

Flip the Script

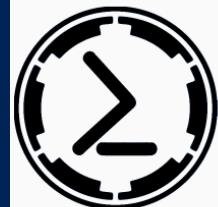
Microsoft's Incident Response Language

Jared Atkinson, Specter Ops

Agenda

- Defensive PowerShell
- Security Models
- Threat Hunting
- Gather
 - CIMSweep
 - PSGumShoe
 - Uproot
- Analyze
- Investigate
 - PowerForensics

Black Eye for the Blue Shell



PowerShellEmpire

<http://www.PowerShellEmpire.com>

*The best tool these days
for understanding windows
networks is Powerview
[1].*

-Phineas Fisher

<http://pastebin.com/raw/0SNSvyjJ>

PowerShellMafia / PowerSploit

Code

Issues 9

Pull requests 5

Pulse

Graphs

PowerSploit - A PowerShell Post-Exploitation Framework



OJ

@TheColonial



Following

Stageless #Powershell extension init scripts are go! Here's one dumping creds on start.
#meterpreter

PS C:\> [showmecon] PowerShell and You: Using Microsoft's Post-Exploitation Language [obscuresec]

PowerShell and You: Using Microsoft's Post-Exploitation Language

Christopher Campbell
@obscuresec



PSCONF.EU
POWERSHELL CONFERENCE EU

To Disable or Not To Disable...

Advising my customers to disable powershell using imagefileexecutionoptions until MS block this functionality or av vendors catchup.

I see more and more malware uses Powershell for its execution. Just disable it. You don't need it anyway.



How to Enable and Disable PowerShell V2 on Windows 8

Windows 8 by default includes PowerShell V3 installed, which is now an essential part of the operating system and cannot be removed. PowerShell V2 on the other hand is a different story....

sapien.com

12:30 PM - 6 Apr 2016

Question for pentesters/IT admins: Do you all disable Powershell, or see it being disabled on networks?



SecuriTay @SwiftOnSecurity · May 28

If you're in IT in an admin role, and you feel like you're not cool, learn scripting. It will change your life and your company forever.



97

253

...



Ctrl_Alt_Pasta

@Ctrl_Alt_Pasta



[Follow](#)

@SwiftOnSecurity powershell is disabled domain wide 😭



Matt Graeber

@mattifestation



[Following](#)

Hey does anyone know the PowerShell command to install Bash and permanently remove PowerShell?

RETWEETS
21

LIKES
23



12:50 PM - 30 Mar 2016



PSCONF.EU
POWERSHELL CONFERENCE EU

Where there's a will there's a way...



Ben Ten (0xA)
@Ben0xA



Following

I give you "Not PowerShell" (nps). Nice when you can drop a binary, also has encode/decode
github.com/Ben0xA/nps

```
C:\Downloads>nps.exe
usage:
nps.exe "{powershell single command}"
nps.exe "& {commands; semi-colon; separated}"
nps.exe -encodedcommand {base64_encoded_command}
nps.exe -encode "commands to encode to base64"
nps.exe -decode {base64_encoded_command}

c:\Downloads>nps.exe "Get-Date"
12/18/2015 12:59:40 PM

c:\Downloads>nps.exe "& Get-Date; Write-Output 'Winning'"
12/18/2015 12:59:53 PM
Winning

c:\Downloads>nps.exe -encode "& Get-Date; Write-Output 'Winning'"
JIBHZXQtRGF0ZTsgV3JpdGUT3V0cHV0ICdXawSuaW5nJw==

c:\Downloads>nps.exe -decode JIBHZXQtRGF0ZTsgV3JpdGUT3V0cHV0ICdXawSuaW5nJw==
& Get-Date; Write-Output 'Winning'

c:\Downloads>nps.exe -encodedcommand JIBHZXQtRGF0ZTsgV3JpdGUT3V0cHV0ICdXawSuaW5nJw==
12/18/2015 1:03:36 PM
Winning

c:\Downloads>_
```



Casey Smith
@subTee



Following

This is solid work by @jaredcatkinson
Remote PowerShell Exec as System via WMI.
Testing it more this morning.

leechristensen / UnmanagedPowerShell

Code

Issues 0

Pull requests 0

Wiki

Pulse

Graphs

Executes PowerShell from an unmanaged process

PS>Attack

A portable console aimed at making pentesting with PowerShell a little easier.



