Endpoint Threat Hunting

Brian Olliff

Defensive Engineering Instructor

Key Concepts

- Endpoint hunting intro
- Threat hunting in Windows
- Hunting using SIEM applications

MAJOR TOPICS

- Endpoint hunting IOCs
- Windows hunting
  - Processes and services
  - Sysmon
- Endpoint hunting with Splunk
- Endpoint hunting with ELK

LEARNING OUTCOMES

- Recognize use of malicious Windows services and processes
- Understand the importance of Sysmon and how it’s used in endpoint hunting
- Be able to perform effective hunts in both Splunk and ELK systems

PREREQUISITES

- Understand basics of Windows systems
- Cybersecurity fundamentals
- Previous use of Splunk/ELK (helpful, but not required)

LET'S GO!

Introduction to Endpoint Hunting

Endpoint Threat Hunting

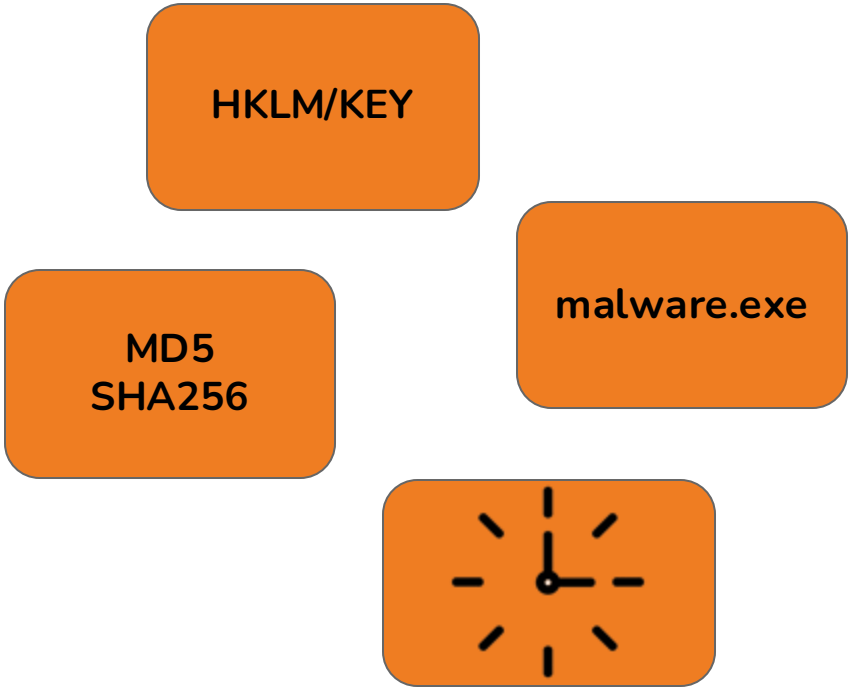
- Endpoint-based IOCs and hypotheses
- IOC or TTP hunts
  - IOCs can be endpoint or network types
- Can begin with
  - Specific intelligence information
  - Reports from IR/SOC teams
- Typically performed on collected logs
  - Authentication, web, applications, system events, etc
    - PowerShell logging is critical
  - Ingested from multiple locations
  - Aggregated into central location (SIEM)

Hypothesis or Trigger

“Attackers have compromised a workstation and are executing files with these hashes and communicating with a C2 server at <IP>.”

“Attackers have established a foothold on the network and created admin accounts to maintain their access.”

