlcudes name, log, and destination
  - Event ID 21 – WmiEventConsumerToFilter activity detected
    - Consumer to filter bindings
    - Includes consumer name and filter path

WMI Log Files

The old way

- Pre-Vista – WMI Service maintained it's own log files in %system32%\wbem\logs

  - Wbemcore.log
  - Wbemess.log
  - Mofcomp.log
  - Wmiadap.log
  - Wbemprox.log
  - Framework.log
  - Winmgmt.log

"These files do not contain a consistent format that is suitable for reading programmatically"

WMI Log Files

The new hotness

- Extended Logging Options, including boot logging
- Extended WMI logs via Event Tracing for Windows (ETW)
  - WMITracing.log – binary file format – similar to other event logs
- Disabled by default
  - Via Event Viewer or command line
    - **Wevtutil.exe sl Microsoft-Windows-WMI-Activity/Trace /e:true**
- Uses Existing Windows Logging Facility
  - Compatibility with existing tools

Searches

Here's What To Look For

- Use of wmic.exe
  - Process logs from Windows Events or Sysmon
  - Look for argument 'process call create'
- Use of winrm.exe
  - Lateral Movement
- Execution of mofcomp.exe
  - New Provider compilation
- Parent Process
  - Children of WmiPrvse.exe may be indicative of WMI process instantiation (remote or local)
- File writes
  - Creation of MOF files in specific directories

Searches

Here's What To Look For

- Windows WMI Events
  - Event IDs 5859 and 5861
    - Best bet for finding malicious activity
    - May need to filter out known good
  - Event ID 5860
    - As permanent event filters get more scrutiny
- Sysmon WMI Events
  - Appear to be much less noisy
  - Event ID 19
    - Look for new Event Filters being registered
  - Event ID 21
    - Look for Binding (similar to Windows event 5861)

Other Thoughts

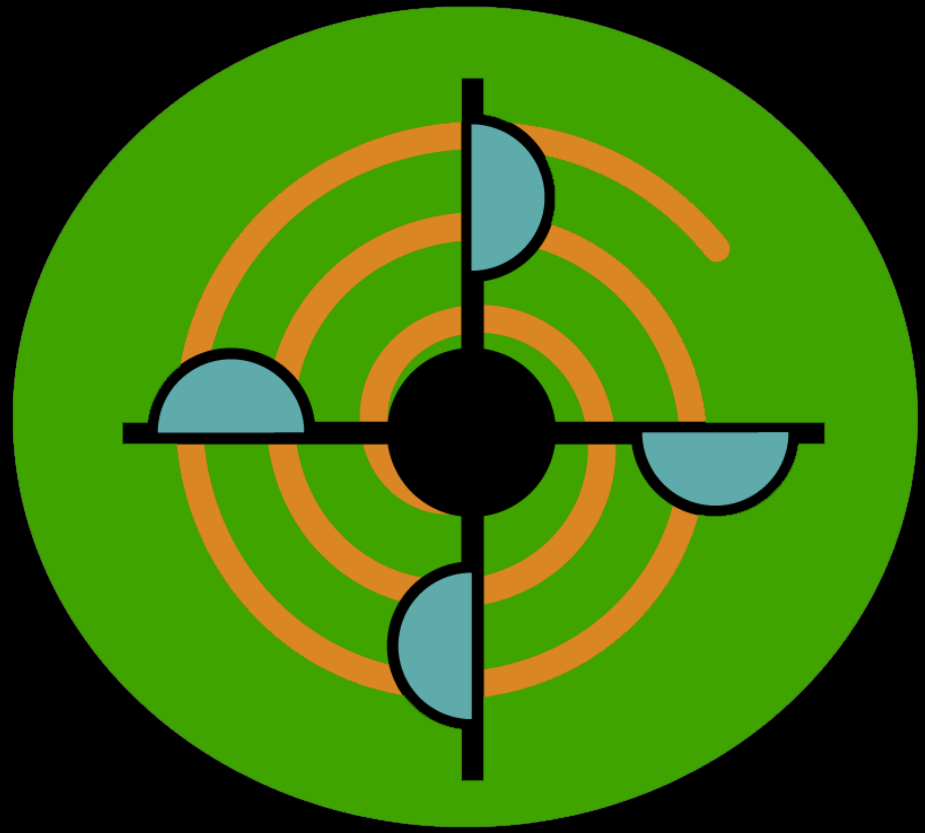
Why not?

- More support for traditional tools
  - Autoruns
- Disable WMI
  - Likely impractical in most environments
- Restrict WMI to a port and restrict
  - Allow specific connections via Windows Firewall
- Use WMI to generate events
  - Using same event filters
  - Expansion of this idea – WMI IDS
    - https://github.com/fireeye/flare-wmi/tree/master/WMI-IDS

Content in ESCU

WMI Story In Next Release

- Enterprise Security Content Updates
  - Analytic Story on suspicious WMI
  - Contains all searches discussed
  - Release in next few weeks



References

- Microsoft Docs on WMI -
  - https://docs.Microsoft.com/en-us/windows/desktop/WmiSdk/wmi-start-page
- FireEye Paper –
  - www.fireeye.com/content/dam/fireeye-www/global/en/current-threats/pdfs/wp-windows-management-instrumentation.pdf
- Trend Micro – Understanding WMI Malware
  - http://la.trendmicro.com/media/misc/understanding-wmi-malware-research-paper-en.pdf
- Good write up on tracking WMI -
  - https://www.darkoperator.com/blog/2017/10/14/basics-of-tracking-wmi-activity
- WMI Query Language
  - https://docs.microsoft.com/en-us/windows/desktop/wmisdk/querying-with-wql

Q&A

Rico Valdez | Principal Security Research Engineer

Thank You

Don't forget to rate this session in the .conf18 mobile app