PSCONF.EU
POWERSHELL CONFERENCE EU

Cyber Kill Chain

- Discovery
- Delivery
- Exploitation
- C2 Installation
- Privilege Escalation
- Lateral Movement
- Data Collection
- Data Exfiltration



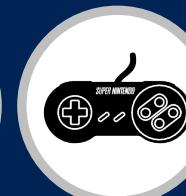
Discovery



Delivery



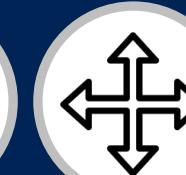
Exploitation



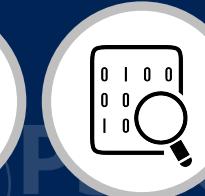
C2 Installation



Privilege Escalation



Lateral Movement



Data Collection



Data Exfiltration

Defense in Depth

- Stop the attack before it happens!
- Defense in Depth
 - Firewall
 - Anti-Virus
 - Web Proxy
 - Signature based Intrusion Detection Systems
- Users are weakest link
 - Attackers target humans to bypass technical controls
 - “...more than two-thirds of [Cyber Espionage] incidents ... have featured phishing.” -Verizon



Discovery



Delivery



Exploitation



C2 Installation



Privilege Escalation



Lateral Movement



Data Collection



Data Exfiltration

Incident Response

- Initiated by:
 - Network security monitoring alerts
 - Third party notification
 - Public disclosure
- Focuses on incident post-mortem
 - How did the attacker get in?
 - What data was taken?
 - What is the scale of attacker footprint?
 - Remediation
- Often too late to alleviate repercussions



Discovery



Delivery



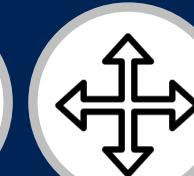
Exploitation



C2 Installation



Privilege Escalation



Lateral Movement



Data Collection

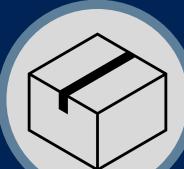


Data Exfiltration

Is something missing?



Discovery



Delivery



Exploitation



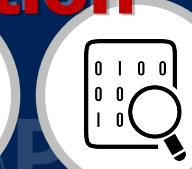
C2 Installation



Privilege Escalation



Lateral Movement



Data Collection



Data Exfiltration

Threat Hunting

- Rooted in the Assume Breach ideology
 - Determined adversaries can and will gain access to YOUR network
- Focuses on portion of kill chain between exploitation and exfiltration
- Proactively search for attacker actions (tactics) on network
- Defense with an Offensive Mindset
 - Covert
 - Step ahead of adversary
 - Plugging holes in the traditional defenses