Endpoint IOCs



- Intelligence sources
  - Reports from ISACs/vendor
  - Paid/free feed
  - Network team/SOC
- System logs
  - Built-in events
  - Applications
  - Sysmon
- Commonly correlated with others

Endpoint IOCs

File-Based IOCs

- Filenames
  - Not most reliable - can easily be changed
  - Still include in searches, esp if known unique or known pattern
- File hashes
  - MD5 & SHA256 are most common to see
  - E6CD360FDEFE3EDF307C49612478D386
  - 85325FC3DF13AC10D514EB7E6B8F3BE85E62BF6220FFA46CB161A506C0488307
  - Fairly easy to modify
- File path
- File size

Windows System-Based IOCs

- Registry keys
  - *reg add*
  - *\Software\Microsoft\Windows\CurrentVersion\Run*
  - Sysmon event ID 12/13/14, Windows Event ID 4657
- Scheduled tasks
  - *schtasks /create*
  - Windows Event ID 4698 (create), 4701 (disable), 4702 (update)
- Windows services
  - *sc.exe create*
  - Event IDs 4698/7045 (new service)

Windows System-Based IOCs

- Windows processes
  - Any running executable **or** Windows system process
  - Unusually named or excess running processes
  - Renamed file running as “legitimate” system process
- Event IDs
  - 1102 - audit log cleared
  - 4624 & 4625 - successful & failed logons
- PowerShell
  - Event IDs 4103 & 4104 - module logging & scriptblock logging
  - *Invoke-Expression* (iex)
  - *Invoke-WebRequest* (iwr)
  - *EncodedCommand*

Behavioral and Other IOCs

- Unexplained/unusual activity
- Deviations from baselines
  - Security settings, local firewall configuration
  - Installed software
- Phishing
  - Email itself, also indicators inside email
- New user creation
  - Frequently used for persistence
  - Event ID 4720
- File downloads
  - PowerShell or other utilities (certutil.exe)

Windows Processes and Files

Windows System Processes

- Identification
  - Name
  - PID
  - Executable path

- Key questions to answer to determine if legitimate
  - Did expected parent process spawn this process?
  - Is it running from the expected location?
  - Is it spelled correctly?
  - Is it signed by MS?

Commonly Abused System Processes

Session Manager	C:\Windows\System32**smss.exe**
Client/Server Run Subsystem	C:\Windows\System32**csrss.exe**
Windows Login	C:\Windows\System32**winlogon.exe**
Windows Initialization	C:\Windows\System32**wininit.exe**
Service Control Manager	C:\Windows\System32**services.exe**

Commonly Abused System Processes

Local Security Authority Subsystem	C:\Windows\System32**lsass.exe**
Generic Service Host Process	C:\Windows\System32**svchost.exe**
Generic Host Process	C:\Windows\System32**taskhostw.exe**
Windows Explorer	C:\Windows**explorer.exe**

Recognizing Malicious Processes

- Any process that runs a child process of:
  - cmd.exe
  - wscript.exe
  - powershell.exe

- Sysmon Event IDs
  - 1 - Process creation
  - 10 - ProcessAccess

Windows Services

Windows Services

- Processes running in background

- Multiple ways of starting
  - On boot (automatic)
  - Manually by user or trigger
- Used by attackers for persistence
  - Malicious process to start at boot
  - Connect to C2 infrastructure
  - Ensure elevated access

Attacker Abuse

- Create new service
  - *sc.exe create*
  - PowerShell *New-Service*
  - WMI commands
  - Registry modifications
    - HKLM\SYSTEM\CurrentControlSet\Services\
  - Third-party tools
- Modify/replace existing service
  - Change executable
- Change recovery options on service
  - Run program when service fails

Recognizing Malicious Services

- Any application can create service
- Service creation indicators
  - ***sc.exe create*** in logs
  - ***New-Service*** in logs
  - ***reg add*** in **HKLM\SYSTEM\CurrentControlSet\Services**
    - Sysmon Event ID 12, 13, 14 - RegistryEvent
  - Event ID 4697 or 7045
- Unrecognized/unauthorized users creating/modifying services

Scheduled Tasks

Scheduled Tasks

- Automated tasks to run programs or scripts
- Multiple triggers
  - Date/time
  - Regular intervals (ex: every 2 weeks)
  - At log on, start up
  - Workstation lock/unlock
- Frequently used by attackers for persistence and stealth