Discovery



Delivery



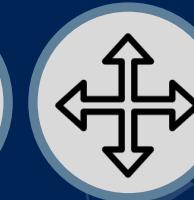
Exploitation



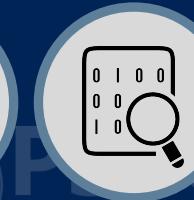
C2 Installation



Privilege Escalation



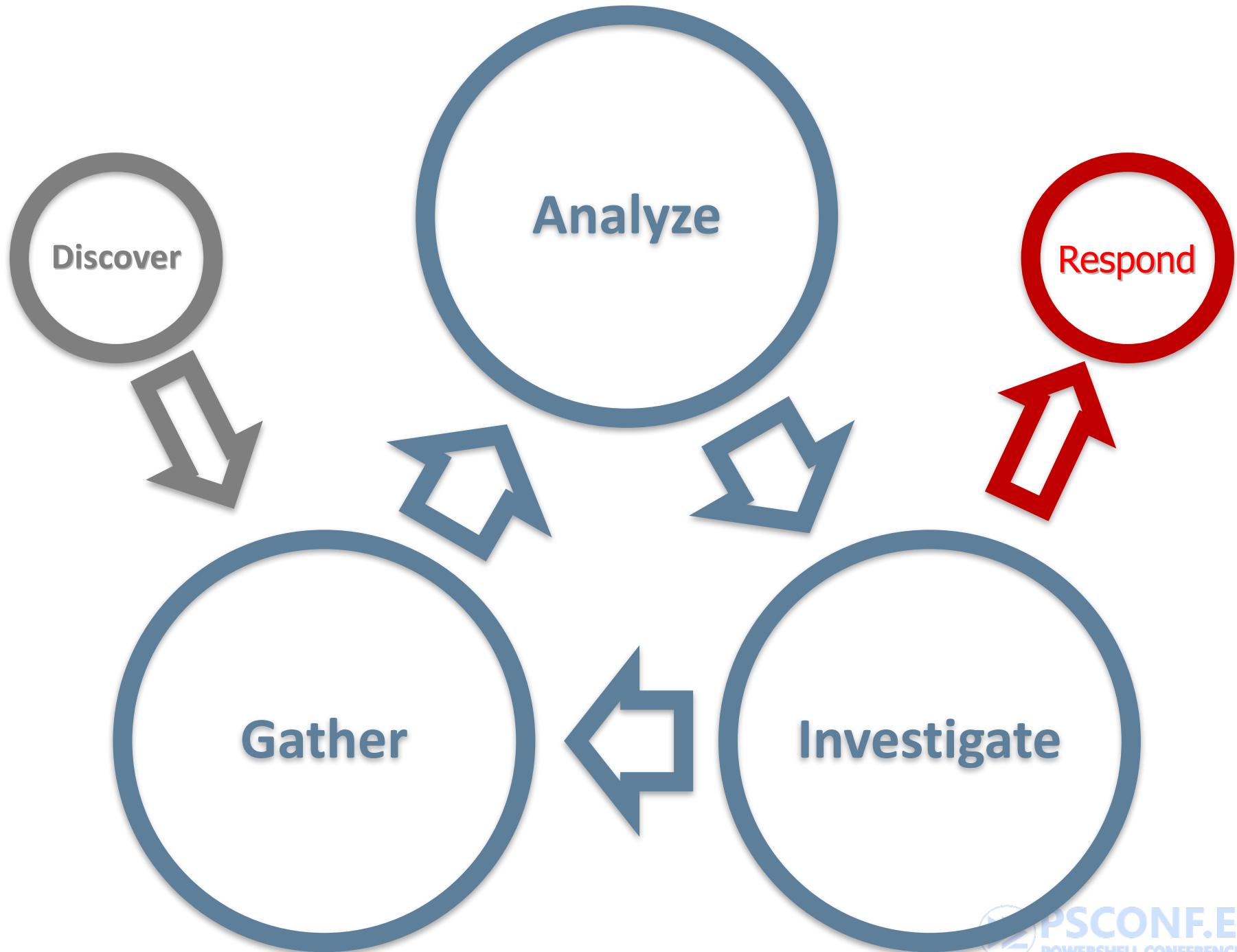
Lateral Movement



Data Collection



Data Exfiltration



Gather

- Gather host data from endpoints
 - Auto Start Entry Points
 - Evidence of Code Injection
 - Processes (Creation/Termination)
 - Services (Creation/Modification/Deletion)
 - Network Connections (Creation)
 - Registry (Writes/Reads Deletes)
 - Files System details
 - Logs

Gather Protocols (Supported by PS)

- WMI
 - Available since Windows 95
 - (Almost) always available on network
 - Data is limited to WMI classes
- WinRM
 - Downlevelable to Windows XP (kind of)
 - Unlocks remote PowerShell functionality (.Net and Win32 API)

To Pull or To Push

- Pull
 - Pros
 - Little to no setup required
 - Great for long lasting artifacts
 - Auto Start Entry Points
 - Injected Code
 - Cons
 - Relatively slow
 - Redundant data points
 - Blind to activity between “pulls”
 - Example Projects:
 - CIMSweep
 - PSGumshoe
 - Infocyte (commercial)
- Push
 - Pros
 - Near real-time
 - Distributed
 - No missed events
 - Cons
 - Requires “agent” on endpoints
 - Example Projects:
 - Sysmon + WEF
 - Uproot
 - Carbon Black (commercial)



Value of WinRM

- Single port (5985/5986)
- Encrypted by default
- Enables numerous valuable capabilities
 - PowerShell Remoting (Pull)
 - Windows Event Forwarding (Push)
- Provides alternative to vulnerabilities of current admin protocols
 - SMB
 - Many of the most prolific exploits have targeted this service
 - MS 08-067 (CONFICKER)
 - MS 17-010 (ETERNALBLUE)
 - RDP
 - Interactive logon (exposes administrator credentials to be stolen)

CimSweep - Pull

- CIM/WMI-based tools that enables the ability to perform hunting operations remotely across all versions of Windows
- Built on PowerShell's CIM cmdlets
 - Can use DCOM or WSMAN for transport
 - WMI is running by default on Windows 2000 and later OS's
 - CIM Sessions can be established once and reused to speed up Sweeps

PSGumshoe - Pull

- Windows PowerShell module for the collection of OS and domain artifacts for the purposes of performing live response, hunt, and forensics.
 - Analytic Scripts (Damerau-Levenshtein)
 - Artifact (Named Pipes)
 - Attacker Techniques (Get-InjectedThread)
- Transport agnostic
- Backward compatible
- Call to Action

<https://github.com/PSGumshoe/PSGumshoe>

Get-InjectedThread

- Built on PSReflect
- PowerShell function to identify injected threads via detection methodology:
 - Use Windows Toolhelp API to get all threads
 - Iterate through each thread
 - Identify the thread's Base (Memory) Address
 - Query the memory page for the Base Address
 - if (\$_.State -eq MEM_COMMIT -and \$_.Type -ne MEM_IMAGE)
- Returns details regarding offending process and thread