Attacker Abuse

- Malicious task creation
  - Command line: *schtasks /create*
  - PowerShell: *New-ScheduledTask*
  - GUI: Task Scheduler
    - Can be harder to detect in hunts
- Modify existing tasks
  - *schtasks /change*
  - *Set-ScheduledTask*
- If using PowerShell, command may be encoded

Recognizing Malicious Tasks

- Task creation indicators
  - *schtasks /create*
  - *New-ScheduledTask*
  - Event ID 4698 - Scheduled Task Created
  - Event ID 4700 - Scheduled Task Enabled
  - Event ID 106 - Scheduled Task Created (in Task Scheduler channel)
- Task modification indicators
  - *schtasks /change*
  - *Set-ScheduledTask*
  - Event ID 4702 && 140 - Scheduled Task Updated

Sysmon

What is Sysmon?

- Part of Sysinternals suite
- Monitors system and logs activity to Windows event log
  - Additional logging not built in to Windows
    - File creation
    - Network connections
- Operates as Windows system service/device driver
  - Runs as protected process
- Should be installed on all endpoints as necessary
  - Some data not logged by default Windows Events
- Logged events (ideally) forwarded to SIEM

Capability Overview

- Logs process creation with full commands entered
  - Includes current and parent process
- Records hash of executed files
  - Multiple algorithms
- Logs driver and DLL loading with signatures and hashes
- Can log network connections
  - Source process, IP address, port number, hostname
- File creation/overwrite information
  - Including changes in file creation timestamp

Useful Sysmon Events

Event ID	Description
1	Extended information about created processes
2	Process changed file creation timestamp
3	Network connection logged
6	Driver loaded into system
8	Thread created in another process
9	Raw disk access using \\.\
10	Process opens another process

Useful Sysmon Events

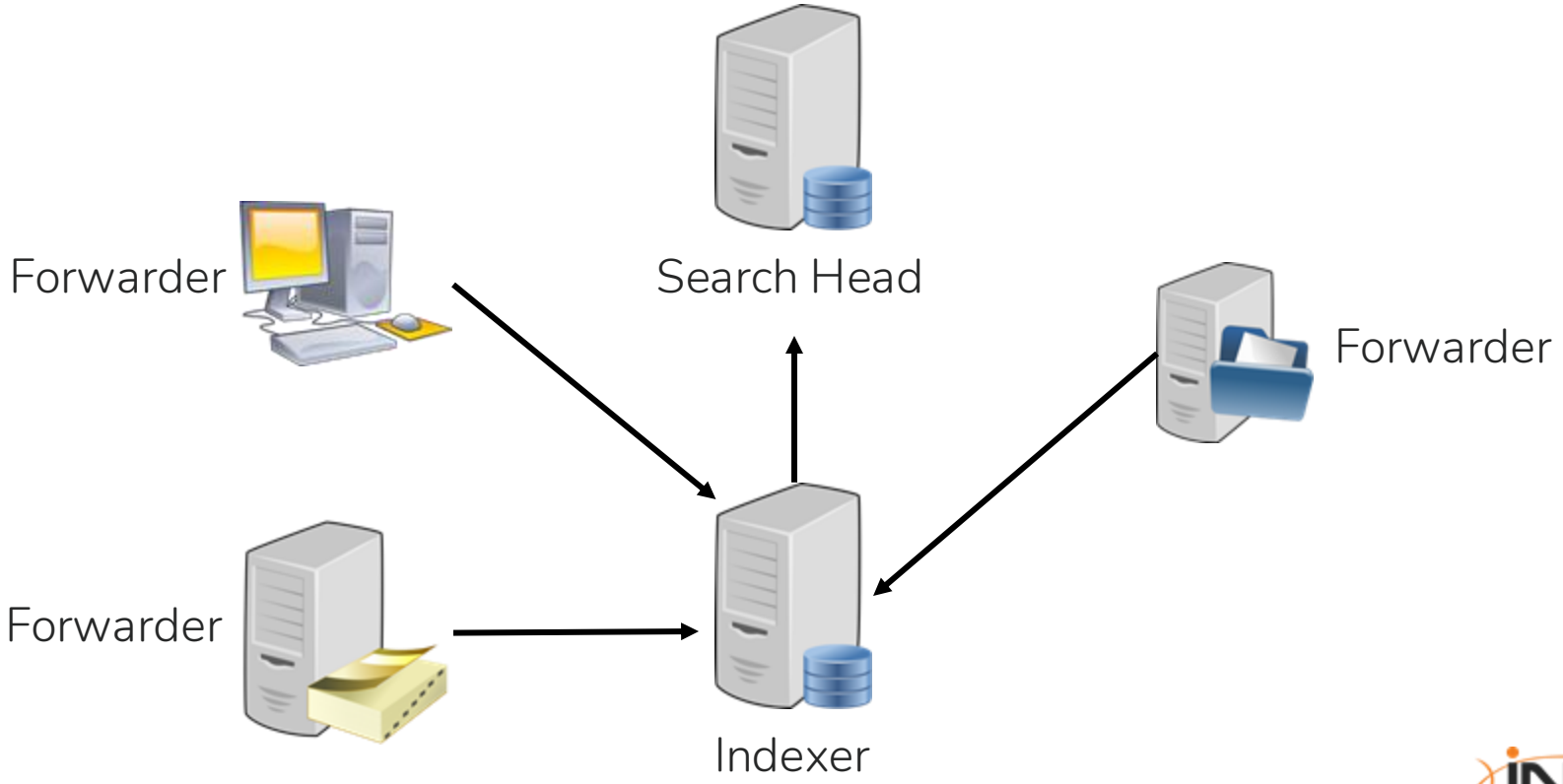
Event ID	Description
11	File created or overwritten
12, 13, 14	Registry key/value created or deleted; value modified; key/value renamed
19, 20, 21	WMI Events
22	DNS query
23	File deleted (and archived to C:\sysmon)
25	Process tampering detected

Splunk
Introduction

What is Splunk?

- Security Information & Event Management (SIEM)
- Log/data analysis
- Collects logs from multiple sources
  - Aggregation and correlation
- Capable of real-time analysis and alerting
- Dashboards & visualizations
- Raw log searches (SPL)
- Add-on applications to extend functionality

Splunk Architecture



Splunk Processing Language

SPL

- Index
- Sourcetype
- Filters
- Pipes
- Commands
  - table, stats, sort, top, dedup
- Field=value
- Boolean logic & grouping
- Raw event search vs transforming search

Common Splunk Queries

Building a Query

- Good queries start with good hypotheses
  - Know what you're looking for
- Example
  - Attackers are running suspicious scripts from temporary folders
  - Search possibilities:
    - .ps1 files executed within scope timeframe
    - Any non-executable from C:\Windows\Temp or C:\Temp
    - Script files created in temp folders

Common Query Purposes

- New user creation
- Brute force attempts
- Unexpected outbound network connections
- Suspicious PowerShell activity
  - Encoded commands
  - Downloads
- Looking for persistence techniques
  - Scheduled tasks
  - Autorun
  - Malicious services

Splunk Search Tips

- Ensure proper time range is appropriate for hunt
- Format results into tables for easier readability
  - Useful for summary of information
- Use *stats*, *count*, *sort*, *top* & *rare*
- Remember **|** to add functionality
  - *where* - filters results further
  - *eval* - rename fields, create new fields based on calculations

Threat Hunting with Splunk

Putting it All Together

- Threat hunting is proactive
  - Not relying on alerts or notifications
- Hypothesis-driven
  - Not random queries
- Using Splunk in hunts
  - Centralized visibility with historical data
  - Ability to easily pivot searches, aggregate data, visualize query results

Example Hunt

“An attacker has used PowerShell to download malware using obfuscated or encoded commands”