Get-InjectedThread Demo

<https://youtu.be/9TIxt0Z3iZI>



PSCONF.EU
POWERSHELL CONFERENCE EU

Uproot - Push

- WMI based Intrusion Detection System
- Leverages WMI Permanent Event Subscriptions to detect/report:
 - General System Information
 - Introduction of Persistence
 - Lateral Movement
- Reports Events via:
 - Windows Event Log (Windows Event Forwarding)
 - HTTP POST

Uproot Demo

<https://youtu.be/uBKeG1GEh8s>



PSCONF.EU
POWERSHELL CONFERENCE EU

Analyze - Your SIEM of Choice

1. Identify and (sometimes) exclude known goods
2. Identify and report known bad activity
3. Tactics, Techniques, and Procedures (TTP)
 - MITRE ATT&CK (attack.mitre.org)
4. Least Frequency of Occurrence (compared to peers)
 - The “WinUpdate” service is running on 1 of 10,000 endpoints
5. First Seen
 - Connections to never before seen domains
6. Identify unknown unknowns that need further investigation
 - Avoid polarizing words like “malicious”, “indicator”, or “finding”
 - Use words like “item of interest” or “anomaly”



Investigate

- Determine context surrounding events identified in the Analyze phase
 - Data Pivoting
 - Network Connections
 - Running Processes
 - Persistence Locations
 - Temporal context (Timelining)
 - File System activity
 - Historical context
 - Event Logs
 - Proxy Logs

PowerForensics

- C# PowerShell based Forensic Toolkit
- Allows inspection of NTFS and FAT data structures
- Create an event timeline based on forensic artifacts
 - File Operations
 - Registry Activity
 - Program Execution
 - User Activity
- Upcoming HFS+ and EXT4 File System Support

PowerForensics Demo

This demo is based on the @security4arabs Digital Forensics challenge by @binaryz0ne. Please visit <http://goo.gl/CVoEpo> to download a copy of the challenge.

<https://youtu.be/H7clQOhbHIQ>



PSCONF.EU
POWERSHELL CONFERENCE EU

Timeline Visualization Demo

This demo builds on work by @_RyanBenson . For more information on leveraging Gource for timelining, please see Ryan's blog post
<https://www.obsidianforensics.com/blog/visualizing-usn-journal-activity>

<https://youtu.be/v5mYegFG1DA>



PSCONF.EU
POWERSHELL CONFERENCE EU

Administrator: Windows PowerShell ISE Preview

File Edit View Tools Debug Add-ons MyCommands Expert Level Compatibility Help

Untitled2.ps1*

no functions in script (click icon on left hand side to create one)

search

```
1 # Create beginning bound of our investigation window
2 $start = Get-Date -Year 2015 -Month 09 -Day 01 -Hour 00 -Minute 00 -Second 00
3
4 # Create end bound of our investigation window
5 $end = $start.AddDays(4)
6
7 # Create a forensic timeline of the target volume
8 $timeline = Get-ForensicTimeline -VolumeName N:
9
10 # Filter for timeline events that reside within the xampp directory and its children AND occurred within our investigation window
11 # Sort events based on the Date field
12 # Format events to match Gource's input template
13 # Pipe events into Gource for visualization
14 $timeline | Where-Object {($_.Date -gt $start) -and ($_.Date -lt $end) -and ($_.Filename.Contains('\xampp'))} |
15 Sort-Object Date | ConvertTo-Gource | .\gource.exe --log-format custom --disable-bloom -f -
```

Completed. Last command took 86.814 seconds.

Editor Input

Ln 10 Col 1

10:06 PM
2/23/2016



PSCONF.EU
POWERSHELL CONFERENCE EU

Summary

- PowerShell for Threat Hunting
- Gather -eq Push or Pull
 - CIMSweep
 - PSGumShoe
 - Uproot
- Investigate
 - PowerForensics

Questions?

About_Author

- Jared Atkinson
 - Technical Director, Adversary Detection – Specter Ops LLC
 - Former
 - U.S. Air Force Hunt (2011 – 2015)
 - Veris Group's Adaptive Threat Division (2015-2017)
 - 2015 Black Hat Minesweeper Champion
 - Microsoft MVP (Cloud and Datacenter Management/PowerShell)
 - Open Source Developer
 - PowerForensics
 - Uproot IDS
 - PSGumShoe
 - Researcher of forensic artifact file formats



PSCONF.EU
POWERSHELL CONFERENCE EU