Example Hunt

1. Identify relevant logs
    a. Windows Event Logs
    b. Sysmon logs
    c. Network data (DNS, firewall, etc)
2. Execute initial queries
    a. Use correct index(es)
    *b. index=sysmon EventCode=1 Image=”*powershell*” CommandLine=”*enc*”*
    c. Analyze results to determine next steps
3. Pivot as necessary

Additional Queries

```
index=sysmon EventCode=1 Image="*powershell*" CommandLine="*update.ps1*"
index=wineventlog EventCode=4688 New_Process_Name="*powershell*"
    Command_Line="*update.ps1*"

index=sysmon EventCode=1 CommandLine="*update.ps1*" |
    table _time, Computer, Image, CommandLine, Hashes

index sysmon EventCode=1 Hashes="*<hash>*" |
    table _time, Computer, Image, CommandLine, User, Hashes
```

Best Practices

- **Remember indexes!**
- Ensure time range is correct
- Include *sourcetype* if known
  - Reduces scope of search
- Use fields when searching
  - *Image=”*powershell*”*
- Wildcards only when needed
  - Can affect performance

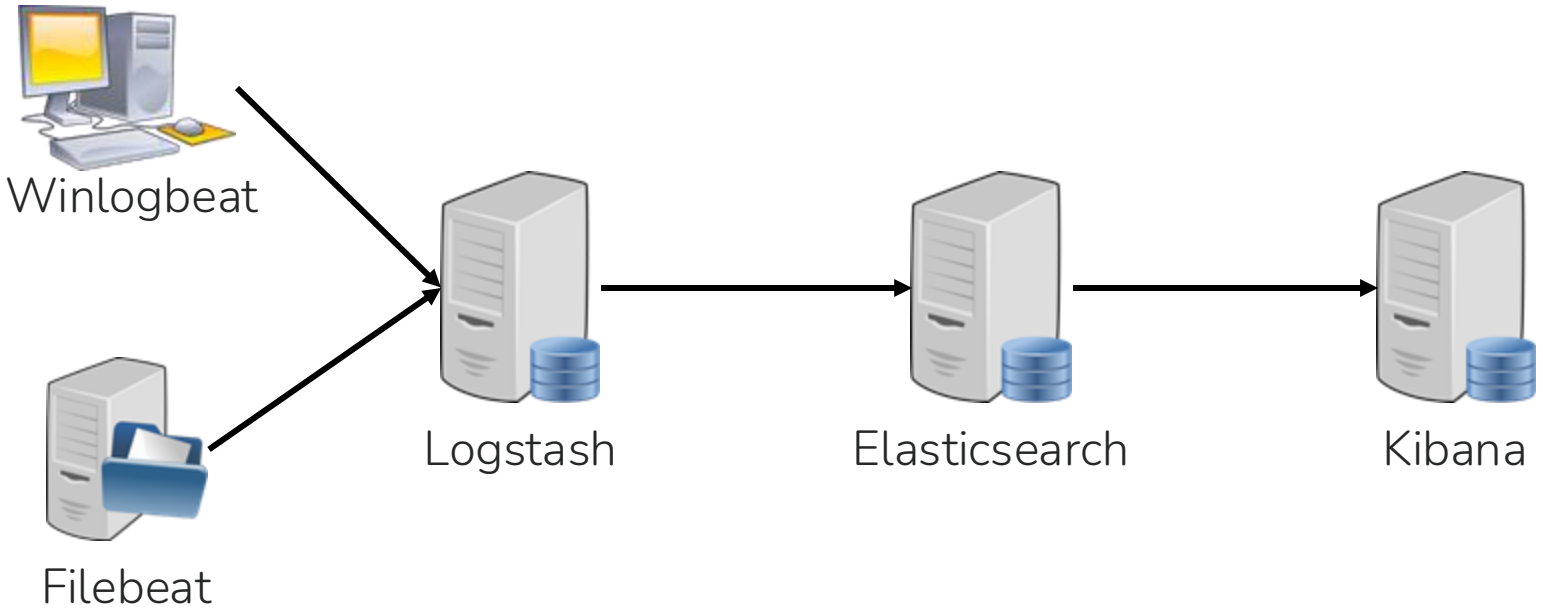
Practical Splunk Threat Hunt

ELK Introduction

What is ELK?

- Three components working together
  - **E**lasticsearch
    - Search and analytics
  - **L**ogstash
    - Data processing and transformation
  - **K**ibana
    - Visualization and dashboards
  - Often includes Beats (Winlogbeat/Filebeat) - used for shipping log files
- Similar uses and functionality to Splunk
  - Visualizations and real-time searches

ELK Architecture



ELK Search Languages

Searching in ELK

- ELK supports multiple query languages
- KQL - Kibana Query Language
  - Faster, simpler queries
  - Field-based matches
- EQL - Elasticsearch Query Language
  - Event correlation
  - Sequences (multi-step searches)

KQL

- Field-based searching
  - : - equals
- Boolean logic
  - AND, OR, NOT
- Wildcards
- Field selection for ad hoc filters
- Best practices
  - Field-based searches instead of full-text
  - Save commonly used queries

EQL

- Excels at searching multi-step events/attacks
- Simple query
  - event_type where condition
  - process where process.name == “powershell.exe”
- Logical operators
  - ==, !=, <, >, and, or, not
- Sequences
  - “This happened, then that happened”

```
sequence by host.name with maxspan=5m
  [process where process.name == "cmd.exe"]
  [network where destination.port == 4444]
```

Common
ELK Queries

Common Queries

- Suspicious account logins
- Suspicious process behavior
- Unexpected outbound network connections
- Suspicious PowerShell activity
  - Encoded commands
  - Downloads
- Looking for persistence techniques
  - Scheduled tasks
  - Malicious services

Threat Hunting with ELK

Example Hunt

“An attacker has dumped the SAM database to steal password hashes and used them in a Pass-the-Hash attack to authenticate across the network.”

Example Hunt

1. Identify relevant logs
    a. Windows Security Logs
    b. Sysmon logs
2. Build initial queries
    a. *winlog.event_data.Image:"*reg.exe* and winlog.event_data.CommandLine:"*save*" and winlog.event_data.CommandLine:"*\\HKLM\\SAM*"*
    b. Analyze results to determine next steps
3. Pivot as necessary

Additional Queries

*winlog.event_data.CommandLine:(”*procdump.exe*” or “*mimikatz.exe*”)*

*event.code:4624 and message:”*ntlm*”*

*winlog.event_data.CommandLine:(“*wmic.exe*” or “*psexec.exe*” or “*smbexec.py*”) and NOT winlog.event_data.User:”*SYSTEM*”*

*winlog.event_data.ParentImage:”*wmiprvse.exe*” and winlog.event_data.CommandLine:(“*cmd.exe*” or “*powershell.exe*”)*

Practical ELK
Threat Hunt

Endpoint Threat Hunting - Summary

Key Concepts - Recap

- Endpoint hunting intro
- Threat hunting in Windows
- Hunting using SIEM applications
  - Splunk
  - ELK

Learning Outcomes Recap

+ **Recognize use of malicious Windows services and processes**
+ **Understand the importance of Sysmon and how it's used in endpoint hunting**
+ **Be able to perform effective hunts in both Splunk and ELK systems**

Next Steps

- Continue with Threat Hunting learning path
- Revisit any courses or videos on endpoint IOCs or Windows system fundamentals
- Continue practice in hands-on labs
- Begin preparing for eCTHP exam (if taking)

Thank you!

Endpoint Threat Hunting

+ Endpoint IOCs
+ Windows services, processes, Sysmon
+ Hands-on hunting in Splunk and ELK
+ Thank you for your time!

EXPERTS AT MAKING YOU AN EXPERT